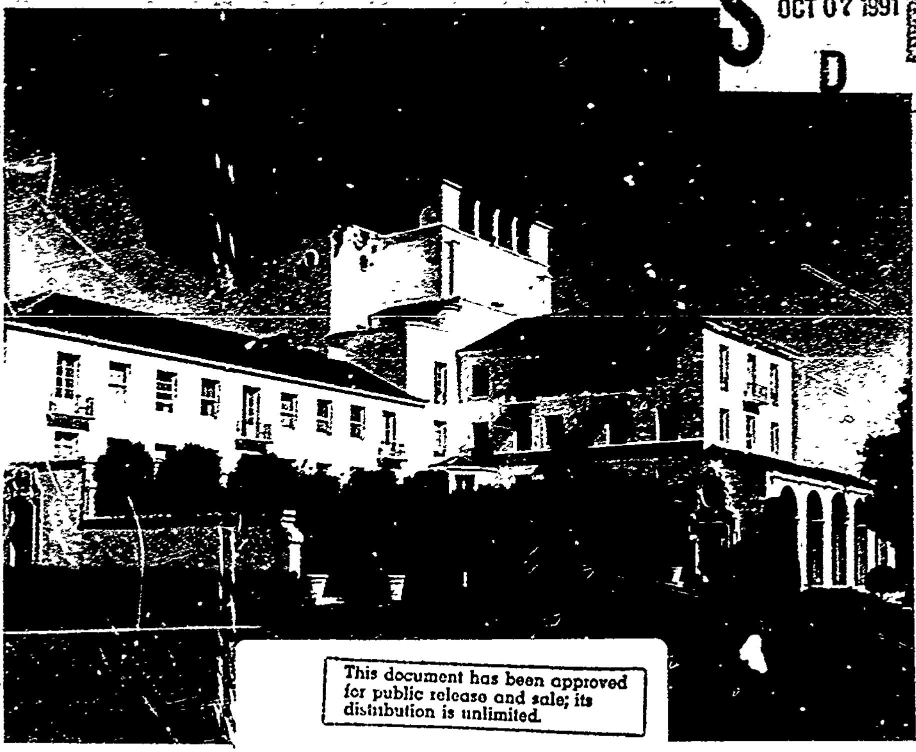
1988 Symposium on Command and Control Research



Basic Research Group,
Joint Directors of Laboratories
and
National Defense University

Science Applications International Corporation
McLean, Virginia 22102

September 1988

Proceedings of the
1988 Symposium on Command and Control Research

Held at

Naval Post Graduate School

and

Monterey Resort Inn

Monterey, California

June 7-9, 1988

Sponsored by

Basic Research Group

Technical Panel on C^3

Joint Directors of Laboratories

and

National Defense University

and in Cooperation with

IEEE Control Systems Society

Information Systems Division

Science Applications International Corporation

McLean, Virginia 22102

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER: SAIC-88/1866	2. GOVT ACCESSION NO.
	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Proceedings of the 1988 Symposium on Command and Control Research	5. TYPE OF REPORT & PERIOD COVERED: Final, 7-9 June 1988
	6. PERFORMING ORG. REPORT NUMBER: SAIC-88/1856
7. AUTHOR(s): Inara Gravitis (compiler)	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Science Applications International Corporation, 1710 Goodridge Drive, McLean, VA 22102	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Command & Control Research Program, INSS, National Defense University, Ft. Lesley J. McNair, Washington, DC 20319	12. REPORT DATE: 30 September 1988
	13. NUMBER OF PAGES: 638
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office): JDL-TPC^3, C.O.V.J. Monteleon, Naval Ocean Systems Center, San Diego, CA 92152	15. SECURITY CLASS. (of this report): UNCLASSIFIED
	15a. DECLASSIFICATION/DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Unlimited

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

Contains remarks of 3 plenary speakers and papers presented in working group sessions.

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Command, Control, Communications; C^2; C^2 Research; Theory and Modeling of C^3; Decision Aids; Behavioral Aspects of C^3; Testing and Evaluation of C^3.

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

Presents results of basic research in Command and Control in four areas; Theory and Models of C^3, C^3 Systems and Components, Decision Support Systems and Behavioral Aspects of C^3, and Testing and Evaluation of C^3.

DD FORM 1 JAN 73 1473 EDITION OF 1 NOV 65 IS OBSOLETE

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

FOREWORD

"It wasn't the Roman Legions that crossed the Rubicon, it was Caesar." After all, the Roman Senate had at its disposal twice the number of legions that Caesar commanded. They were well rested, well provisioned, and occupied good defensive positions. Why, then, was there no battle?

The question is, of course, rhetorical. There was no battle because the defenders had no stomach for standing up to Caesar, the man who had proven himself to be the outstanding commander of his day. His skill at planning was superb, his ability to react to developments on the battlefield was unmatched and, perhaps most significant, he never lost "control" of his troops. As he committed his forces to battle, he always retained an avenue of escape, and an ability to disengage his forces if the battle were going badly. This kept tactical setbacks in battle from becoming catastrophes.
The above is just one example of a fighting force that overcame enemy forces (otherwise every bit as capable) due to better command and control.

Ten years ago, a group of analysts gathered under the sponsorship of the Office of Naval Research and the Massachusetts Institute of Technology to ask the question. Are there elements that can be identified that make for successful command and control on the battlefield? Can these be identified, quantified, analyzed, and taught?

This initial gathering grew into a series of nine annual workshops on Command and Control that presented pioneering work in the field and established command and control research as an area of basic inquiry. Perhaps more important, their influence caused the defense community to realize that good C^3 depended on more than good communications and automatic data processing equipment. A Defense Science Board task force reached the same conclusion and it established a Command and Control Research Program at National Defense University to encourage and co-ordinate research in command and control with a special emphasis on joint operations. Three years ago, it was recognized that the field had matured sufficiently to pass the sponsorship to the Basic Research Group of the Joint Directors of Laboratories, an organization that has full participation from all three military departments.

The 1988 Symposium on C^2 Research was held at the Naval Postgraduate School and the Monterey Resort Inn in Monterey, California, from the 7th to the 9th of June This was the second annual Symposium sponsored by the Basic Research Group of the Technical Panel on C^3 of the Joint Directors of Laboratories The Symposium was organized into four working groups, each one of which provided a track throughout the three days The program included five plenary sessions, four of them featured a presentation by an eminent speaker, while the fifth one provided an opportunity for assessing the symposium and the work presented. This volume contains most of the papers presented at the symposium.

The first plenary speaker, Vice Admiral Jerry O Tuttle, USN, director of J-6 (Command, Control and Communications Systems) of the Joint Chiefs of Staff, discussed C^3 needs from the commander's perspective His compelling, and entertaining remarks impressed on the participants the fundamental shift that has taken place in the defense community The community recognizes that effective command and control is a vital ingredient in military success, every bit as important as good communication systems and indeed good weapon systems He challenged the symposium to explore the fundamentals of command and control and help the defense community develop materials that could be used in instruction and training of senior military officers

The second plenary speaker, Dr. Harold W. Sorenson, Chief Scientist of the Air Force, talked about lessons learned and changes in perspective during his tenure as chief scientist regarding the development and use of C^3 systems. Mr. Edward Brady, vice president of MITRE, addressed C^3 problems from the CINC's perspective using a top down approach. Finally, Dr. Harry L. Van Trees, presented a personal but comprehensive view and assessment of progress in C^3 research during the last ten years. We are very pleased that several of the speakers provided a written version of their talk for inclusion in this volume. These papers are to be found in the first section, Invited Papers. We also take this opportunity to thank the four speakers for their presentations and for their participation in the lively discussions during the symposium. They set the tone for the technical program of the meeting and provided the basis for many of the discussions during and between the sessions.

The first working group, Theory and Models of C^3, was chaired by Dr. Richard P. Wishner from Advanced Decision Systems. The presentations ranged from the most recent update of the C^3 Reference Model to mathematical models of the C^2 process, and to specific technical problems such as dynamic resource allocation or data fusion.

The second working group, C3 Systems and Components, was chaired by Prof. John H. Lehoczky from Carnegie-Mellon University. From the twenty three papers originally scheduled, eight were not presented, primarily because of the freeze in travel that was in effect at the time of the symposium. Some of the authors who were not able to attend did submit their papers and since the circumstances were beyond their control we broke with custom and included the papers in this volume. The papers in Section 3 emphasize communications, computer science, and networks.

The third working group, Decision Support Systems and Behavioral Aspects of C3, was chaired by Dr. Stanley M. Halpin of the Army Research Institute. Sixteen papers are included in Section 4. They range in content from organization theory to presentation of experimental results to knowledge based decision aids. The human element and its impact on system performance and system design constituted one of the main threads in the sessions.

The fourth working group, Testing and Evaluation of C^3, was chaired by Dr. Stuart Starr of the MITRE Corporation. This group of papers was characterized by the strong effort by the chairman to integrate them and present a sweeping view of the problems, issues, and approaches to testing and evaluation. Thirteen of the twenty three papers are included in Section 5.

On the last day of the Symposium, a questionnaire was distributed to the attendees with the request that the fill it out anonymously. The purpose of the questions was to identify the strengths and weaknesses of the Symposium design and solicit ideas for its improvement. There are three issues that a substantial number of respondents raised. The first was that the demarkation between working groups was not clear, with the result that attendees had difficulty selecting the papers that interested them or that similar material was presented in different working groups. In the current symposium design, prospective authors indicate the working group they prefer and, in general, their wish is respected. The working group chairmen grouped the selected papers into sessions in the best way possible. The process was done in a decentralized manner. An alternative design is for a program committee consisting of the four working group chairmen and the Symposium chairmen to consider all the papers together, organize the selected papers into cohesive sessions and then organize the sessions into tracks. This approach requires increased effort by the program committee members, but the result is expected to be an improvement in the organization of the technical program. A side benefit of the alternative approach is higher and more uniform standards for the selection of papers.

The second common comment was that there were too many papers and a person had difficulty not missing presentations that he wanted to hear. There were a number of suggestions to reduce from four to three or even two the number of parallel sessions. We believe that part of the difficulty could be resolved by organizing the symposium by session rather than by working group so that parallel sessions do not have overlapping scope. We do not believe that the number of parallel sessions should be reduced further; note that in 1987 there were six parallel sessions and in 1988 there were only four.

The Symposium is a unique forum that brings together persons who are doing basic research on C^2, whether in academia, government laboratories, or industry, sponsors and users of the basic research, and some - not enough yet - from the operational community. C^2 has a very broad technological and behavioral base and we have been striving to capture this breadth in the Symposium. The individual technologies or sciences have disciplinary forums such as the technical meetings organized by professional societies in communications, computer science, control theory, psychology, etc., where results can be presented. One of the roles of the Symposium is to keep reminding the attendees of the breadth and complexity of C^3 so that, as the saying goes, we do not miss the forest for the trees.

The third comment concerned the proceedings and the need to have the papers distributed early. At the time of this writing the expectation is that the proceedings will be distributed in September, two months earlier than in 1987. It is not possible to have them ready much earlier than that, unless authors are requested to submit their papers before the meeting. Many authors prefer to present the paper first, receive feedback and then write the version to be published. In this way, the most recent work in the area appears in the written record.. While every effort will be made to publish the proceedings in as timely a manner as possible, for 1989 we will continue the practice of distributing a complete set of abstracts, better organized and with an index, at the Symposium, to be followed by the Proceedings soon after.

Finally, we would like to thank the four working group chairmen, Dick Wishner, John Lehoczky, Stan Halpin, and Stu Starr whose technical judgement, hard work, and commitment produced the 1988 Symposium which was, according to your comments, a success. We would like to express on behalf of all of the attendees our appreciation to Ms. Inara Gravitis of SAIC who, as the administrator for the Symposium, made it happen.

We are already planning the 1989 Symposium that is to be held at the National Defense University, Fort Leslie J. McNair, Washington DC on June 27 -29. We expect to see you all there.

Alexander H. Levis *Stuart E. Johnson*

Co-chairmen

TABLE OF CONTENTS

Page

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

TABLE OF CONTENTS (Continued)

Section I

Invited Papers

C3, AN OPERATIONAL PERSPECTIVE

Vice Admiral Jerry O. Tuttle
Director, Command, Control and Communications Systems Directorate
The Joint Chiefs of Staff

Good Morning, Dr. Johnson, Dr. Jones, Ladies and Gentlemen, leading scientists of the world. I am delighted and humbled to address such an august group.

All of you are the reason why I am in my current position as the director, Command, Control, and Communications Directorate, in the Joint Chiefs of Staff and why I am here today. Because of your assistance during my 40 months as a battle group commander, some people mistakenly thought that I know a great deal about C3 systems. I owe you and your counterparts an enormous debt of gratitude.

Early in my tour as a carrier battle group commander, I recognized the tremendous resource of knowledge and technology those of you in our DOD laboratories possessed. Until then my command and control universe had been limited to the unnecessarily restricted radius of 200-300 nautical miles of the customary range of action for the carrier air wing.

I immediately sensed that a carrier battle group should control the battlefield under an arbitrary 1,000-nm envelope. Nevertheless I was sagacious enough not to specify any distances lest they be considered limits and chiseled in stone. I did not want to accept any limits. From a commander's viewpoint, I had created a major C3 challenge. With the then-existing systems, I might just as well have been in a 55-gallon drum.

Fortunately and solely by chance, I turned to you in the laboratories to satisfy my requirements because I realized that the system was too slow to meet my needs during my tour. You responded magnificently. Through frustration and necessity, I discovered the tenets so ably articulated in the National Defense university's and AFCEA's Science of Command and Control: Coping with Uncertainty I recommend it to you for reading. I read only the foreword and introduction, and purposely did not read any further. It was clear that the editors/authors had masterfully captured the essences of Command, Control and Communications (Dr. Stuart Johnson and Dr. Alexander Levis, editors). Reading further would have conditioned me to restate unintentionally what you already have available. Besides, I have some other insights that I want to share with you.

Frankly, the challenges and responsibilities for resolving those existing C3 shortfalls reside primarily with me and are not due to a lack of technology. Clearly, there are unconquered and undiscovered spheres of research and technology that remain and existing technology that needs to be applied and/or matured. The most fruitful areas reside in improving the "System"; Management; Changing Cultures; and yes, Leadership.

I am totally convinced that experts like you can engineer, design, and package C3 systems to satisfy our every recognized C3 requirement if the users can properly articulate what is needed. That is the gulf we must bridge. As your director for C3 systems, I hope to contribute to and exploit the vast capabilities that already exist. I am confident about our prospects. This symposium, as well as books like Science of Command and Control: Coping with Uncertainty (Dr. Johnson--Frank Synder, NWC--Tony Ottinger, Harvard--Carl Jones), helps to inspire this optimism.

My staff had prepared a "Strawman" for this address. Frankly, it missed the mark because it was prepared by those who were products of the existing system.

The laws of physics have not changed since creation. Our creator kindly provided us with a spectrum between D.C. and light, and you gentlemen are masters of exploiting this spectrum. Fiber optics are an exciting e. ple of a capability that still begs to be fully exploited.

Let me tell you where I have channeled my energies during this past year for your critical review and conclude with an ongoing veritable C3 success story that needs to be told and one in which you should feel pride, because you have made it possible.

Permit me to proceed through the salient topics in more or less chronological order.

When I was informed of my current assignment, I thought that I should learn about the Worldwide Military Command and Control (WWMCCS) ADP System. To me, they were just letters in the alphabet. While at CINCLANTFLT, I arranged to be given a "Dick and Jane show and tell."

An Army Major struggled through a hands-on demonstration but frequently deferred to a lady who worked on the staff in logistics. She politely provided many of my answers, but I came away with the perception that WWMCCS ADP was murderous to the user, a "White Elephant," correctly put antiquated, and being used by the wrong people. More accurately, the people who should have been using the system were not. The worldwide military command and control ADP system was not being used to command and control anything.

In order to become familiar with the system, I had a WWMCCS intercomputer network (WIN) terminal installed in my office prior to my arrival. My discovery was astounding, shocking, unbelievable, and depressing. The fastest, most secure, pervasive C3 system was not being utilized by operators. (They did not know how.) No terminals had been installed in our numbered fleet flagships. There were no "yellow pages" (that is, a list of who were on the system) and no catalog of what programs or files were available in the system. Most amazing of all, the chairman of the Joint Chiefs of Staff could not send his CINC's messages due to a lack of procedures. I discovered that more data were transferred through the ether every night by the joint deployment agency, those concerned with logistics, etc., than went over the autodin system.

Believing in the Christopher's motto aphorism that it is better to light one candle than to curse the darkness, I initiated an awareness program, recommended to the Operations Directorate on the Joint and Unified/Specified Commands use of the existing system, and openly marketed its capability.

Today, the chairman can send messages immediately to any or all of his CINC's through a teleconferencing net; OPLANS, OPORDS, warning orders, alert orders, and execute orders are routinely sent to the CINC's and joint task force commanders; all numbered fleet flagships now have WIN terminals; OPSEC has improved greatly; and the CINC's are now unanimously in favor of the follow-on WWMCCS information system or "WIS." Today, CINCCENT, CINCLANT, MAC, and CINCSOUTH use the WIN system daily for command and control of real world contingencies. This is quite an accomplishment when one considers that some of their operators could not log on to the system as recently as December 1987.

WIS had experienced a turbulent funding history and could not have survived without the successes demonstrated on the WIN network. Despite the rocky road of WIS, at 5 test sights early next year we should field a local area network; an automatic message-handling system; and the first increment of the Joint Operational, Planning, Execution, SSystem (or JOPES) or a link-up of the Joint Deployment System and the Deliberate Planning System.

That's the good news! The bad news is that is should not have taken that long or cost as much as it did to design and field a local area net and an automatic message-handling system The blame certainly does not rest with you in the laboratories. However, it does lie with those who failed to state the requirements adequately, failed to include the users from the beginning, imposed incredibly stringent test and evaluation requirements, established an unwieldy organization whereby no-one could be held accountable, and then tolerated poor management until the past year.

Where are we headed? Hopefully, the release of GECOS-8 will make the WIN far more capable, faster, user-friendly, and greatly expand the sample of users. WIS will come to fruition and with the assistance of a system that I have put together during the past 6 years define the execution phase of JOPES better.

Being from outside the receptive C3 community when I reported a year ago, I wanted to remain open minded in order to learn where the problems were while not revealing my ignorance and thereby losing credibility. I became actively involved with the C3 professional groups, such as AFCEA, IEEE, AIAA, and EIA. They provided me with valuable feedback. I learned that we were not making our requirements known to industry. We were not making available our stated joint required operational capabilities (ROC's) or multiple required operational capabilities (MROC's) until we went out for RFP's. This seemed myopic to me, particularly in view of the fact that as joint operations become far more fashionable, unquestionably there would be more joint ROC's/MROC's. I wanted industry to be working on what we need, not what they thought we need.

We immediately made our ROC's/MROC's available to industry so that their technological bases could be applied to our requirements sooner. Industry, I hasten to add, has been alert to making the necessary changes in how they conduct business since the enactment of the 1986 reorganization act.

By reviewing our ROC's/MROC's for release, we discovered that some had been overtaken by events; that is, the RDJTF no longer existed, etc. Most were poorly written. It was taking up to 3 years to validate requirements. No review was being conducted to measure the effectiveness of the services to satisfy the CINC's requirements, etc. I found that the directive that governed the ROC validation process was of 1982 vintage and seriously flawed. We have rewritten it to speed up the process greatly, provide the CINC's with a quarterly situation report, and participate actively in all phases of satisfying the requirement from validation to full operational capability.

For openers, I found that we mistook "feasibility", "practicality", "affordability", and "priorities" for "requirements." I can count to four. If one of those CINC's says he has a requirement, then this three-star (and certainly not a committee of iron-majors) is not going to tell him that he does not have a requirement. If General Piotrowski, CINCSPACE, says that he needs to talk with Voyager on the ether side of Pluto in less than 4 hours, then he has a requirement. I then come to you to bail me out and find some transmission media faster than light.

The entire C3 operational requirements process has made major progress, but it is still unsatisfactory. Nevertheless, we will prevail.

There will be an ever-increasing number of joint requirements as we conduct far more joint exercises. This is good because it will nurture interoperability. Since Grenada, we have come light years in our ability to interoperate. However, it is not enough that the US Air Force, Marines, Army, and Navy are interoperable. We must be able to operate together with our allies while using compatible C3 systems. This encompasses language, procedures, technologically compatible C3 systems, doctrine, etc.

We are actively pursuing all of these areas, and, frankly, I am pleased with progress. I do not believe that we are mistaking activity for achievement. B. Gen. George Bombel is doing a brilliant job in leading the joint tactical C3 agency.

Interoperability must start from standards. We in J-6 simply were not taking an active enough role in establishing standards and in ensuring that C3 system developers were rigidly adhering to these standards. I do not want to reveal that embarrassing state of affairs, but we are very much involved in creating standards and ensuring compliance. Some of the more salient areas that we are pursuing are fiber optics, HF anti-jam wave form, over-the-air crypto rekeying (more on that later), etc.

Perhaps the most significant accomplishment regarding standards has been the utilization of an existing standard -- the message text format (MTF) standard. When, by fiat, JINTACCS MTF became the standard message format for the 132 selected messages on 30 September 1986; for various reasons (primarily ignorance) it was considered that they should be utilized for exercises or contingency operations only. Admittedly, many of the message formats were for shore bombardment, naval gunfire support, etc. Nevertheless, it is foolish to introduce an unfamiliar procedure or system when embarking on contingency operations. Clearly, we had to find a way to familiarize everyone with MTFS by using them routinely.

To provide leadership, I directed that all messages drafted by my directorate be in MTF. The response by the CINC's has been phenomenal. As recently as this past December, for all practical purposes, the Military Airlift Command did not use MTF. On their recently concluded exercise, 98 percent of all messages originated by MAC, and 82 percent of the messages received from their components were in standard MTF.

What we have discovered by daily use of MTF is that our automatic message distribution systems require reprogramming. This discovery led to the realization that we had three different automatic message distribution codes -- one for Air Force, another for Navy, and yet another for NATO -- all of which were inefficient computer programs. As I speak, we are working on a standard, efficient automatic message-routing code for us and our allies.

As an operator, one of my objections to JINTACCS MTF was to an air gap and message format change at the joint task force level. Because the joint task force is the last place that you want a discontinuity, we have now corrected this discrepancy, and the system is transparent from the unit level to the national command authorities. The message I want to deliver with the MTF story is that it is another example of a capability that existed but was being seriously restricted by ignorance and lack of procedures. It also provides a convenient lead-in to my next topic.

No system is any faster than its control mechanism. Regrettably, our C3 system is far too dependent on recorded messages. This is unacceptable, certainly for air defense and most other dynamic scenarios. Even those who graduated magna cum laude from Evelyn Wood's speed reading course cannot envision the picture from a paper-based system. By achieving universal use of the machine-readable MTF, at least we will be able to correlate MTF reports with other faster sensor-derived data in order to reduce the uncertainty addressed in the National Defense University book: Science of Command and Control: Coping with Uncertainty. It is this uncertainty that I want to ameliorate, having been in command of something for the greater part of my Navy career.

I am confident that, with minimum resources, I can cause numerous existing data bases to be fused in order to form a common red and blue data base that can be used at all echelons of command -- from the national command authorities to the individual missile shooter on scene. The data base at each tier in the echelon can be tailored locally by granularity or geography to suit the commanders' needs.

This tailored composite red/blue picture must be projected in a manner so that it can be conveniently, constantly, and critically analyzed and challenged.

The creation and management of these data bases can be made far more automated, and a more equitable division of labor in intelligence production at all echelons can be achieved while reducing redundancies. There can be economies made in personnel resources, but, more importantly, alcon (all concerned) will be operating from a common data base.

I have initiated this effort in the National Military Command Center, which currently has 115 individual systems (a number clearly unmanageable), no composite picture, and no way to project it automatically if we had one. That is not totally correct, inasmuch as we did throw together a system literally over night in the Operations Center Decision Room (ODCR) for the recent U.S./Iranian encounter in the Persian Gulf.

As a result of our being able to project on a large-screen display near-real-time locational data for units in the north Arabian Sea and Persian Gulf, Secretary of Defense Carlucci and Chairman, JCS, Admiral Crowe could make far better and faster decisions. As a matter of fact, they could sit back and observe General Crist in Florida orchestrate the entire affair. Of course, we had provided General Crist a like capability.

With the on-scene commander, RADM Tony Less, the CINC (Gen. Crist in Tampa, Florida), and SECDEF/Chairman all having the same picture and same data bases, the requirement to communicate diminished markedly.

By having the red and blue forces depicted in one composite picture, the relative urgencies for decisionmaking could be readily determined and priorities made more intelligently.

How have we arrived at this juncture? Certainly not for the lack of technology. It has been accomplished by knocking down barriers and changing a culture.

It started from a Joint Operational Tactical System (JOTS) that had evolved from a desktop computer, tactical decision aid, and planning tool to a management information system and thence to a C3 system -- first for a carrier battle group commander, then for joint task force commander, then for a numbered fleet commander, and finally for a fleet commander. Now we are developing it for the world duty officer.

At this juncture, I want to point out an illness in our systems approach that must be ameliorated. It is natural and easy to see why it happens, and it is the reason why we had 141 separate C3 systems in the NMCC. Understandably, any system's requirement is articulated from the perspective of the echelon of command for whom the requirement exists. There is little or no consideration of where the system that satisfies these requirements fits into the whole.

The situation is further exacerbated by the executive agent and program manager for the system because they want and need to have limits on their project.

There is a project rapidly maturing today that feeds from multiple external sensors, but it is a closed system in order to satisfy a single-attack option when its unique sensor could be used to great advantage by many other weapons systems. I must arrange to have this system interoperate digitally with many other systems.

The worse examples of this can be found in the intelligence systems. I call them "stovepipes." This is not an indictment of the intelligence community by any stretch of the imagination, but rather a recognition of the fact that the titular head for all of their systems is a J-2, or intelligence officer -- not the officer in charge who has the ultimate responsibility in his Area Of Responsibility (AOR).

Currently, intelligence consumes approximately 65 percent of our communications capacity and is expected to increase to 85 percent during a conflict. Clearly, we need to go on a communications diet and analyze data as opposed to handle data. I am working with DIA and NSA in order to achieve a more comprehensive and composite picture for all echelons of command while burning up less of the airways.

Returning to the Joint Operations Tactical System -- JOTS, I was demonstrating its capability to a JCS operations officer one afternoon after having just reported aboard. He said: "That's great! Have it in place for tomorrow's first reflagged tanker escort mission" (Operation "Ernest Will").

We worked all night getting the connectivity for the JOTS picture back to the Pentagon. We took the officer in tactical commands (OTCIXS) JOTS broadcast off the Indian Ocean satellite and had my friends in Naples rebroadcast over the Atlantic satellite to CINCLANTFLT in Norfolk and over landline to me in the crisis action room. Wanting to bet on aces, straights, and sure things, I did a like "M" hop in the pacific through Guam and NOSC, San Diego.

Now the connectivity is not impressive and, in fact, is routine, but what occurred that morning at 0630 when SECDEF and CJCS watched (near-real-time, at the time of TV) the <u>Ernest Will</u> ships steam through the Straits of Hormuz and out of the silkworm envelope changed a culture.

Afterwards, SECDEF picked up the red phone and said: "Chris, super job," he departed saying that this was the first time that we had beat CNN. What General Crist could not figure out was why SECDEF and the chairman had been so quiet and had not asked any questions.

Well, I rapidly became the most popular three-star in DOD. Every four star whose area of responsibility I had used to pipe a picture of General Crist's AOR to Washington, called me and "reviewed my heritage".

General Crist had changed it all, for he intrepidly said that he did not care if his superiors had any, and all, information as long as he had the same information at the same time. We provided him with the same capability that we possessed, and we have grown together ever since.

The other morning around 0530 it was gratifying to be able to sit there and have the complete picture. However, there is another major issue here. All depicted data were being challenged constantly and by every echelon of command. If the same information had been in message format or in a data base, the degree of uncertainty would justifiably have been greater.

We did not carry the red, blue, commercial, or unknown air tracks that day because (RADM Tony Less) thought that if every 5 minutes JOTS reporting was good, every minute would be great. So he strangled the system.

I am so committed to developing this system that will provide worldwide tactical C3 to the NMCC and all the CINC's for contingencies covering the spectrum from low-intensity conflict to theater nuclear war that I have established an entire division on my staff to work the problem.

I want this tactical C3 system to be used continuously, be reliable, be fairly robust, and become ever-increasingly less vulnerable. It is my "Use, Learn, Develop" philosophy.

To utilize the foregoing and existing C3 systems better, we have just concluded an over-the-air rekeying for crypto devices worldwide exercise. We demonstrated that we can send a crypto key list over existing communications channels covered by KG-84's and Vinson crypto devices -- a capability that has existed for a long time but had never been utilized. I can not even imagine the amount of money, trees, and careers that will be saved by this procedure.

As I approach the end of this presentation you will note that I have not asked you for a thing. The reason is that until I can fully exploit the capabilities that you have already provided, I do not intend to ask for more. Just continue your superior work.

By the way, if you solve the multi-level security problem, I'd be very interested. We are devoting tremendous resources to this issue and may be demanding of our hardware and software engineers unrealistically lofty requirements and an electronic system to compensate for a lack of discipline in our security system.

We need to work together on two things:

1) Embedded crypto in our C3 systems

2) GPS weapons systems integration.

Now I would like to conclude with a success story that is largely unrecognized and one for which you are largely responsible.

Let's mentally take a trip halfway around the world to the mouth of the cradle of civilization. There we see French, British, Belgian, Dutch, Saudi, and Italian ships steaming in hostile waters with numerous oil tankers, ships carrying military cargo, Iranian combatants, and (yes!) a sizable U.S. naval presence.

All of these link-capable ships are in the U.S.'s and/or Saudi AWAC's link and the E-2C's link when present. The U.S. ships can communicate on secure with the allied ships in the theater, with aircraft for deconfliction with the reflagged tankers, ashore with all of the gulf coast states' Air Forces and Navy headquarters, and with our embassies.

Now consider that the CINC who has the ultimate responsibility for this theater resides on the other side of the world, 7,000 miles away. His Navy component commander is in Hawaii; his joint task force commander embarked in his sixth flagship in the north Arabian Sea; and his surface commander is in the gulf. It is testimony how a C3 system can be engineered to support the war fighter. This scenario is even more noteworthy when you realize that this was a virgin C3 area a year ago and that some who are now communicating over secure circuits would not have exchanged greetings before if they had met on the street.

Thank you for permitting me to participate in your symposium. I hope that it, like the county fair, gets bigger and better every year.

I leave you with this thought: turbulent progression is preferable to tranquil stagnation.

ROLE OF THE C^2 RESEARCH COMMUNITY IN SUPPORT OF THE COMMANDERS-IN-CHIEF (CINCs)

Edward C. Brady
Stuart H. Starr

The MITRE Corporation

A. Introduction

Recently the Scientific Adviser to the Supreme Allied Commander Europe (SACEUR) wrote to a member of the Joint Directors of Laboratories (JDL) to comment on the most recent JDL C^3 Research and Technology Program (reference 1). Based on discussions with SACEUR's staff he concluded: "The general feeling was that the program plan focused on the theoretical aspects and was too detached from real world C^3 combat effectiveness. We need to understand and develop C^3 technology to support operational needs. Thus, this program needs to be a balance of theory and empirical measurement."

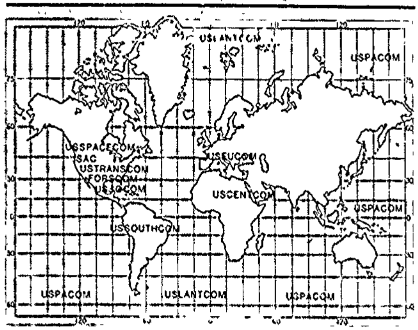
These observations pose a critical question for the C^2 research community: what should be its role vis-a-vis the roles and missions of the Unified and Specified (U&S) Commanders-in-Chief (CINCs)? To address that question, this paper describes and characterizes the evolving responsibilities of the U&S CINCs. Emphasis is placed on the enhanced role of the U&S CINCs arising from recent institutional initiatives and the limitations that the U&S CINCs face in executing these expanded responsibilities. In light of these needs, the position of the C^2 Research Community is characterized. This includes a description of the community's current support to the U&S CINCs and a projection of the role that it could ultimately play. Based upon these assessments, several actions are proposed to make the C2 Research Community more responsive to the needs of the U&S CINCs.

B. CINC Responsibilities and Needs

1. CINC-Unique Features

Figure 1 broadly depicts the geographical responsibilities/peace time locations of headquarters for the ten U&S commands. The precise composition of these commands has changed over the last few years with the creation of new commands (e.g., U.S. Special Operations Command (USSOCOM)), the evolution of existing commands (e.g., creation of U.S. Transportation Command, assimilating the Military Airlift Command, Military Sealift Command, and Military Traffic Management Command), and the disestablishment of prior commands (e.g., U.S Readiness Command)

Figure 1
Unified and Specified Commands



From the perspective of the C^2 Research Community there are several characteristics of the U&S CINCs that affect the support they require.

First, the U&S CINCs can be characterized by the nature of their mission, warfighting versus support. For example, USTRANSCOM exists to support the other CINCs in the execution of their operational responsibilities and this support role strongly affects their C^3 needs. Second, the U&S CINCs can be distinguished by the locations of their headquarters vis-a-vis their theaters of responsibility. For example, USCENTCOM has its headquarters in Tampa, Florida, while its area of responsibility is Southwest Asia This has serious implications on its C3 infrastructure needs. Third, the level of C3 system support from the Services and Defense Agencies varies significantly across the U&S CINCs. For example, close ties exist between CINCSAC and the Air Force and CINCLANT and the Navy. In addition, Defense Agencies have set up several field offices to assist the CINCs on C3-related activities (e.g., DCA has a European office collocated with USEUCOM in Stuttgart, FRG, and a Pacific office collocated with USPACOM in Hawaii) However, there are cases, most notably US CINC Southern Command (USCINCSO), where organic and external support for C3 systems are extremely limited. Fourth, it should be recognized that USSOCOM is the only U&S CINC with direct research, development, and procurement authority

These observations suggest that each U&S CINC must be analyzed individually to identify the specific assistance that it requires from the C2 Research Community. However, as shall be explained below, there are several roles that the U&S CINCs play that imply many common areas of need in C3 support.

2. Enhanced Roles of CINCs

In recent years, institutional initiatives have enhanced the roles of the U&S CINCs in four areas: planning, programming and budgeting; requirements generation, development and acquisition, and training. This section reviews those initiatives, emphasizing their impact on C3 activities and the increased demands that they place on the U&S CINCs.

a. Planning, Programming, and Budgeting

The DoD deals with the process of resource allocation among the Services and Defense Agencies through an integrated Planning, Programming, and Budgeting System (PPBS). The enhanced role of the U&S CINCs in each phase of the PPBS is discussed below.

(1) Planning

In 1982, the JCS issued "Policy and Procedures for the Management of Joint Command and Control Systems" (reference 2). That memorandum tasked the CINCs to prepare annual C2 System Master Plans as their primary vehicle for identifying deficiencies to the JCS. Although that memorandum is currently under revision, it is still the intent to have the CINCs develop plans that identify the missions supported by C3 systems, the C3 needs and priorities associated with those missions, and the mid- and long-term objectives to satisfy those needs. In addition, the draft memorandum requires the CINCs to contribute to a JCS-sponsored Global C3 Assessment (reference 3), that assesses CINC C3 capabilities and shortfalls from a mission perspective

As a complementary action, the Deputy Secretary of Defense (DepSecDef) sought to enhance the role of the CINCs in the planning process by requesting that they submit annual integrated priority lists (IPLs) (reference 4). The CINCs were directed to define requirements in broad mission and functional areas and to suggest solutions in terms of required platforms, systems, and items. Although this activity transcends the domain of C3, it is notable that C3 issues were generally high on each CINC's list.

(2) Programming

To enhance the CINCs' voice in the allocation of resources, the DepSecDef established an institutional role for them in the review of the Service and Defen. Agency Program Objective Memorandums (POMs) (reference 5). The DepSecDef specified that the U&S CINCs would meet with the Secretary of Defense and the Defense Resources Board (DRB) to present their views on strategy and to discuss the adequacy of the POMs to meet that strategy. To facilitate that dialogue the Services have been directed to develop POM annexes that estimate the costs to satisfy CINC requirements and to identify shortfalls in satisfying those requirements. In addition, CINC representatives are enpowered to serve on POM issue teams.

(3) Budgeting

In the Goldwater-Nichols DoD Reorganization Act of 1986 (reference 6), it directed that effective with the submission of the FY1989 budget, a separate budget proposal be submitted to Congress for activities of each U&S Command as determined by the Secretary of Defense with the advice of the chairman, JCS (CJCS). The separate funding requests may be for joint exercises, force training, contingencies, and selected operations.

b. Requirements Generation

Historically, the CINCs have submitted their requirements in the form of Required Operational Capabilities (ROCs), as dictated by reference 2 There has been widespread dissatisfaction with this process. First, there is concern that the generation and validation of these ROCs is too time consuming. Second, there is a perception that there are too many ROCs, many of which are overlapping. Third, there has been a tendency for the CINCs to submit ROCs that reflect preconceived solutions rather than statements of operation need. In response to these perceived problems, reference 2 is being revised to streamline and rationalize the requirements process

A more revolutionary change in the requirements generation process may be forthcoming if the recommendations of the Packard Commission (reference 7) are more fully implemented Historically, the CINCs have developed ROCs from a "bottoms-up" perspective, drawing extensively upon their experiences in crises and exercises The Packard Commission recommended that a complementary "top-down" perspective be taken in which the U&S CINCs would support the CJCS in appraising the threat, developing military objectives, developing national military strategy, identifying five year force/capabilities within fiscal constraints, developing military options, and preparing net assessments

c Development and Acquisition

In 1980 the CINC Command and Control Initiatives Program (C2IP) was initiated to give the CINCs very limited discretionary resources to acquire critically needed C2 systems, subject to JCS approval. This initiative has given the CINCs a small, but highly leveraged role in systems development and acquisition. As noted by Gen Rogers, former SACEUR, "since the program began in 1980, at least 200 small but critical projects have been funded" (reference 8). Under its aegis, several CINCs have selectively elected to develop unique, one-of-a-kind systems to support pressing needs using off-the-shelf technology. In addition, selected CINCs have instituted system testbeds to provide early fielding of critical capabilities and to enhance the dialogue between the development and operational communities. One such testbed is the Limited Operational Capability Europe (LOCE), which provides an initial capability for managing sensor systems and fusing their outputs.

d. Training

The Goldwater-Nichols DoD Reorganization Act (reference 6) reiterated that the combatant commanders would have full operational command over all assigned forces. However, they redefined the term "full operational command" to include all aspects of military operations and joint training.

3. Perceived CINC Shortfalls

The initiatives cited above have dramatically expanded the responsibilities of the U&S CINCs. There is a perception, however, that the command staffs will experience significant shortfalls in responding to these challenges.

Although the primary focus of this discussion is the J-6 (C3) element of the CINC staff, the full resources available to the CINC must be considered -- most notably, the J-2 (intelligence), J-3 (operations), and J-5 (plans). As a group, this staff has impressive strengths, and significant shortfalls, in responding to its enhanced responsibilities.

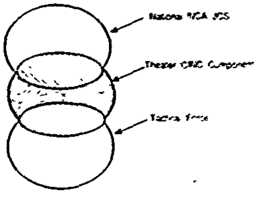
In general, the CINC staffs focus on present-day issues vice those of the future. Their primary attention is generally directed towards pressing problems in training and readiness. Their most notable shortfalls have been in three interrelated areas: they generally lack in-depth knowledge of key factors that will influence the future (e.g., the evolving threat, capabilities and limitations of emerging technologies); they generally lack the skills needed to develop credible long range plans and architectures; and they frequently are limited in their ability to communicate with the technical community. Although these shortfalls could be ameliorated by selectively increasing staff size, just the reverse trend is occurring. There is considerable pressure from OSD and Congress to reduce CINC staffs appreciably.

Another major obstacle that the CINCs face in discharging their enhanced responsibilities is their lack of access to key data. This is particularly apparent in the programming and budgeting review processes where the data base is highly volatile. It is exceedingly difficult for the CINCs staff to formulate or assess issues when they lack timely, detailed programmatic information.

Underlying many of the CINCs' new responsibilities is a need for a broad range of tools. There are four areas where this need is particularly pressing: evaluating C3, formulating requirements, performing programmatic assessments, and supporting realistic joint training. The issues of requirements formulation tends to illustrate this need for tools. In order to do this task credibly, one must be able to consider and balance the operational needs; available resources; future threat, doctrine, and force structure; and evolving technology. Without adequate tools it is exceedingly difficult to ensure that all of these factors are considered consistently.

Finally, the CINCs face varying levels of shortfalls in dealing with C2 system development and acquisition. As noted in section B.1, selected CINCs (notably USCINCSO) lack adequate linkages to the Service System Commands. In addition, nearly all of the CINCs face substantial shortfalls in their ability to integrate new component systems into their evolving C3I system and in their capability to support unique, CINC-developed systems (e.g., training, maintenance). The CINC role in C3 system development and acquisition is further clouded by uncertainty over their domain of responsibility. While they clearly have responsibility for their own organic C3 resources, there is still ambiguity as to their role in the C3 systems associated with the national-theater interface, the theater-tactical interface, and tactical forces (see Figure 2).

Figure 2
Issues: CINCs Area of Responsibility



4. New Skills Needed By CINCs

In order to redress these shortfalls, the CINCs require new skills. This section will be limited to a discussion of those skills that could conceivably be supported by the C2 Research Community The skill needs have been aggregated into the four areas where the CINCs' responsibilities have been increased: planning, programming, and budgeting; requirements generation; C2 system development and acquisition; and training.

a. Planning, Programming, and Budgeting

If the CINCs are to participate effectively in the PPBS process on C3 issues they will require the ability to evaluate the contribution of C3 to mission effectiveness. This need has been recognized in a recent request from the Scientific Adviser to SACEUR to the head of the Defense Science Board (reference 9). As a complement, they will also require the capability to perform cost analyses of C3 systems so that they can perform cost-effectiveness tradeoffs.

In the related area of C3 long range planning and architectural developments, the CINCs require an infusion of new skills. They require organic resources to assist them in developing CINC-unique products and in monitoring compliance with these plans and architectures.

b. Requirements Generation

The CINCs need complementary bottoms-up and top-down tools to assist them in the generation of C3 requirements. Currently, many of their requirements arise from anecdotal experiences that occur during exercises and crises. They need systematic means of generating credible "lessons learned" from those experiences that can be used to justify operational needs. In addition, they require mission-oriented assessment tools to enable them to comply with the spirit of the Packard Commission recommendations for a top-down requirements process.

c. C2 Development and Acquisition

Nearly every recent "blue ribbon" panel study of C2 management (e.g., reference 10) has identified the communications gap between the operational and systems communities as the major obstacle to effective C2 development and acquisition. In order to bridge that gap, the CINCs need the ability to conduct an effective dialogue with the systems community. Testbed environments have been identified as a promising mechanism to support that dialogue. Although this is perceived to be an attractive solution, it must be recognized that it requires the CINCs to gain access to a major new set of skills to implement it (e.g., experts in computer hardware and software, modelers, experimental designers, human factors specialists).

In addition, the CINC faces the problem of assimilating new C3 component capabilities into his evolving system. This raises the need for personnel skilled in all levels of interoperability issues, ranging from knowledge of protocols and standards through changes in procedures and concepts of operation.

d. Training

The CINCs' new responsibilities in joint training pose unique problems in the area of C3. It is particularly difficult to emulate realistically the stresses that joint/combined staffs must confront at increasing levels of conflict. Testbeds have been proposed as a means of coping with this issue, but it poses the need for the broad set of technical skills cited above.

C. C2 Research Community

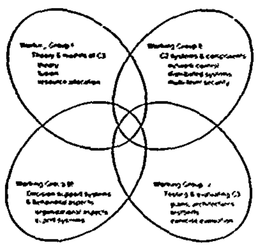
This section deals with the second facet of the issue: the roles that the C2 Research Community could play to respond to the CINCs' needs. As a foundation for this discussion, a taxonomy is introduced to characterize the C2 Research Community and to describe the work that the community is currently performing to support the CINCs. That is followed by a set of challenges that the C2 Research Community should confront.

1. Nature of the C2 Research Community

The C2 Research Community is comprised of highly non-homogeneous, eclectic participants who have defied prior attempts to classify them. However, the structure of the JDL C2 Research Symposium provides a useful taxonomy for discussing the community's interests and describing its activities.

The Symposium has subdivided the community into four overlapping domains (see Figure 3).

Figure 3
Decomposition
of the C2 Research Community



Working Group I is involved in the theory and models of C3. It is striving to establish a theoretical foundation for C3 and is performing applied work in areas such as sensor fusion and resource allocation. Working Group II is focusing on state-of-the-art enhancements to selected C2 systems and components. Major sub-areas of interest include network control, distributed systems, and multi-level security. Working Group III is analyzing behavioral aspects of C3 and developing decision support systems. Their domain of interest subsumes the organizational aspects of C3 and expert systems. Finally, Working Group IV is exploring means of testing and evaluating C3. This includes the development of C3 plans and architecture and the development and application of the full spectrum of evaluation techniques (e.g., analyses, testbeds, exercises).

2. Ongoing C2 Research Support to CINCs

These four segments of the C2 Research Community are currently involved, in varying degree, in supporting the U&S CINCs.

The first group, theory and models of C3, is probably least involved in the direct support of the CINCs. This is not surprising in light of the comments of the Scientific Adviser to SACEUR cited in the Introduction (reference 1). However, some of the group's applied efforts are responding to pressing CINC needs. For example, the community's efforts to develop sensor fusion algorithms have found application in many CINC programs (e.g., LOCE).

The second group, C2 Systems and Components, has been quite active in working with the component commands and, to a lesser degree, with CINC communities. As an example, DARPA has sponsored a number of research programs to explore distributed C3 concepts. In concert with the Army, they embarked upon the Army DARPA Distributed Communications and Processing Experiment (ADDCOMPE) to develop and apply packet switching techniques and distributed data bases to the needs of the tactical community. Although the effort was focused on the XVIII Airborne Corps, initiatives are underway to transfer the resulting technology to USCENTCOM. DARPA is currently in the initial stages of an analogous program in the strategic arena, the Survivable Adaptive Planning Experiment (SAPE). This program will assist CINCSAC and the Joint Strategic Targeting and Planning Staff by exploring more flexible adaptable communications and selected decision aids. The Defense Communications Agency (DCA) is also supporting the CINC community through its recently expanded Command Center Improvement Program (CCIP). The CCIP subsumes both fixed and mobile command centers, associated information systems, the standardization and institutionalization of the modular building concept, and a command center laboratory for rapid prototyping and evaluation The CINCs have benefited substantially from this program, most notably through the MAC model command center and the Proof-of-Concept/Experimental Testbed (POC/ET) for SAC, a derivative of the modular building concept.

The third group, decision support systems and behavioral aspects, has had mixed success in supporting the CINCs. In the area of decision support systems there has been a great deal of activity, frequently in concert with the C2 systems concepts and components group, cited above. However, much of this work tends to be focused on component needs. For example, one of the major successes in ADDCOMPE was the Automated Loading Planning System (ALPS), which enabled planners to configure complex loadings on transport aircraft more rapidly with reduced manpower. As a second example, DARPA is working with the Army to develop prototype corps- and division-level planning aids through the Airland Battle Management (ALBM) program. The subset of this group involved with organizational research appears to be less effectively coupled with the CINC community. There are, however, several notable, promising initiatives in this area: the National Defense University is supporting USCINCSO, by exploring alternative organizational concepts for the command and Net Assessment, OSD, is employing Carnegie Mellon to apply organizational theory to assess elements of USEUCOM's command structure.

The fourth group, testing and evaluating C3, has had selected successes in supporting the CINCs. As noted in section B, the CINCs are responsible for generating C2 System Master Plans and contributing to the Global C3 Assessment. In support of those activities, DCA has developed and refined a mission-oriented approach to C3 planning (reference 11). That approach is being used extensively by the CINCs and is being adapted to meet the planning needs of NATO's Major NATO Commanders (references 12, 13).

In the area of exercise evaluation, two major successes have emerged under the leadership of DCA. In the strategic arena, they manage the POLO HAT exercises under the sponsorship of JCS (reference 14). The purpose of these exercises is to measure the effectiveness of C3 systems that provide two-way connectivity between the National Military Command System and the strategic nuclear forces. Over the years, these efforts have significantly enhanced our udnerstanding of the strategic problem and stimulated major improvements to the system. In the theater/tactical arena, significant progress is being made by DCA in its development and application of the Headquarters Effectiveness Assessment Tool (HEAT) (reference 15). It has been employed in several exercises to quantify the performance of the headquarters staffs

Finally, significant resources have been allocated for developing testbeds to assist the CINCs in training their staffs and evaluating C3 performance. Three of the more notable initiatives have been the Warrior Preparation Center, Ramstein, FRG, the Joint Warfare Center, Hurlbert Field, FL, and the Naval War Gaming System, Naval War College, Newport, RI Significant research is underway to establish the technology to link selected testbeds to permit training and evaluating of distributed C3 concepts (reference 16)

3. Challenges for the C2 Research Community

In many ways, the current level of support that the C2 Research Community provides to the CINCs is quite impressive, even if it is not widely recognized. However, in view of the greatly expanded responsibilities of the CINCs, there are many areas where the C2 Research Community can, and should, focus its energy. The following challenges constitute our perception of the more fruitful avenues for the C2 Research Community to channel that energy.

The sub-community involved in the theory of C3 should strive to develop a foundation upon which the CINCs could base the generation of doctrine, tactics, and plan. Our current state of knowledge in this area can be illustrated by relating it to the field of bridge construction. For thousands of years, mankind built bridges before the theoretical underpinings were available to guide their design and implementation. However, once that theoretical foundation was developed, it enabled us to fabricate consistently more cost-effective and reliable structures. By analogy, we can anticipate far more useful doctrine, tactics, and plans once the theoretical underpinings of C3 have been developed and applied.

The sub-community involved in new C2 systems and components should pursue a set of initiatives to address several CINC-unique and -common needs. Three areas in particular stand out. First, every CINC is confronted with the problem of better perceiving the state and intentions of his adversaries. This requires the development of a set of tools to assist him in the collection process (e.g., tasking available sensors) and the assessment of the collected information. An ancillary requirement in this process is the effective management of the electromagnetic spectrum, since the CINC faces conflicting demands on the use of friendly C3 systems (e.g., sensors, communications), intelligence resources (e.g., SIGINT), and C3CM assets (e.g., jamming of enemy systems). Second, there is a pervasive need for a CINC C3 system that can survive at appropriate levels of conflict There are a spectrum of techniques available to achieve those levels of survivability including hardening, mobility, and distributed communications and data bases, with suitable adaptive management systems. It is necessary to develop and draw upon these techniques to create a customized, survivable capability that is well matched to the unique needs of each CINC Finally, there is heightened awareness of the need to develop trusted computer systems for the CINCs Although this is a formidable undertaking in its own right, it is compounded by the unique environments that confront the individual CINCs For example, in Korea, the systems must not only deal with security issues but must cope with both English and Hongul Conversely, in Europe we face other theater-unique problems Currently, if the FRG provides inputs to selected US systems, the information receives a US classification which may preclude its re-transmission back to the FRG As an initial step in this arena, DCA is developing a security testbed through its CCIP. However, a significant infusion of research activity will be required on security issues if the individual CINC needs are to be understood and addressed effectively.

The sub-community involved in decision support systems and behavioral aspects must ensure that the role of the human in the C3 system is not overlooked by C3 technologists. As noted above, the focus of many of the decision support system applications has been at the component level. In order to make that work useful to the CINCs, this sub-community must seek to understand better the C3 process at the CINC level. Although a significant sub-set of that process may be common for most CINCs, it must be recognized that each CINC operates in very different political/military/economic environments. These differences must be appreciated if future CINC decision support systems are to win acceptance. In the area of organizational theory, many CINCs could profit from an extension of the support that the NDU is providing to USCINCSO. There is a particular need to provide a sounder organizational theory that would enhance our understanding of preferred arrangements for using US forces in joint and combined operations.

The sub-community involved in testing and evaluating C3 has several overt challenges that have recently been issued by the CINCs As noted above (reference 9), the Scientific Advisor to SACEUR requested that the DSB initiate a Task Force ". . to recommend an overall research and test program to develop the analysis, simulations, experiments, and exercises to allow us to evaluate the impact of C3 on combat (military missions)." In addition, SACEUR has recently enlisted the aid of DARPA to deal with a related issue and develop a distributed war-gaming system to support realistic training of his staff.

This sub-community also has a major challenge in assisting the CINC staffs in developing long range C3 plans and architectures Although the mission-oriented approach provides a useful framework for addressing these products, there is still a significant void in user-friendly tools to implement that approach There is a particular need for tools that facilitate the derivation of system deficiencies from an understanding of proposed concepts of operation, assist in formulating cost-constrained packages of C3 systems, and support the assessment of the impact of C3 on mission effectiveness The CINC staffs also require tools that would help them derive lessons learned from exercises and crises

D. Caveats

There are several important obstacles that must be recognized and overcome if the C2 Research Community is to provide more effective support to the CINCs These can be cast in the form of actions that the CINCs must take and potential courses of action that the C2 Research Community must avoid.

As a foundation for these proposed initiatives, the CINCs must enter into the dialogue with the C^2 Research Community by formulating forward thinking doctrine. Since the C^2 Research Community tends to deal with the world of the future, it requires an understanding of doctrine that reflects future concepts of operation. This doctrine should take account of differing objectives on the use of military force and reflect the political/economic/command relationships of CINCs and their forces in different parts of the world.

Conversely, the C^2 Research Community should be extremely sensitive to three key points. First, it must resist the temptation to "push" studies and technologies that aren't well matched to the CINCs' needs and capabilities. If that temptation is not resisted, the result will be user dissatisfaction. Second, the C^2 Research Community must not forget that US forces are "general purpose" and that standardization is important. It should resist the temptation to develop wholly unique products when generalization may be feasible that could expand their utility considerably. Finally, the community must not "end-run" the Service Systems Commands in developing and fielding C^3 products. Without the participation of these commands, the penalty can be unsupportable, non-interoperable products.

This last issue poses a major dilemma for the C^2 Research Community: how can it enlist the support of the Service System Commands without incurring severe time delays and bureaucratic burdens? A potential institutional mechanism to resolve that dilemma has recently been proposed by a DSB Task Force on Technology Base Management (reference 17). They formulated the concept of Advanced Technology Transition Demonstrations (ATTDs). The objective of the ATTD would be a "proof of principle" demonstration in an operational vice laboratory environment. They projected major roles for the user (operator) as program sponsor and the developer (Systems Command) as program manager. Programmatically, they envisioned a typical program of three years at a total cost of \$10-100M in 6.3A funds. To facilitate the transition of technology, they foresaw the need for a transition plan to be in place at the outset of the ATTD. Ultimately, they recommended that half of all 6.3A programs (i.e., approximately \$1B) employ this concept. It might be propitious for the C^2 Research Community to be in the vanguard as this concept is developed and implemented.

E SUMMARY

As documented in this paper, the role of the CINCs has expanded significantly in the areas of the PPBS, the requirements generation process, C^3 development and acquisition, and joint training. However, it is becoming apparent that selected CINCs lack adequate resources to discharge their enhanced responsibilities. Although the C^2 Research Community currently supports several CINCs, that support has been uneven. It has most often been successful when the C^2 Research Community's support is well matched to CINC needs and performed within institutional channels (e.g., performed in concert with a Systems Command).

However, there are several facets of the support that have been disappointing. First, it is perceived that the community has not effectively transferred technology to the CINCs. There is a strong feeling, often justified, that sophisticated commercial-off-the-shelf products are easily available to the civilian sector, while the CINCs are languishing with systems that are technologically obsolete. Second, there is a perception that many of the most exciting fruits of the Services' C^2 research efforts are overly focused on component force issues, to the detriment of the CINCs' needs. Finally, there is a sense that the C^2 Research Community is not adequately attuned to the unique operational context in which the individual CINCs must function. In order to serve that client base effectively, it requires an in-depth understanding of the diverse command issues, force structures, threat environments, and interoperability problems that confront the individual CINCs.

If the C^2 Research Community can be cognizant of these factors and respond accordingly, it has an extraordinary opportunity to support the CINCs The most pressing areas where support is required involve the development of quantitative tools to evaluate the impact of C^3 on mission effectiveness and the development of testbeds to support the training of CINC staffs and to facilitate the dialogue between the operational and technical communities.

It is recommended that the JDL consider the utility of C^2 research activities to CINCs when it is formulating and prioritizing the C^2 research program. This would have two beneficial consequences First, it would enhance the likelihood that the C^2 Research Community's products are transferred and used effectively Second, it would serve to solicit CINC support and advocacy for the C^2 research program In our contemporary stringent budget environment, that support may be critical if the C^2 research program is to survive.

References

1. Correspondence from P. J. Berenson, Scientific Advisor to SACEUR, to R. Hillyer, Technical Director, NOSC, 16 June 1988.

2. "Policy and Procedures for the Management of Joint Command and Control Systems," Joint Chiefs of Staff, JCSM-7-82, 11 January 1982.

3. "Global C^3 Assessment," Joint Chiefs of Staff, June 1986.

4. Deputy Secretary of Defense Memorandum on Revisions to POM Review, Fall 1985.

5. Deputy Secretary of Defense Memorandum on FY88-92 Program Review, Memo DRB 86-4, 24 April 1986.

6. Goldwater-Nichols DoD Reorganization Act of 1986, HR3622, 30 September 1986.

7. "A Quest for Excellence," Blue Ribbon Commission on Defense Management, June 1986.

8. Gen B. Rogors, SACEUR, Testimony Before the Senate Armed Services Committee, 12 March 1986.

9. Correspondence from P. J. Berenson, Scientific Advisor to SACEUR, to R. R. Everett, Chairman, DSB, 4 February 1988.

10. Report of the DSB C^2 Systems Management Task Force, July 1987.

11. D. T. Signori and S. H. Starr, "The Mission Oriented Approach to NATO C^2 Planning," SIGNAL, September 1987, pp 119-128.

12. M. Sovereign and G. Hingorani, "Long Term NATO C^3 Planning," 1988 JDL C^2 Research Symposium, 7-9 June 1988.

13. D. Griswold, "Mission-Oriented C^2 Planning in Allied Command Atlantic/Allied Command Channel," 1988 JDL C^2 Research Symposium, 7-9 June 1988

14. W. Gorham, "Strategic Nuclear C^3 Test and Evaluation," 1988 JDL C^2 Research Symposium, 7-9 June 1988.

15. P. Feld, "Supporting the Evolutionary Acquisition of C^2 Systems," SIGNAL, June 1988, pp 211-217

16. V Monteleon, "Distributed C^3 Testbed and User Interactions," Command, Control and Communications Technology Assessment Conference Proceedings, MTP-87W00001, July 1987, pp II-16 - II-22.

17. Report of the DSB 1987 Summer Study on Technology Base Management, OUSDA, December 1987.

C^3 Systems Research: A Decade of Progress

Harry L. Van Trees

Abstract

In 1978, the Defense Science Board criticized DoD for the lack of a useful conceptual framework for evaluating and specifying C^3 Systems and for the absence of any significant research in the command and control field. They recommended that DoD develop a broad research program encompassing technological, economic, organizational, cognitive and other aspects of command and control system design and performance. Since that time, a number of research initiatives have been started. In 1987, the Defense Science Board conducted another study on command and control systems management. They concluded that, "although the amount of research has grown since the 1978 DSB report, we find that the research is unfocussed and largely technology oriented". Once again, they strongly recommended strengthening the intellectual base for command and control through a coordinated research program in command and control.

In this paper, we will survey some of the progress that has been made in the last decade in C^3 systems research. We will highlight some of the significant results and point out areas where more emphasis is appropriate

Introduction

Good morning I appreciate the opportunity to speak to you on a topic that is of vital importance to our national security. The title of the talk, " C^3 Systems Research: A Decade of Progress", was motivated by the fact that many of the activities at this conference resulted from the 1978 report of the Defense Science Board Task Force on C^3 [1] and the actions that DoD took in response to the report. The objective of the talk is to look at the progress that has been made in the broad area of C^3 systems research since that 1978 report. I accepted the invitation because I knew it would force me to thoroughly review the effort of research community over the last ten years. It has been a challenging effort.

The outline of the talk is shown in Figure 1.

OUTLINE

- INTRODUCTION
- STRUCTURE OF C^2 PROBLEM
- PROGRESS IN MODELING
- PROGRESS IN SIMULATION & TESTBEDS
- ANY FUNDAMENTAL PRINCIPLES?
- THE LAWS OF C^3
- PROBLEMS AND SHORTFALLS
- SUMMARY

Figure 1

First, a brief summary of the DSB recommendations and the history of the problem will be given Then, the structure of the C^3 problem will be discussed. The next two sections review the progress that has been made in various areas. Based on this review, some fundamental principles are suggested. Next, I hypothesize some basic laws of C^3. An assessment of major problems and shortfalls is presented and then, the topic is summarized . I hope all of you will listen to the speech in the context of my being a member of the C^3 research community rather than an outsider.

The starting point for the discussion is the report of the Defense Science Board Task Force on C^3 The committee which was headed by Dr Solomon Buchsbaum and included a number of experienced C^3 experts came up with several recommendations I want to focus on the one shown in Figure 2

DEFENSE SCIENCE BOARD REPORT - 1978

"The DoD should develop a coordinated program of research and testing on command and control concepts, design, and system performance to provide the intellectual base to guide the evolution of improved command and control systems."

Figure 2

Some of the key phrases in that recommendation are "coordinated program","research and testing","concepts, design, and performance,"and "intellectual base". The guidance was broad and the report didn't provide many specific suggestions as to what the committee had in mind. Partially in response to this recommendation, Andy Marshall, Director of Net Assessment in OSD and I (then, Principal Deputy Assistant Secretary (C^3I)) sponsored a Conference on "Quantitative Assessment of the Utility of Command and Control Systems" at the National Defense University in January, 1980 [2,3] The conference was valuable in bringing together diverse interest groups to assess the current state-of-the-art in C^2 utility (or effectiveness) theory. In 1982, Bob Hermann wrote a report reemphasizing the importance of the problem Two results that could be partially attributed to the Hermann report were the creation of the Joint Director of Laboratories C^3 Research Program and the addition of about two million dollars of funding in the C^3 research area. It should be noted that, starting in 1978, ONR had sponsored a significant research program at Massachusetts Institute of Technology and held an annual Command and Control Workshop [4-12]. In 1987, the Basic Research Group of the Joint Directors of Laboratories held the first C^2 Research Symposium at the National Defense University [13,14]. The Symposium superceded the annual MIT/ONR Conference and expanded the participation. This symposium [15] is the second in the series.

In 1987, the Defense Science Board sponsored another study on C^3 [16], again Dr. Buchsbaum was the Chairman. The recommendation that I want to focus on is shown in Figure 3.

DEFENSE SCIENCE BOARD - 1987

"TO STRENGTHEN THE INTELLECTUAL BASE FOR COMMAND AND CONTROL WE RECOMMEND,

- THAT A COMPREHENSIVE PROGRAM DEVOTED TO RESEARCH ON COMMAND AND CONTROL BE DEFINED AND IMPLEMENTED. THE RESEARCH PROGRAM SHOULD DELVE INTO ALL ASPECTS OF COMMAND AND CONTROL, NOT JUST THE TECHNOLOGICAL ASPECTS. IT SHOULD FORM CLOSE LINKAGES TO THE SEVERAL RESEARCH AND GRADUATE EDUCATION PROGRAMS IN COMMAND AND CONTROL IN SERVICE AND DEFENSE EDUCATIONAL INSTITUTIONS AND SHOULD EXPLOIT AND FOSTER RELATED RESEARCH PROGRAMS IN OUR UNIVERSITIES. DARPA SHOULD PLAY A KEY ROLE IN THIS ENDEAVOR."

Figure 3

Once again, there is an emphasis on developing an intellectual base A new element is "close linkages to the several research and graduate education programs in command and control in service and defense educational institutions and should exploit and foster related research programs in our [civilian] universities". I strongly believe that this linkage is key to a successful research program

Clearly, the command and control problem is central to our national security. Some of the reasons for this importance are shown in Figure 4

IMPORTANCE OF THE PROBLEM

- C^3 SYSTEMS ARE CRUCIAL TO FORCE EFFECTIVENESS
- NUMEROUS EXAMPLES OF C^3 SYSTEM FAILURES
- C^3 SYSTEMS ARE EXPENSIVE

Figure 4

Over the years, the demands made on command and control systems have grown exponentially. The increased range, speed, and accuracy of weapons systems have increased the commander's volume of interest significantly and, at the same time, decreased the required reaction time significantly Concurrently, technological developments have provided commanders and their staffs more capabilities to cope with the C^2 problem. It is well-accepted by both the U.S. and its potential adversaries that C^3 systems are crucial to force effectiveness. There have been numerous examples of C^3 problems: e.g., the Pueblo, the Liberty,and NORAD computer failures. Finally, C^3 systems are expensive. For example, the cost of each Milstar satellite on orbit will be one billion dollars.

Everyone believes that C^3 is important. It is less widely accepted that C^3 theory is important. In fact, C^3 theory can have a widespread impact. The range of potential relevance of C^3 theory is shown in Figure 5.

POTENTIAL RELEVANCE OF C^3 THEORY

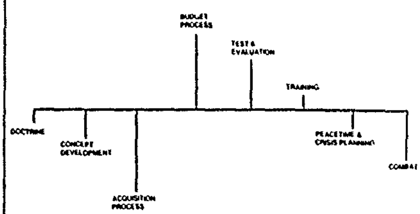


Figure 5

The process begins with the development of doctrine and concepts If the C^2 implications of a proposed doctrine or warfighting concept or weapons systems is considered from the beginning, then more realistic results will be obtained A good example is the Strategic Defense Initiative After the initial announcement of the concept, it became clear quickly that battle management and C^3 would play a crucial role Perhaps if the C^2 implications had been considered earlier, the overall concept would have been different During the acquisition process, the ability to trade-off C^2 system requirements in the context of force effectiveness would be a major contribution. A central feature of the DoD budget process is deciding how to allocate resources among competing demands. For example, should one spend more money on JTIDS (Joint Tactical Information Distribution System), a C^2 system which makes F-15s more effective in accomplishing their mission, or should one buy more F-15s? Quantitative trade-offs on contributions to force effectiveness could make an important contribution. In the test and evaluation phase, C^3 theory could help define appropriate tests and measures. There is a potential for significant savings in the training area if we could develop realistic C^3 simulations. These simulations could augment field exercises and CPX's and make them more effective. The potential saving in training costs easily exceeds the entire C^3 systems research budget. In the peacetime and crisis planning area a better understanding of C^3 can aid the process. Finally, in combat the theoretical results must be applicable in real time. If a theory could generate useful fundamental principles or guidelines it could augment the commander's experience in making decisions. Decision support systems can also play a useful role. This spectrum provides a rich menu of areas to which the research community can contribute.

Figure 5 demonstrates the relevance of C^3 theory to a wide range of problems. From the standpoint of a researcher, it is also important to point that C^3 theory is an intellectually exciting area that is in its infancy. This is in contrast to both communications and control theory which are relatively mature fields (e g., [17]).

With this as background, I set out to review the progress that had been made in the last decade. One of my colleagues remarked that I was spending an inordinate amount of time in preparation: since I had been responsible for C^3I in OSD, I should be familiar with the C^3 research area. The problem is illustrated in Figure 6.

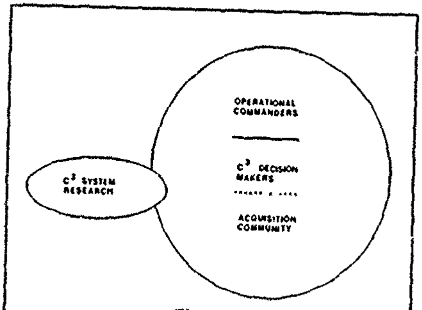


Figure 6

The size of areas could represent either money spent or the number of people involved. Most of decisions made by the acquisition community, operational commanders, or C^3 decision makers is done independently of what is going on in the C^3 systems research community. The basic problem is that the intersection of the sets is almost empty. Very few people are involved in both worlds. In fact, at the present time it includes Gen. Herres, Adm. Tuttle and a few others. This is a fundamental problem we face in the research community. One of the points that I want to make is that, if we are going to be successful, we must get more of the larger community involved in the results of our research.

Structure of the C^2 Problem

A structure for the C^2 problem is shown in Figure 7.

STRUCTURE OF C^2 PROBLEM

- THEORY OF WARFARE
- CLASSICAL DEFINITIONS
- DUAL NATURE OF C^2 SYSTEM
- QUANTITATIVE EVALUATION

Figure 7

The first point I would like to make is that you cannot study C^2 theory in a vacuum. It is possible to study communications theory in an abstract manner and obtain useful results. You can do useful research on coding theory or modulation theory without having to know how a modem or a decoder works. I don't think that is true in the C^2 area. I maintain that everyone involved in C^2 research should start by learning something about warfare. There is not an adequate theory of warfare, but there is a lot of good research and excellent discussions. Some of these are shown in Figure 8.

THEORY OF WARFARE

- CLASSICS
  - SUN TSU, "THE ART OF WAR"
  - CLAUSEWITZ, "ON WAR"
- CONTEMPORARY
  - VAN CREVALD, "COMMAND IN WAR"
- OFFICIAL
  - FM 100-5
  - MARITIME STRATEGY, NWP
  - AFM1-1, TACM 2-1

Figure 8

Sun Tsu is an early and worthwhile reference [13,14]. Writing in the 5th Century BC, he is often referred to as "the father of the theory of strategy." If you adapt his language a little from translated Chinese, you find that many of his ideas are completely relevant to current strategies.

Strategy as we currently know it came into being around 1800 with Napoleon's campaign. Karl von Clausewitz's major work, "On War"[21,22] which was published in 1832 has had a major impact in many nations. Many of his observations such as

"Many intelligence reports in war are contradictory; even more are false, and most are uncertain."

"Everything in war is very simple, but the simplest thing is difficult. The difficulties accumulate and end by producing a kind of friction that is inconceivable unless one has experienced war."

"A critic should never use the results of theory as laws and standards, but only as the soldier does-as aids to judgement."

are as valid today as they were in 1832. Earlier, in 1714, Chevalier Folard had introduced the idea of "the Fog of War" [23,24] to denote the uncertainity inherent in warfare. Fog and friction still remain major challenges to our C^2 systems. An excellent contemporary reference is Van Crevald's "Command in War" [25]. In the last category in Figure 8 are several official documents dealing with doctrine and warfighting [26-29].

As indicated by Figure 9, there is a reasonably logical path leading from Clausewitz to FM 100-5 for the Army case.

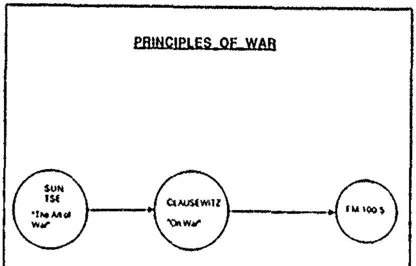


Figure 9

One of the interesting results in FM 100-5 are the Army "Principles in War" shown in Figure 10.

Each of these nine principles has a C^2 implication. In particular, the C^2 system will play an essential role in the unity of command principle. The Army uses these principles as a basis for its doctrine, strategy and tactics. The doctrine, strategy and tactics form the basis of the requirements for C^2 systems.

There are analogous historical bases for the current principles of Air Force and Navy operations. In the Air Force, the heritage leading to the current doctrine comes from Douhet [30], Mitchell [31], and Seversky [32]. (Although some of their theories were not borne out in World War II.) In the Navy, it comes from Mahan [33-35] (Cf [36] for a summary).

The point is that a C^2 researcher must have a reasonable appreciation for the history and principles of warfighting to set the context for his C^2 research efforts.

ARMY PRINCIPLES OF WAR

OBJECTIVE	DIRECT EVERY MILITARY OPERATION TOWARD A CLEARLY DEFINED DECISIVE AND ATTAINABLE OBJECTIVE
OFFENSIVE	SEIZE, RETAIN, AND EXPLOIT THE INITIATIVE
MASS	CONCENTRATE COMBAT POWER AT THE DECISIVE PLACE AND TIME
ECONOMY OF FORCE	ALLOCATE MINIMUM ESSENTIAL COMBAT POWER TO SECONDARY EFFORTS
MANEUVER	PLACE THE ENEMY IN A POSITION OF DISADVANTAGE THROUGH THE FLEXIBLE APPLICATION OF COMBAT POWER
UNITY OF COMMAND	FOR EVERY OBJECTIVE, ENSURE UNITY OF EFFORT UNDER ONE RESPONSIBLE COMMANDER
SECURITY	NEVER PERMIT THE ENEMY TO ACQUIRE AN UNEXPECTED ADVANTAGE
SURPRISE	STRIKE THE ENEMY AT A TIME OR PLACE, OR IN A MANNER, FOR WHICH HE IS UNPREPARED
SIMPLICITY	PREPARE CLEAR, UNCOMPLICATED PLANS AND CLEAR, CONCISE ORDERS TO ENSURE THOROUGH UNDERSTANDING

Figure 10

With this brief discussion of warfighting we can now continue with the command and control problem. Figure 11 repeats the definition of command and control from JCS Pub 1 [37].

COMMAND AND CONTROL (JCS PUB 1)

"THE EXERCISE OF AUTHORITY AND DIRECTION BY A PROPERLY DESIGNATED COMMANDER OVER ASSIGNED FORCES IN THE ACCOMPLISHMENT OF HIS MISSION. COMMAND AND CONTROL FUNCTIONS ARE PERFORMED THROUGH AN ARRANGEMENT OF PERSONNEL, EQUIPMENT, COMMUNICATIONS, FACILITIES, AND PROCEDURES WHICH ARE EMPLOYED BY A COMMANDER IN PLANNING, DIRECTING, COORDINATING AND CONTROLLING FORCES AND OPERATIONS IN THE ACCOMPLISHMENT OF HIS MISSION."

Figure 11

This definition leads us to the two aspects of C^2 systems shown in Figure 12.

The first aspect is that of an integrated system and represents a physical view of the C^2 system. The second aspect focuses on the functional view. What does the system do and how does the process work? It is important that we study the systems from both aspects.

One of essential elements in understanding C^2 systems is the evaluation of their capability. Figure 13 shows the spectrum of techniques that are available for evaluation.

THE TWO ASPECTS OF C^2 SYSTEMS

- INTEGRATED SYSTEM
  - DOCTRINE
  - PROCEDURES
  - ORGANIZATIONAL STRUCTURE
  - PERSONNEL
  - EQUIPMENT
  - FACILITIES
  - COMMUNICATIONS
- C^2 PROCESS
  - WHAT DOES THE SYSTEM DO?

Figure 12

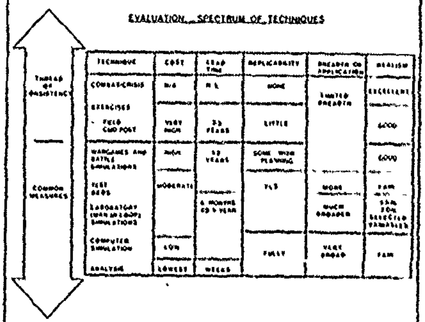


Figure 13

The techniques in order of increasing complexity are:

- expert judgement
- analysis
- computer simulation
- man-in-the-loop simulations
- test beds
- wargames/battle simulations
- exercises
- combat

The figure gives a qualitative assessment of how the various techniques compare in cost, lead time, replicability, breadth of application, and realism. The arrow on the left conveys two important points:

> "A common set of measures that are applicable and measureable across the spectrum of techniques is essential".

> "These measures must give consistent results so there is a *thread of consistency* across the techniques".

One must have a common set of measures that are applicable and measurable across the spectrum of techniques. Coupled with the common measures is the thread of consistency across the techniques. A simple example of this idea is the performance of a digital communication system. An analysis will derive the probability of error as a function of the transmission rate and signal-to-noise ratio. This probability of error is measureable in field tests and combat and, if the analysis was done correctly and the equipment meets its specification, the results will be consistent. In more complex systems, particularly those with humans in the loops, it is generally harder to find consistent measures. As we discuss progress in modeling, I will return to this point repeatedly. It is the responsibility of the research community to demonstrate that the models are consistent with reality.

With these observations about C^3 systems as background, we can begin our discussion of progress in the modeling area.

Progress in Modeling

The outline of this section is shown in Figure 14.

PROGRESS IN MODELING

- ISSUES
- FRAMEWORKS
- PROCESS MODELS
- COMMAND & HEADQUARTERS MODELS
- COMBAT & CONFLICT MODELS
- SURVEILLANCE & FUSION MODELS
- COMMUNICATIONS MODELS
- EW & COUNTER-C^3 MODELS

Figure 14

The first issue concerns the model hierarchy approach shown in Figure 15.

A MODEL HIERARCHY APPROACH

- NO SINGLE MODEL CAPTURES A COMPLETE C^2 ISSUE
- A HIERARCHY OF MODELS MUST BE EMPLOYED
  - MISSION AREAS
  - GEOGRAPHIC SCOPE/LEVEL OF FORCE AGGREGATION
  - LEVEL OF SYSTEM DETAIL
- MUST ENSURE A THREAD OF CONSISTENCY THROUGHOUT THE SET OF MODELS EMPLOYED

Figure 15

No single model is going to capture all of the C^2 issues. A collection of models must be employed The appropriate model will depend on the mission area of interest, as well as other factors The purpose of Figure 16 is to emphasize the disparity between missions.

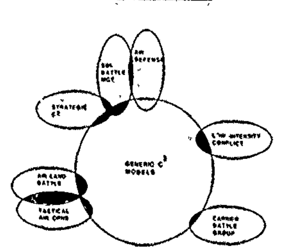


Figure 16

While there is a role for generic C^3 models, many of the important features will be mission specific. The appropriate model also depends on the geographic scope of the model and the level of force aggregation. The choice of a microscopic or macroscopic model will depend on the level of interest. As in any modeling problem, one must match the data available to the degrees of freedom in the model. Once again, it is important to ensure a thread of consistency through the models dealing with a given mission.

Figure 17 is a representative of collection of useful functional models for the C^2 problem

C^3 FUNCTIONAL MODELS

MODELS	TYPICAL ELEMENTS
PROCESS MODELS	SYSTEM DYNAMICS
COMMAND AND HEADQUARTERS MODEL	DECISION MAKING MODEL, STAFF ELEMENTS DATA FUSION
COMBAT AND CONFLICT MODEL	PHYSICAL RED-BLUE ENGAGEMENT MODEL
SURVEILLANCE AND FUSION MODEL	CLASSICAL RADAR, IR, SONAR, PHOTO (BOTH FIXED MOVING), INTERNETTING, PRELIMINARY FUSION (RED, BLUE, ENVIRONMENT) STATUS REPORTING, INTELLIGENCE, I&W, WEATHER FORECASTING
COMMUNICATIONS MODEL	CAPACITY, CONNECTIVITY, S/N, ERROR RATE, SECURITY, AJ CAPABILITY
EW AND COUNTER-C^3 MODEL	DECEPTION, DESTRUCTION, JAMMING EXPLOITATION
INFORMATION AND CONTROL MODEL	TOPOLOGY PROCESSING AT VARIOUS NODES, DATA BASE DISTRIBUTION & MANAGEMENT CONSTRUCTION & DISSEMINATION OF TASKING ORDERS, EAM'S

Figure 17

The models in the first category, process models, focus on the dynamics of the overall C^2 system. Often a process model will have one of the other functional models embedded in it

The functional models correspond to various functions that are carried out in the C^2 system. The taxonomy in Figure 17 is not unique but appears to be a useful approach Some of the functional models will be described subsequently

To understand a C^2 system we must be able to measure its capability. Figure 18 shows the three types of measures that are generally employed.

MEASURES

- MEASURES OF PERFORMANCE (MOP) - MEASURE OF C^2 SYSTEM'S CAPABILITY TO PERFORM ITS OWN "INTERNAL" ACTIVITIES
- MEASURE OF EFFECTIVENESS (MOE) - MEASURE OF C^2 SYSTEM'S CAPABILITY TO ENHANCE THE COMBAT OPERATION IT IS SUPPORTING
- MEASURES OF FORCE EFFECTIVENESS (MOFE) - MEASURE OF HOW WELL THE FORCE AND ITS C2 SYSTEM PERFORM THEIR MISSION

Figure 18

Measures of Performance (MOPs) assess how well a system or subsystem in the overall C^2 system can perform its own internal functions. For example, in the communication system, suitable MOPs include bit error rate, message error rate, end-to-end message delay, antijam margin, and network survivability under various attack scenarios. In a radar system, MOPs include the probability of detection as a function of range, target size, and required probability of false alarm, target resolution capability, and coverage volume. The second level of measures, measures of effectiveness (MOE) assess the capability of the C^2 system to enhance the combat mission it is supporting. An example of an MOE is, a particular communication system (JTIDS: Joint Tactical Information Distribution System) supports the F-15 in its air-air mission. A relevant MOE is the change in kill probability due to the addition of the JTIDS system The final measures are Measures of Force Effectiveness (MOFE) These measure how well the force and the C^2 system perform their mission. A typical example from the strategic area would be the damage inflicted on the target set by the ICBM force The C^2 system would enter into this result through the sensors, the data processing and display, the communication for conferencing, the support to decision-making, and the distribution of the emergency action messages

As one would expect, MOPs are the easiest to define and calculate. In almost all C^2 systems and subsystems MOPs can be obtained. MOEs are somewhat more difficult Sometimes there is an ambiguity about whether a particular measure is an MOP or MOE The resolution is not vital as long as the quantity being measured is clear One of the elements of a comprehensive theory will be an "Effectiveness Theory" MOFEs have been a topic of research for many years (e g ,[38-40]) The impact of the C^2 system adds another degree of complexity to the problem

With these issues and definitions as background, several frameworks for analysis can now be defined as shown in Figure 19.

FRAMEWORKS

- MISSION-ORIENTED ANALYSIS (MOA)
- MODULAR C^2 EVALUATION STRUCTURE (MCES)

Figure 19

The first is the Mission-oriented Analysis (MOA) approach which is depicted graphically in Figure 20.

MISSION ORIENTED ANALYSIS

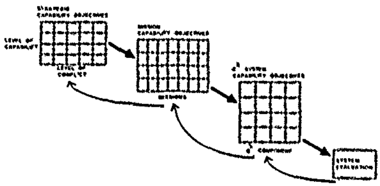


Figure 20

There have been several papers at this conference discussing various aspects of MOA as well as articles (e.g.,[41-44]) so the discussion will be brief. The analysis begins by defining the capability objectives desired for each level of conflict. The levels of conflict are normally defined as peacetime, crisis, conventional war, theater nuclear war, and strategic nuclear war. At each conflict level, various levels of "strategic" capability are defined. The missions and the corresponding mission capability levels necessary to accomplish a particular strategic capability level are then defined. Finally, the C^2 components and the corresponding system capability objectives to accomplish the missions are defined. The C^2 system (either existing, programmed, proposed or hypothesized) is then evaluated and the results are flowed upwards to the original level. Two points concerning MOA are useful:

- The quantitative evaluation at the subsystem level is classical analysis of MOP-type measures. The difficulty arises as one aggregates results to move up the chain
- The dimensionality increases rapidly so the analyst must be skilled in recognizing the important cases

The Defense Communications Agency and the JCS are currently using the technique and have achieved some useful results.

The Modular C^2 Evaluation Structure (MCES) shown in Figure 21 is a second framework.

MODULAR C^2 EVALUATION STRUCTURE (MCES)

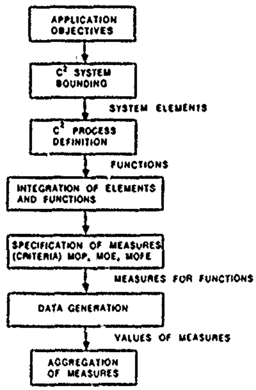


Figure 21

MCES is described in [45] and is discussed and applied in [46-49] MCES can be really considered a check list to guide the analyst in setting up and solving the problem. As in the MOA approach case, the key to success is in the skill of the analyst and the relation of the detailed model to the actual system.

The final technique that should be mentioned is the System Effectiveness Analysis (SEA) technique developed by Bouthonnier and Levis [50-51]. They construct a multidimensional space whose axes correspond to various effectiveness measures. MOPs of the C^2 systems are then mapped into loci in the "effectiveness space." This procedure is really closer to an analysis technique than a framework and could be used with either of the previous frameworks.

The first class of models are process models I will briefly review some of the classical C^2 paradigms and then look at some representative process models The first model of interest is Lawson's classical C^2 model in Figure 22 that was first published in 1978 [52] and has been discussed in subsequent papers [53,54]

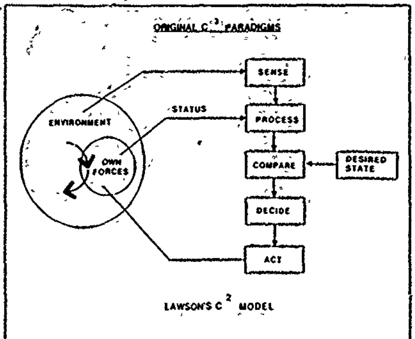


Figure 22

Another classic model, Wohl's SHOR paradigm is shown in Figure 23 [55,56].

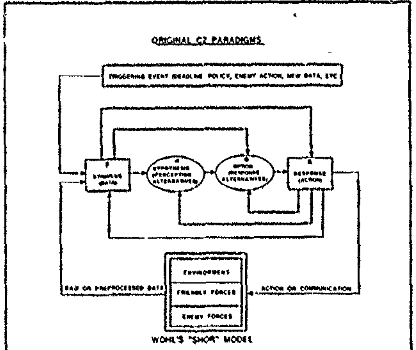


Figure 23

The original work here dealt with Air Force tactical operations Both of these models can be mapped into the model shown in Figure 24 [57].

The first box is the "sense" function. In this function, data is collected by various means to describe the current and projected situation Specifically we are interested in the status of enemy forces and their possible actions, status of friendly forces, and the status of environment (e g., weather, geographical considerations, possible nuclear effects, political constraints, and neutrals) In carrying out this function, we employ a myriad of different sensors, communication systems, and data fusion systems In the second function "Assess", the data from the "Sense" function is used to hypothesize the capabilities and possible intention of the enemy, to review the status and capabilities of friendly forces, and compare the situation to the "desired state" The term "desired state" comes from control theory rather than the usual military lexicon. It can range from a tactical objective such as taking a hill or bridge to a major strategic state. The next step is to generate options or alternative courses of action to get the desired state. The next block includes evaluation of the options and choosing a course of action. In the planning step, the necessary implementation details are developed in order to carry out the selected courses of action. Finally, the "direction" block indicates the dissemination of orders to the appropriate forces. This command model interacts with the environment and the friendly forces in a closed loop manner Over the years a number of alternative paradigms have been developed (e g., Mayk and Rubin[58,59] or Nichols[60]). In general, they include the same functions relabeled or repackaged in a different manner

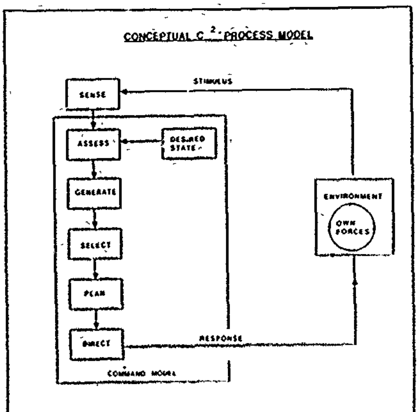


Figure 24

One of the deficiencies in these paradigms is that they do not give adequate emphasis to the enemy. The enemy is not just an element in the environment, he is an adversary who is deliberately trying to achieve his own objectives and deny us our objectives. To emphasize the two sided nature of the problem, I prefer the model shown in Figure 25 which I originally introduced at the Quantitative Evaluation Conference in 1980 [2].

The command section of the previous paradigms are embedded in the boxes, "Blue Command" and "Red Command" The other elements of the C^2 process are identified specifically and can be associated with the model taxonomy in Figure 17. Most importantly, the fundamental two-sided nature of combat is made explicit

In all cases the models must be replicated (or nested) vertically to show the command hierarchy and horizontally to show adjacent units

The significance of the results in the analysis will depend on how well the analyst treats the various elements of the boxes. In a given application, one should choose the model that best illuminates the important features in particular

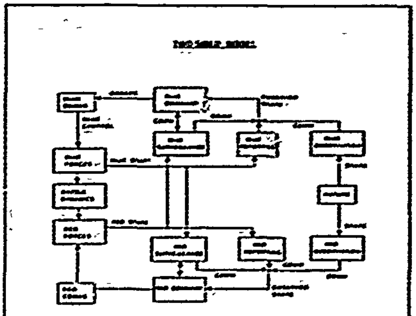


Figure 25

situation at hand. At the current stage of C^2 theory development a standard or generic model is not needed.

The other way that one should view the C^2 system is the classical timeline approach as shown in Figure 26 (from [45]).

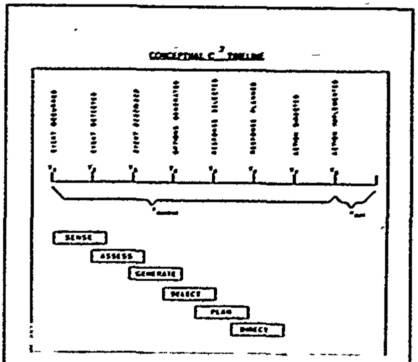


Figure 26

It displays the functions in the model in Figure 21 in a timeline sequence. An interesting application of this approach is shown in Figure 27 (from [61]).

The figure illustrates a possible timeline for a Soviet attack against the United States using ICBMs and SLBMs and the actions required to respond with a "Launch Under Attack" option. Following through the timelines shows that there is a very small "Window of Opportunity" in which the NCA (National Command Authority) must make a decision in order for this option to be successful. Timeline analysis is an important element in most C^2 analyses. There are useful discussions of timeliness in [62] and [63].

Some representative dynamic models are shown in Figure 28.

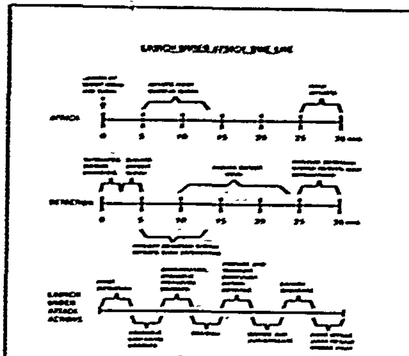


Figure 27

DYNAMIC MODELS

- CLASSICAL STATE VARIABLE MODELS
  - THERMODYNAMIC (LAWSON)
  - MARKOV MODELS (RUBIN, MAYK)
- STATISTICAL MECHANICS MODELS (INGBER)
- POSSIBILISTIC MODELS (ZADEH, GOODMAN)
- CATASTROPHE AND CHAOS (DOCKERY, WOODCOCK)
- ADAPTIVE CONTROL (STRACK)

Figure 28

Most of them have been discussed at one or more of the MIT/ONR or JDL C^3 conferences so I will only comment briefly. Classical state variable models have a natural appeal because of their success in treating analogous problems in control theory. Lawson [53] used a thermodynamic analogy to develop a simplified model.

Rubin and Mayk [64-66] have treated several interesting cases using a stochastic C^3 model which is based a discrete-time, discrete-state Markov process model. They divide the battle into a sequence of stages which are then subdivided into phases. By numerically solving recurrence equations, they obtain probability distributions for various force effectiveness measures.

Ingber [67-69] has used a statistical mechanical approach to derive dynamic models. The central feature of his stochastic model is the path integral Lagrangian. In his paper at this conference [70] he discusses an approach which is consistent with the theme that I will emphasize later with respect to models. He is taking data from the National Training

Center and running it with his model, comparing his model results with outcomes at NTC.

Goodman [71-73] has developed a general model for C^3 processes by viewing them as interacting networks of node complexes of decision-makers. The nodes represent a wide variety of entities, (e.g. troops, tanks, ships, fusion centers, and command centers). The node state and structure, its input signal variables, and its output signal variables (for all the nodes in both the friendly and enemy system) are used to characterize the C^3 processes. In order to characterize the evolution of the state of each of the nodes, he introduces various logical relationships (e.g. classical Boolean logic, probability logic, Zadeh's fuzzy logic, or Dempster-Shaefer belief logic). With these ideas as a basis, he formulates a formal procedure for evaluating the evolution of the system.

Dockery and Woodcock have developed models using catastrophe and chaos theory [74-77]. Chaos theory (e.g., Gleick [78]) deals with physical phenomena in which small changes in initial conditions lead to drastically different system behavior due to nonlinear interactions.

Strack [79] models the C^2 system as an adaptive control system and uses regression analysis to develop C^2 system effectiveness results.

At their present stage of development it is premature to do a comparative analysis of the value of these various models. To make these, or other dynamic models, useful, three steps are important:

1. Model a reasonably realistic scenario,

2. Correlate the results with a simulation, testbed, or actual experience and understand the differences between the results.

3. Explain, in terminology that a CINC or the JCS is familiar with, the lessons learned and the significance of the conclusions.

The last step is particularly important in helping C^2 research in making an impact on the real world.

The first functional area of interest is command theory. Some of the representative elements of a command theory are shown in Figure 29.

ELEMENTS OF COMMAND THEORY

WHAT DECISIONS ARE GOING TO BE MADE AT EACH LEVEL IN THE HIERARCHY?

WHAT CONSTRAINTS ARE THERE ON THE COMMANDERS OPTIONS?

WHO DOES THE DECISION-MAKER HAVE TO INTERACT WITH?

WHAT INFORMATION IS REQUIRED TO MAKE EACH DECISION?

WHAT IS THE DECISIONS TIME LINE?

HOW TIMELY, ACCURATE AND COMPLETE MUST THE INPUT INFORMATION BE?

HOW SHOULD THE INFORMATION BE PRESENTED TO THE COMMANDER?

WHAT DECISION AIDS ARE NEEDED?

HUMAN BEHAVIOR IN A STRESSED ENVIRONMENT

Figure 29

Cushman [80] has identified some of these elements in what he calls "A Commander's Catechism". The first element is basic. It really identifies the entire structure of the C^2 system: who is authorized to make what decisions. The other elements delineate other issues that must be considered in the command function. Once again, the list is representative, not exhaustive.

Representative elements of a headquarters theory are shown in Figure 30.

ELEMENTS OF HEADQUARTERS THEORY

- INFORMATION FLOW PATTERNS
- INTRA-NODAL PROCESSING, COMMUNICATIONS AND DISPLAYS
- PHYSICAL TOPOLOGY OF HEADQUARTERS FUNCTION (DISTRIBUTED, DISPERSED, CENTRALIZED)
- DATA BASE STRUCTURE AND MAINTENANCE
- DECISION AIDS, THE USE OF ARTIFICIAL INTELLIGENCE
- SURVIVABILITY OF HEADQUARTERS FUNCTION

Figure 30

The headquarters function are those processes needed to support the commander(s). Although both the command and headquarters functions have been carried out for several centuries, often with spectacular success, the development of adequate theories is still an area of active research. In many cases the successes appear to have been due to applying Clausewitz's dictum. "Always have a genius in charge, first in general and then at decisive points" [21].

Some representative command and headquarters models are shown in Figure 31.

COMMAND AND HEADQUARTERS MODELS

- HEADQUARTERS EFFECTIVENESS ASSESSMENT TOOL (HEAT)
- DATA FLOW AND DECISION-MAKING STRUCTURES (PETRI NETS)
- MODELS OF DECISION MAKERS
- RESOURCE ALLOCATION

Figure 31

The Headquarters Effectiveness Assessment Tool (HEAT) was originally developed by Hayes and is being evolved further by various researchers (e.g., [81]). The HEAT model is shown in Figure 32.

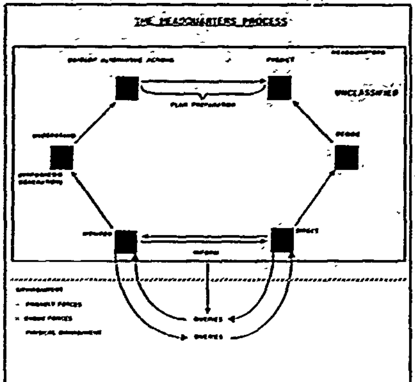


Figure 32

It was originally used for headquarters which were primarily responsible for planning, supporting and coordinating fighting forces (e.g., JCS or Corps level). It treats the C^2 process as an information management system and trys to measure effectiveness in terms of military mission accomplishment. Research is being carried on with generalizations of HEAT.

Some useful work has been accomplished in modeling data flow and decision-making structures using Petri nets. Petri nets which are widely used in computer science (e.g., [82,83]) appear to have first been applied to C^2 problems by Tabak and Levis [84-85]. These papers and subsequent papers [86-90] have used this technique to develop a structured approach to the design of command and control organizations. They first generate a data flow structure using the Petri net formalism. The various types of activities such as data fusion, situation assessment, and response selection can be represented in this formalism. Various MOPs such as accuracy, timeliness, task processing rate, and workload are evaluated. The data flow structures are then transformed into C^2 organizational designs which allocate functions to various decisionmakers. Then, MOEs are evaluated for the various organizations.

A series of researchers [91-93] have developed models of decision-makers. A typical model is shown in Figure 33.

The quantity $\underline{x}'$ represents a set of input symbols from a collection of sources. It is corrupted by noise $\underline{n}$ so that the system receives $\underline{x}$. In the situation assessment stage, one of a set of U algorithms is selected to observe the input and hypothesize the situation. The output is a situation assessment $\underline{z}$. The second stage is a response selection stage, where one of V algorithms is selected in accordance

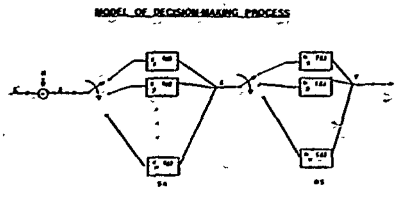


Figure 33

with a response selection strategy. The model is developed using N-dimensional information theory. The total rate of activity in the system is divided into throughput, blockage, coordination, and internally-generated terms. One can then use the idea of bounded rationality to study the maximum rate the system can operate without overload. The simple model in Figure 33 can be extended to include memory and interacting decisionmakers. It is important to note that diagrams such as Figure 33 are not models of commanders like Napolean or Patton,but can play some role in modeling staff and intermediate-level decision makers.

Resource allocation deals with the problem of optimally, or at least efficiently, assigning N_1 weapons to N_2 targets. Often the commanders and their weapons are geographically separated and may have overlapping areas of responsibility. A representative reference is [94].

One of the objectives of the research on command and headquarters theory should be to understand the type of issues highlighted in Figures 29 and 30. Through understanding, we can improve the performance of the commander and his staff. We should note that there is a large body of work on decision support systems which we have not covered.

The next category of models are combat and conflict models (Figure 34).

COMBAT AND CONFLICT MODELS

- LANCHESTER-TYPE EQUATIONS (DETERMINISTIC DIFFERENTIAL EQUATIONS)
- STOCHASTIC COMBAT MODELS (MARKOV PROCESSES)
- GAME THEORY

Figure 34

The earliest work in this area was by Lanchester [38]. A simple form of his equations for aimed fire are shown in Figure 35.

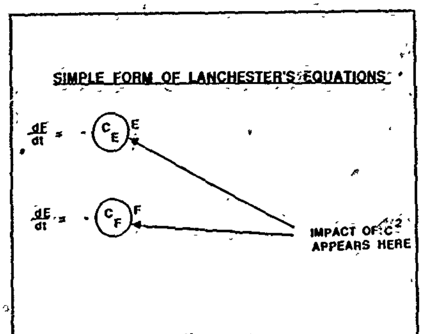


Figure 35

The variables F and E are the size of the friendly and enemy forces respectively and C_E and C_F are the kills/sec/unit of force. In these simple equations, the impact of C^2 appears in these coefficients. For example, improved detection and resolution capability, better target location accuracy would raise C_E. Delay in receiving target location information for moving targets would decrease C_E. There have been numerous generalizations of Lanchester's equations. Taylor [95] provides a comprehensive overview (without any C^2) emphasis. Moose and Wozencraft [96] presented a useful generalization at an earlier symposium. (See [97] also). Shubik's *Mathematics of Conflict* [98] contains several relevant papers. The various models attempt to incorporate the effects of resupply, self-attrition, heterogeneous forces, and information uncertainty. In order to incorporate the effects of maneuver, a set of coupled partial differential equations may be written [99]. Unlike Lanchester's original equations which could be solved explicitly, most of the generalizations require a numerical solution.

An obvious generalization is to incorporate the stochastic nature of the combat process to obtain a Markov process model. Suitable references include [100-103]. The behavior of the mean value of the process relates back to a version of generalized Lanchester's equations. Game theory has also been employed to model combat (e.g., [104-105])

Communications theory is perhaps the most advanced of the various functional areas. Representative elements of communication theory are in Figure 36.

ELEMENTS OF COMMUNICATIONS THEORY

- USER REQUIREMENTS (CONNECTIVITY, QUALITY, CAPACITY, SURVIVABILITY, ENVIRONMENT)
- GENERIC PROPERTIES OF MEDIA AND SYSTEMS
  - CAPACITY
  - QUALITY
  - AJ PERFORMANCE
  - LPI PERFORMANCE
  - SURVIVABILITY
  - RELIABILITY
  - FLEXIBILITY
  - CONNECTIVITY
  - TIME DELAYS

Figure 36

Starting with the user requirements, the general design procedures are well understood and in most cases the generic measures shown in Figure 36 can be evaluated. There are a large number of classic papers and texts that consider these issues(Cf. [106-109]). Communications is an area in which the communications research community has had a major impact on DoD procurement of systems. As examples, the ARPAnet had a significant influence on the current DoD communications networks and Lincoln Laboratories satellite research has impacted the current military satellite architecture.

We should note that the measures in Figure 36 are all MOPs. An important research area is assessing what impact a given communications MOP such as capacity has on mission effectiveness. Considerable work is still needed in that area.

Representative elements of a surveillance and fusion theory are shown in Figure 37.

ELEMENTS OF SURVEILLANCE & FUSION THEORY

- VOLUMES OF RESPONSIBILITY AND INTEREST
- GENERIC PROPERTIES OF SENSORS (E.G., DETECTION, TRACKING, CAPACITY, PROCESSING CAPABILITIES)
- TASKING PROCEDURES (E.G., RESPONSIBILITIES, TIMELINESS)
- INFORMATION FLOW FROM NON-ORGANIC SENSORS
- TOPOLOGY OF SENSOR INFORMATION FLOW AND FUSION

Figure 37

The first idea is that of volumes of responsibility and interest. A simple definition of the Commander's volume of interest is that space around the commander where actions could have a "real-time" or "near-term" impact of his mission accomplishment. This volume has grown continuously over the years as the range and accuracy of weapons systems has increased. Napoleon's volume of interest was probably a circle on the ground with a radius of 100 or so miles. The JCS volume of interest consists of the entire globe including, the ocean depths and space up to at least the geosynchronous satellite orbit altitude. Sensors have continually tried to keep up with this volume of interest in order to reduce our uncertainty about the enemy's intentions. New weapons systems such as steathy cruise missiles cause the race to continue. The volume of responsibility is generally a smaller space and must be clearly defined when the C^3 problem is structured

The generic properties of sensors and their MOPs are reasonably well understood. As in the communications area there are a number of classic papers and texts (Cf. [110-112]) that explain how to design radar and sonar systems and measure their performance. Translating these MOPs to MOEs is still a challenging problem but a number of useful results have been obtained. Tasking procedures for the sensor systems connect back to the command structure problem.

The state of fusion theory is far less mature. There is separate panel under the JDL, the Data Fusion Panel, that deals with this area (Cf. [113]). The fusion area remains a major challenge to the C^3 research community.

The two remaining functional models in our taxonomy are the EW and Counter- C^3I model and the Information model. In the interest of brevity, we will not discuss them in detail. However, they are both significant areas and should not be neglected in the C^3 research program.

The final point to make with respect to the model taxonomy is that there are significant research problems in each of the areas. The C^3 research program should be balanced in its efforts.

Progress in Simulation, Test Beds, and Exercises

The other area of interest is simulations, test beds, and exercises. They deserve equal attention with the modeling area for several reasons:

1. As indicated in Figure 13, they are closer to the real world;
2. They are much more expensive, DoD probably spends 2-3 orders of magnitude more money in this area than in the modeling area
3. They tend to have more impact on operational commanders and decision-makers so improving their fidelity is important.

However, in the interest of time and space, we will simply list some representative facilities for simulation (Figure 38), test beds (Figure 39), and field exercises (Figure 40)

REPRESENTATIVE SIMULATION FACILITIES

- NAVY WAR COLLEGE: WARS, NWGS
- TACTICAL AIR WARFARE CENTER: BLUE FLAG
- ARMY WAR COLLEGE. JOINT THEATER LEVEL SIMULATION
- USAFE, RAMSTEIN. WARRIOR PREPARATION CENTER
- IFFN JOINT TEST AND EVALUATION
- NOSC: IBGTT/RESA
- MAXWELL FIELD. COMMAND READINESS EXERCISE SYSTEM
- JCS- JESS
- JDL - SIMNET
- NPG: WARGAME LAB

Figure 38

REPRESENTATIVE TEST BEDS

- NATIONAL TRAINING CENTER
- RED, GREEN FLAG
- 9TH INFANTRY DIVISION, FT. LEWIS
- ADDCOMPE, FT. BRAGG
- AEGIS TEST BED, MOORESTOWN, NJ
- SDI NATIONAL TEST BED

Figure 39

REPRESENTATIVE FIELD EXERCISES

- GALLANT EAGLE
- NORTHERN WEDDING
- REFORGER
- SOLID SHIELD
- GLOBAL SHIELD

Figure 40

Once again we emphasize the importance of a thread of consistency across the various areas

Fundamental Principles

After completing this review of progress, a logical question is: Are there any fundamental principles? If one is looking for something as fundamental as Shannon's Channel Capacity Theorem [114] or Heisenburg's Uncertainty Principle, the answer is clearly "no". If one is looking for important central results such as optimum Wiener filters, Kalman filters, the fundamental importance of eigenvalues and eigenfunctions, or ambiguity functions, (Cf. [109,110]) the answer is probably still "no". The best I was able to do was to list the ideas shown in Figure 42.

ANY FUNDAMENTAL PRINCIPLES?

- TIME - UNCERTAINTY TRADEOFF
- TWO-SIDED NATURE OF PROBLEM
- SPEED OF DECISION CYCLE
- FORCE MULTIPLIERS CAN START AT ZERO

Figure 42

These notions seem to arise frequently.

Uncertainty is fundamental in warfare. As we get more information (vs data) our uncertainty decreases. As time increases the uncertainty will normally approach some minimum value. Unfortunately, the window of opportunity may close before the uncertainty reaches an acceptable level. Thus the commander and his staff must deal with this time-uncertainty tradeoff. It is important for the staff to apprise the commanders of the specific uncertainties that are inherent in the information that they are presenting to him.

The next principle to remember is that combat, and the C^2 associated with it, is inherently a two-sided problem. At the system design level, we must remember that the communications and radar systems must work in a hostile environment c ted by both the enemy and nature. The ability to function n the presence of jamming, physical attack, EMP, and other threats must be an important design factor. At the decision-making level, we must remember we are dealing with an intelligent adversary and consider his possible reactions to our various strategies. We must also remember that we are fighting an information war and that disrupting, confusing, or destroying his C^3 is a key element in an overall strategy.

The "speed of decision cycle" principle is just a way of describing the fact that we must be able to react faster than the enemy. This principle applies from one-on one air engagements to entire campaigns. History is replete with examples, from Alexander, Caesar, and Napolean to Patton and MacArthur where faster reaction times were important to battlefield success. The Son Tay prison camp raid in 1970 is a good example of where a long decision (and implementation) cycle lead to failure. It took about six months from the time the location was first identified until the raid was conducted and, by that time, the prisoners were gone. Every element of the C^2 system and process must recognize the importance of speed and timely decisions.

The final principle is that force multipliers can start at zero. We often refer to C^2 as a force multiplier. The basic idea is that if we take as a reference point two opposing forces with assumed C^2 systems and a set of assumed strategies then there would be a given battle outcome. This outcome (or MOFE) can be measured in different ways (e.g., surviving forces, territory occupied, objectives obtained). By examining the actual C^2 system a given MOFE will be changed. Thus,

$$MOFE_N = \alpha MOFE_R,$$

where the subscripts N and R denote "new" and "reference", respectively, and α is the C^2 force (effectiveness) multiplier. In many cases one can show that improved C^3 can significantly increase the effectiveness of the force. Unfortunately, there are many examples in which the C^2 system can significantly reduces the effectiveness of the force. The capture of the Pueblo is an example where, by being unable to get available airplanes to the scene in time, the C^2 system reduced the force effectiveness to zero.

There are probably other equally important notions as these four principles. Since I was unable to develop any really fundamental principles, it will be useful to discuss what I will immodestly refer to as "Van Trees Laws of C^3".

Van Trees Laws of C^3

Law 1: Budget

The first law deals with the budget and is motivated by Figure 43.

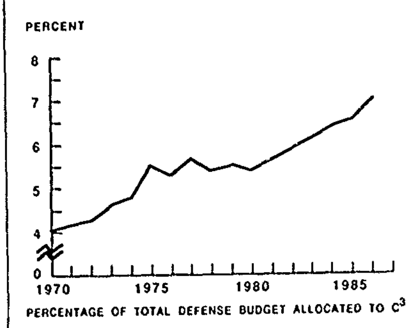


Figure 43

In 1970, C^3 was 4% of the total defense budget. In 1985, this had grown to 7%. Using linear extrapolation we obtain the first law:

"By 2048, the entire defense budget will be allocated to C^3."

The first corollary in this law is that:

"The C^2 force multiplier will have to approach infinity, because there will be no forces left."

It is important to note that the C^3 budget numbers do not include intelligence systems due to their classified nature. If intelligence were included, the result would be even more dramatic.

A second corollary follows if one plots the cost of a communications satellite (on-orbit) starting with the DSCS-I (IDCSP) in the 1960's and concluding with the projected Milstar satellite in the early 1990's. Then compare those costs with the total C^3 budget. Although we do not show the curve, it results in the second corollary:

"By approximately 2032, the entire C^3 budget will be devoted to a single military communications satellite".

The positive side of this corollary is that it should reduce interoperability problems and simplify the C^3 budgeting process.

While the first law states an improbable conclusion, it should demonstrate the importance of understanding the utility and effectiveness of C^3 systems so we can determine how much C^3 is needed.

Law 2: Uncertainty

The second law deals with uncertainty and is illustrated in Figure 44.

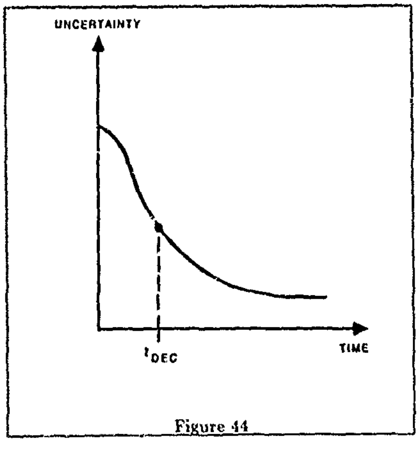


Figure 44

The sketch is a plot of uncertainty using an appropriate measure as a function of time. As more information about the situation is assimilated the uncertainty decreases. Unfortunately, the Second Law says:

"The uncertainty in a C^2 system generally reaches an acceptable level after the required action time is past."

Thus, the commander must work with the uncertain information that is available at the time he must make a decision. The strategic C^2 example in Figure 27 was a dramatic example of this effect. According to that proposed timeline, the NCA must make a decision by about 15 minutes after a Soviet launch if he desires to implement an LUA option. At that time there may be still a significant uncertainty about the situation.

Examples of C^2 decisions throughout history show similar phenomena. Technology is not going to change this basic relationship. For example, as sensor ranges go up, the volume of interest will increase. The successful commander must cope with these uncertainties (the Fog of War) and make appropriate decisions. The role of the C^2 system is to reduce the uncertainties as much as possible and make certain the commander understands them.

Law 3: Communications Capacity

The third law deals with the capacity of the communications systems and is shown in Figure 45.

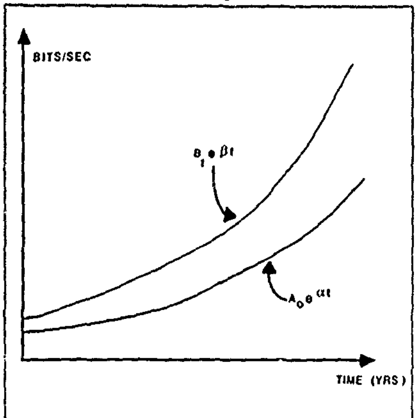


Figure 45

If we plot the communications capacity in bits/sec as function of time, it is easy to show that it has grown exponentially. A good example is the transatlantic military communications capacity in World War I, World War II, 1965, and now. The exact current capacity may be classified, but by simply counting satellites and cables, we can see that it must be in the range of 500 mbps to 1 gbps compared to a miniscule World War I capability Unfortunately, the user demand has grown even more dramatically, leading the third law:

> "The total communication capacity of C^3 systems has grown as $A_0e^{\alpha t}$. The perceived user requirement has grown as $B_1e^{\beta t}$, where $\beta > \alpha$"

In all of the architecture studies, one sees an insatiable demand for more communications. One of the reasons for this relationship is that, unlike the civilian community, the actual military user or the person determining communications requirements, does not get a monthly telephone bill. Thus, the tradeoff between required capacity and cost is less direct. In spite of this, the community must develop methods of determining, "How much is enough" in terms of the contribution of communications capacity to force effectiveness.

Law 4: Processing Capacity

A similar phenomenon for information processing capacity is shown in Figure 46.

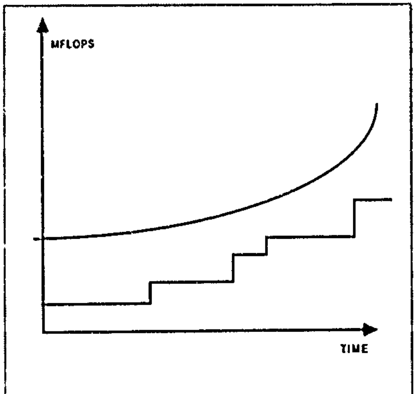


Figure 46

As each new generation of computers has been introduced into command centers the information processing capacity has increased dramatically A curve fitted to the growth would show exponential growth as a function of time However, the Fourth Law states

> "The information processing capacity of computers in command centers has grown exponentially The perceived processing load has grown with a larger exponential "

Examples of this phenomenon abound. For example, in Vietnam, there were 500,000 messages per month coming out the two communication centers in 1966 This corresponded to an enormous information processing load. When you look at the results of JCS exercises or processing loads during crises, you see exactly the same behavior. A significant challenge to the research and operational community is to match the information processing demands to what the commander needs and his staff can absorb.

Law 5: Duality

The next law deals with what I call "Duality". A common approach among mathematicians and researchers is to translate a new problem into one that someone has already solved and then apply the earlier result. One can construct reasonably good analogies between C^2 systems and living (biological) systems. One can also construct analogies between C^2 systems and the management of large companies. Unfortunately, these analogies lead us to Law 5.

> "The C^2 problem is analogous to the living system problem. By mapping one into the other, we can replace one unsolved problem with another unsolved problem."

The corollary 5A is that you can map the C^2 problem into a business management problem that is equally hard to solve We would probably hesitate to use U.S Steel, International Harvester, or Texas Air as models of how to run the military. The point of this law is that analogies are interesting but duality is only useful if it helps obtain quantitative results or aids in intuitive understanding.

Law 6: Acronyms

The next law deals with acronyms and is related to Augustine's 9th Law [115] which is.

> "Acronyms and abbreviations should be used to the maximum extent possible to make trivial ideas profound."

This general law refers to the entire Defense Department However, the C^3 community has developed "Acronymization" to a fine art. The result is Law 6

> "If one takes the 16 most frequently used letters in the English alphabet and creates an ordered triplet randomly from this set, the probability is 0 216 that the result will be a segment of C^3 acronym."

For example, if the triplet comes out SBR, one has the space-based radar, TID gives JTIDS, and so forth Note that by using .216 instead of .25 in the result, we have added credibility to the conclusion

Acronyms are just a visible sign of the difficulty the C^3 community has in communicating with the rest of the world Overcoming this communications problem is an important challenge to the community

Law 7: Complexity

As everyone realizes, C^3 equipment is becoming more complex as each new generation is developed The problem is shown in Figure 47

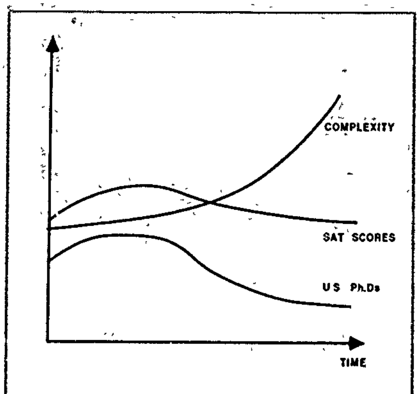


Figure 47

Two metrics that may be relevant in dealing with the population that must develop the basic theory, design the systems and equipment, and operate and maintain the equipment are also shown in the figure. Although the decline in SAT scores appears to have leveled off, the statistics concerning the math and science capabilities of our high school and college graduates is alarming. It should be noted that the Services have done an excellent job in raising the average education level of their recruits. A related and equally disturbing statistics is the decline in Ph.D.'s in Math, Science, and Engineering awarded to U.S. citizens. This trend affects not only our C^3 capability, but the ability of our nation to compete in the world market.

This complexity-ability divergence increases the importance of equipment reliability, built-in test equipment (BITE), and extensive training under realistic conditions. It also emphasizes the importance of reducing equipment complexity by keeping the specifications reasonable. Often the last 0.5dB of performance increases the complexity (and cost) significantly.

Law 8: Models versus Reality

Often C^2 researchers are criticized because their models don't correspond to reality. Actually, the problem may rest with operational commanders as shown in Figure 48.

LAW 8: MODELS VERSUS REALITY

OPERATIONAL COMMANDERS CAN'T SEEM TO UNDERSTAND THAT:

A. TO MOVE A BATTALION 100 MILES IS A SIMPLE STROKE WITH THE "MOUSE"

B. A FLANKING ATTACK IS ACCOMPLISHED BY TRANSPOSING THE TRANSITION MATRIX.

Figure 48

The research community knows that if you portray the Corps sector and the location of its units on a computer screen, then you can move a battalion by simply moving the "mouse." Similarly, if you represent the motion of a particular unit by an appropriate Markov process, then you can conduct a flanking attack by a matrix transposition. Unfortunately, operational commanders feel that the actual operations are not so easy.

This law ties back to one of the important points in the Progress in Modeling discussions. In order for models to be useful; it must correspond a realistic scenario, correlate with simulations or testbeds or actual experience, and be explainable to the operational commander in his terms. The onus for accomplishing these steps rests on the model-maker, not the operational commander.

Law 9: Predicting Future C^3 System Performance

The Ninth Law deals with predicting the performance of C^3 systems as shown in Figure 49.

LAW 9: PREDICTING FUTURE C^3 SYSTEM PERFORMANCE

THE PERFORMANCE OF FUTURE C^3 SYSTEM GENERALLY TURNS "GREEN" AT T_0, WHERE T_0 IS GREATER THAN:

(I) END OF CURRENT ADMINISTRATION PLUS 8 YRS.

(II) PRESENTER'S RETIREMENT AGE PLUS 3 YRS.

Figure 49

Representative mission-oriented analyses typically evaluate the current system, the near term system and a far term system. The normal effect that you get when you use analyses with red, yellow and green as metrics is that, in the current system, there are a fair number of missions that are red and yellow. From this one creates packages or programs to improve it. By the near term you will have a mixture of red and yellows, but with more greens. There is a reasonably general phenomena that almost all missions turn green in the far-term system, particularly if you are willing to supply funds for what the presenter is proposing. The interesting point that this law deals with is the time, T_0, when everything turns green. What the data shows is that T_0 is generally one of two times. It is either at the end of the current administration plus 8 years. in that option it gives the system time to circulate through two new sets of political appointees. Therefore, if it doesn't actually turn out to be green, then the current person has at least one or two generations of preceding people to blame. Alternatively, from the standpoint of the presenter, it is generally his retirement age plus about 3 years.

Unfortunately, past experience indicates the actual performance of future C^3 systems is generally not as "green" as projected in their early stages of planning. Among other factors, Laws 3 and 4 contribute to this problem.

Law 10: Interoperability

One version of this law is:

> "When Clausewitz talked about "friction in war," he was predicting the problems with C^3 in joint operations."

One of the key problems that we have in C^3 systems is interoperability. All this law does is point out that the phenomena was predicted a number of years ago. This is a fundamental problem and there are major efforts underway to correct these interoperability problems. Some of the earlier speakers have given specific examples. It is still one of the most significant problems that we have in the C^3 world and deserves the attention of the research community.

Although there are a number of other laws dealing with such interesting phenomena as sensor range vs volumes of interest, the ratio of signal personnel to the size of the total force, the effects of compartmentation, the characteristics of software, and the size of staffs, it is appropriate to stop at ten. One should also note that a number of Augustine's Laws [115: Laws 5,12,15,17,18,19,37, and 13] can be applied directly to C^3 systems.

While I was presented these laws in a somewhat humorous vein, if you look at the facts embedded in them, there really very fundamental results in the area of C^3. It is important to recognize these phenomena when you are doing your C^3 work.

Problems and Shortfalls

In the final segment of the talk, I would like to discuss some problems and shortfalls as I see them. I have listed seven of them in Figure 50.

PROBLEMS AND SHORTFALLS

- RELATIONSHIP WITH REAL WORLD
- IMPACT ON COMMUNITY, "WHAT'S IT GOOD FOR?"
- BETTER ARTICULATION OF ACCOMPLISHMENTS
- NOT OVER ACADEMIC THRESHOLD
- INCORPORATION OF HUMAN ELEMENT INTO ANALYSES
- INTERACTION WITH CLASSIFIED COMMUNITY
- INTERACTION WITH BROADER OPERATIONS RESEARCH COMMUNITY.

Figure 50

First is the relationship with the real world. We, as a community, have not done an adequate job of relating the research that we are working on to the actual C^3 operational problems that are encountered in practice. It goes back to the diagram in Figure 6. There is a large real world of decision makers and commanders who are dealing with C^3, there is a small research community and there is very little overlap. The onus is on the C^3 research community to create a larger overlap.

The second problem is really related to the first It is important that the research community be able to tell the operational community and the decision-making community what the research is good for: to tell how this research will help them understand C^3 and help them solve their problems.

Third, if you look back at the comments from the DSB Task Force in 1978 and 1987: they are really very similar. The implicit undertone in the 1987 statement is that not really much had been accomplished in a period of 10 years. An important reason for this attitude is that we, as a research community, have not done a good job of articulating our accomplishments. While we still have a long way to go, if you look at the work that is presented here and the work that has been presented in some of the earlier conferences we actually have made a fair amount of progress in understanding parts of the C^3 problem. However, we haven't done a good job of articulating the results and showing what they are is good for.

The fourth point is that C^3 research as a discipline is not yet over the academic threshold. If you look at many of the people in the universities who are working in the C^3 research area; they were already established in their own fields, such as control theory, communication theory, or systems theory before they became involved in C^3 research. It is vital that the C^3 research field have academic respectability if we want to create a body of researchers in various universities. They have to be able to have the same stature as the other disciplines. In order to help accomplish this credibility, we need journals where there are real peer reviews. We need some segment of a professional society that is interested in C^3 research as a major thrust. It is very important that we can show to young professors a career path that can lead to tenure by becoming known as an expert or qualified authority in the C^3 field. This is not uniform situation; there are several universities such as M.I.T., Carnegie Mellon, Connecticut, Naval Postgraduate School, and George Mason where it is recognized as a legitimate and intellectually exciting discipline. This academic involvement is essential to developing a C^3 theory.

The fifth point is that we have not yet done an adequate job of incorporating the human element (e.g., the commander, the staff and the other C^3 personnel) into our analyses. It is vital that we create adequate models or we utilize test beds that give realistic interface with the people involved with the C^3 systems. Otherwise we are not going to have valid results for our system.

The sixth point is that we do not have adequate interaction with the classified C^3 community. I would expect that 70 to 80% of this audience is cleared. There is a fair amount of interesting work that goes on in this community and we have to find a mechanism to move across these boundaries. Perhaps, one approach is to have a classified session as part of this conference.

Finally, I think we need to establish broader interaction with the general Operations Research community. The Operations Research community has at least a fifty year history of doing various kinds of military modeling and analyses to support general military operations.

While they have not emphasized C^3, they have developed a number of interesting techniques, mathematical models, and simulations which we should be aware of. They may be readily adaptable to the problem of interest to us.

Summary

Figure 51 is a brief summary.

SUMMARY

- SIGNIFICANT PROGRESS HAS BEEN MADE SINCE 1978
- THE COMMUNITY MUST DO BETTER AT COMMUNICATING AND INTERFACING WITH THE REAL WORLD
- C^3 SYSTEMS THEORY IS STILL IN ITS EARLY STAGES

Figure 51

I believe that significant progress has been made since 1978.

In spite of the progress (which is partially reflected in the references I have included), we must do a better job in communicating with the real world. Finally, I think C^3 systems theory, however it is defined, is still in its early stages.

Acknowledgements

In the course of doing research for this paper I asked a number of people for inputs. The following were kind enough to respond: S. Andriole, M. Athans, E. Brady, A. Campen, G. Coe, F. Diamond, J. Dockery, I. Goodman, L. Ingber, D. Leedom, A. Levis, M. Muderian, E. Rechtin, R. Robinson, R. Sabat, M. Sovereign, C. Smith, R. Sweet, S. Starr, and J. Wohl, and J. Lawson.

Ms. Ardith Ochs did the manuscript and Ms. Joyce Pattyson did the graphics. The work was done while as employees of M/A-COM Government Systems. The author will join the George Mason University faculty in October 1988 as a Distinguished Professor of Information Technology, Electrical, and Systems Engineering.

References

[1] Office of the Undersecretary of Defense Research and Engineering, "Report of the Defense Science Board Task Force on Command and Control Systems Management", July 1978.

[2] VanTrees, H.L., "Keynote Address" *Proceedings for Quantitative Assessment of the Utility of Command and Control Systems*, National Defense University, Report MTR-80W00025, January 1980.

[3] *Proceedings for Quantitative Assessment of Utility of Command and Control Systems* National Defense University, Washington, D.C., Report MTR-80W00025, January 1980.

[4] Athans, M (Ed), *Proceedings of the First MIT/ESL-ONR Workshop on Naval C^3 Systems*, ESL-R-844-846, M.I.T., Cambridge, MA, Sept. 1978.

[5] Athans, M.(Ed.) *Proceedings of the Second MIT/ONR Workshop on Naval C^3 Systems*, LIDS-R-966-968, M.I.T., Cambridge, MA, March 1980.

[6] Athans, M., Ducot, E., and Tenney, R, (Eds.) *Proceedings of the Third MIT/ONR Workshop on C^3 Problems*, LIDS-R-1020-1023, M.I.T., Cambridge, MA, Sept. 1980.

[7] Athans, M., Ducot, E, and Tenney, R, *Proceedings of the Fourth MIT/ONR Workshop on C^3 Problems*, LIDS-R-1156-1159, M.I.T, Cambridge, MA, October 1981.

[8] Athans, M., Ducot, E, Levis, A, and Tenney, R.(Eds.) *Proceedings of the Fifth MIT/ONR Workshop on C^3 Systems*, LIDS-R-1267, M.I.T., Cambridge, MA., December 1982.

[9] Athans M., Ducot, E., Levis, A., and Tenney, R.(Eds.) *Proceedings of the Sixth MIT/ONR Workshop on C^3 Systems*, LIDS-R-1354, M.I.T., Cambridge, MA., December 1983.

[10] Athans, M. and Levis, A. (Eds.) *Proceedings of the Seventh MIT/ONR Workshop on C^3 Systems*, LIDS-R-1437, M.I.T., Cambridge, MA, December 1984.

[11] Athans, M., and Levis, A. (Eds.) *Proceedings of the Eighth MIT/ONR Workshop on C^3 Systems*, LIDS-R-1519, M.I.T, Cambridge, MA, December 1985.

[12] Athans, M. and Levis, A., (Eds.) *Proceedings of the Ninth MIT/ONR Workshop on C^3 Systems*, LIDS-R-1624, M.I.T, Cambridge, MA, December 1986.

[13] *Proceedings: Command and Control Research Symposium*, JDL Basic Research Group, National Defense University, 1987.

[14] Johnson, S. and A. Levis (Eds), *Science of Command and Control Coping with Uncertainty*, AFCEA International Press, Washington, D.C., 1988.

[15] *Proceedings of the JDL BRG C^2 Symposium*, Monterey, CA, June, 1988.

[16] Office of the UnderSecretary of Defense for Acquisition, "Report of the Defense Science Board Task Force on Command and Control Systems Management", July 1987.

[17] Athans, M., "Command and Control (C^2) Theory: A Challenge to Control Science", IEEE Transactions on Automatic Control, Vol. AC-32, No. 4, April 1987.

[18] Sun Tsu, *The Art of War*, Edited by James Clavell, Delacorte Press, New York, NY, 1983.

[19] Sun Tzu, *The Art of War*, Translated by Samuel B Griffin, Oxford University Press, New York, 1971.

[20] Phillips, T.R, (Ed.), *Roots of Strategy*, Military Service Publishing Co, Harrisburg, PA, 1940.

[21] Howard, M. and Paret, P (Eds), Karl von Clauswitz, *On War*, Princeton University Press, Princeton, NJ, 1976.

[22] von Clausewitz, K, *On War*, Translated by O.S M. Jolles, Infantry Journal Press, Washington, DC, 1950.

[23] Folard, Chevalier *Nouvelles Decouvertes sur la Guerre, 1724*, quoted in Heinl, R.D, Jr.'s, "Dictionary of Military and Naval Quotations", United States Naval Institute, Annapolis, MD, 1966

[24] Newell, C R, "Fog and Friction, Challenges to Command and Control", Military Review, August 1987.

[25] Van Creveld, M., *Command in War*, Harvard University Press, Cambridge, MA, 1985.

[26] U.S. Department of the Army, Field Manual 100-5, *Operations*, Washington, DC, May 1986.

[27] Watkins, J.D., "The Maritime Strategy", Proceedings of the U.S. Naval Institute, Annapolis, MD, January, 1986.

[28] U.S. Department of the Air Force, AFM 1-1, "Functions and Basic Doctrine of the United States Air Force", Washington, DC, February 1979.

[29] U.S. Department of the Air Force, TACM 2-1, "Tactical Air Operations", Washington, DC, April, 1978.

[30] Douhet, G., *The Command of the Air*, translated by Dino Ferrari, Coward-McCann, Inc., New York, NY, 1942.

[31] Mitchell, W., *Skyways-A Book in Modern Aeronautics*, J.B. Lippincott Co., Phila., PA, 1930.

[32] de Seversky, A.P., *Victory Through Air Power*, Simon and Schuster, New York, NY, 1942.

[33] Mahan, A.T., *The Influence of Sea Power Upon History - 1600-1783*, 1890.

[34] Mahan, A.T., *The Influence of Sea Power Upon The French Revolution and Empire, 1793-1812*, 1892.

[35] Mahan, A.T., *Sea Power and its Relation to the War of 1812*, 1905.

[36] Earle, E.M. (Ed.), *Makers of Modern Strategy: Military Thought from Machiavelli to Hitler*, Princeton University Press, Princeton, NJ, 1952.

[37] U.S. Joint Chiefs of Staff, Publications 1, "Dictionary of Military and Associated Terms", Washington, DC, January, 1986.

[38] Lanchester, F.W., "Aircraft in Warfare: The Dawn of the Fourth Arm-No. 5, The Principle of Concentration", *Engineering 98*, 1914.

[39] Morse, P.M. and Kimball, G.E., *Methods of Operations Research*, MIT Press, Cambridge, MA, 1951.

[40] Hughes, W.P., Jr., (Ed.), *Military Modeling*, Military Operations Research Society, Alexandria, VA, 1984.

[41] Signori, D., and Starr, S., "The Mission Oriented Approach to NATO C^2 Planning", SIGNAL, 1987.

[42] Torelli, P., "A Framework for Assessing DoD C^3 Programs", [15].

[43] Griswold, D.B., "Mission Oriented Planning to Support the Maritime Major NATO Communities", [15].

[44] Sovereign, M., and Hingorani, G., "Long-Term NATO C^3 Planning", [15]

[45] Sweet, R., Mensh, D., Gandee, P., Stone, I., and Briggs, K., "The Modular Command and Control Evaluation Structure (MCES) Applications of and Expansion to C^3 Architectural Evaluation", Naval Postgraduate School, Monterey, CA, September, 1986.

[46] Sweet, R., "The MCES and the Search for Generic Measures", [13].

[47] Sweet, R., and Sovereign, M., "Evaluating Command and Control: A Modular Structure", C^3I Handbook, Edition Three, Defense Electronics, 1988

[48] Sweet, R., and Levis, A., "The Modular Command and Control Evaluation Structure (MCES) and Command Center Evaluations: Case Study: Proof-of-Concept/Experimental Testbed (POC/ET)", 1987 IEEE Military Communications Conference, Washington, DC, October 1987.

[49] Gandee, P.L., Gray, M.D., and Sweet, R., "Evaluating Alternative Air Defense Architectures", SIGNAL, January, 1987.

[50] Bouthonnier, V. and Levis, A.H., "Effectiveness Analysis of C^2 Systems", IEEE Trans. on Systems, Man and Cybernetics, Vol SMC-14, No. 1, January-February, 1984.

[51] Levis, A.H., "System Effectiveness Analysis", Laboratory for Information and Decisions Systems, M.I.T., Cambridge, MA, 1987.

[52] Lawson, J.S., "A Unified Theory of Command and Control", 41st Military Operations Research Symposium, 11-13 July, 1978

[53] Lawson, J.S., "The State Variables of a Command Control System", [3].

[54] Lawson, J.W., "The State of Variables of a Command Control System", *Analytical Concepts in Command and Control*, MORS Monograph, 1981.

[55] Wohl, J.G., "Force Management Decision Requirements for Air Force Tactical Command and Control", IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 9, Sept. 1981.

[56] Wohl, J.G., Gootkind, D. and D'Angelo, H., "Measures of Merit for Command and Control", MITRE Report MTR-8217, Bedford, MA, 15 January, 1981.

[57] Sweet, R., Metersky, M., and Sovereign, M., "Command and Control Evaluation Workshop", Military Operations Research Society, Naval Postgraduate School, Monterey, CA, 1985.

[58] Mayk, I., and Rubin, I., "Paradigms for Understanding C^3, Anyone?", [13].

[59] Mayk, I., Rosenstark, S., and Frank, J., "Analysis of C^3 Systems Based on a Proposed Canonical Reference Model", [12].

[60] Smith, Charles L. "Survey and Assessment of the Role and Use of Modeling in the Analysis of Command and Control Systems", [13].

[61] Carter, A.B., Steinbruner, J.D., and Zraket, C.A., (Eds.) *Managing Nuclear Operations*, Brookings Institute, Washington, DC, 1987.

[62] Cothier, H. P., and Levis, A.H., "Assessment of Timeliness in Command and Control", [11].

[63] Lawson, J.S., "The Role of Time in a Command Control System", [7].

[64] Rubin, I., Baker, J., and Mayk, I., "Stochastic C^3 Modeling and Analysis for Multi-Phase Battle Systems", [13].

[65] Rubin, I., and Mayk, I., "Dynamic Stochastic C^3 Models and Their Performance Evaluation", [12].

[66] Rubin, I., and Mayk, I., "Modeling and Analysis of Markovian Multi-force C^3 Processes", [12].

[67] Ingber, L., "Nonlinear Nonequilibrium Statistical Mechanics Approach to C^3 Systems", [12]

[68] Ingber, L., and Upton, S., "Stochastic Model of Combat", [13]

[69] Ingber, L., "Applications of Biological Intelligence to Command, Control and Communications", in *Computer Simulation in Brain Science: Proceedings, University of Copenhagen*, 1986

[70] Ingber, L., "Mathematical Comparison of Computer-Models to Exercise Data: Comparison of JANUS(T) to National Training Center Data", [15].

[71] Goodman, I.R., "A Probabilistic/Possibilistic Approach to Modeling C^3 Systems", [12].

[72] Goodman, I.R., "A Probabilistic/Possibilistic Approach to Modeling C^3 Systems"; Part II, [13]

[73] Goodman, I R, "Toward General Theory of C^3 Processes", [15].

[74] Coe, G Q., and Dockery, J.T, "OJCS Initiatives in C^2 Analysis and Simulation", *Science of Command and Control*, [14].

[75] Dockery, J.T., "A Fuzzy Set Treatment of Effectiveness Measures", TM 729, SHAPE Technical Center, The Hague, Netherlands, January, 1984.

[76] Woodcock, A.E R., and Dockery, J.T., "Application of Catastrophe Theory to the Analysis of Military Behavior", SHAPE Technical Center, STC, CR-56, The Hague, Netherlands, 1984.

[77] Woodcock, A.E.R., and Dockery, J.T., "Mathematical Catastrophes and Military Victories", SIGNAL, April, 1987.

[78] Gleick, J., *Chaos, Making a New Science*, Viking, Penguin, Inc., New York, NY, 1987.

[79] Strack, C., *Elements of C^3 Theory*, draft report for Defense Communications Agency, 30 January 1985.

[80] Cushman, J.H., "A Field Commander's Views on Measures of Merit", MITRE Corp, Bedford, MA, 4 December 1978.

[81] Serfaty, D., Athans, M., and Tenney, R., "Towards a Theory of Headquarters Effectiveness", [15].

[82] Reisig, W, *Petri Nets. An Introduction*, Springer Verlag, New York, 1982.

[83] Peterson, J.L., *Petri Net Theory and the Modeling of Systems*, Prentice-Hall, Englewood Cliffs, NJ, 1981.

[84] Levis, A.H., and Tabak, D, "Petri Net Representation of Decision Models", [10].

[85] Tabak, D. and Levis, A.H., "Petri Net Representation of Decision Models", IEEE Transaction on Systems, Man and Cybernetics, Vol. 5, SMC-15, No. 6.

[86] Levis, A H., [illegible] Hillion, H.P, "Performance Evaluation of Decisionmaking Organizations", [13].

[87] Levis, A H, and Andreadakis, S.K., "Design Methodology for Command and Control Organizations", [13].

[88] MacMillan, J., and Wohl, J.G., "Human Decisions in an SDI Command and Control System: A Petri Net Model", [15].

[89] Madsen, K.R., "Modeling and Simulation of Army Division Headquarters Using Petri Nets", [15].

[90] Levis, A.H and Boettcher, K.L, "Decisionmaking Organizations with Acyclical Information Structures," [8]

[91] Boettcher, K.L and Levis, A.H., "Modeling the Interacting Decisionmaker with Bounded Rationality", IEEE Transaction Systems, Man and Cybernetics, May/June 1982.

[92] Hall, S.A. and Levis, A H. "Information Theoretic Models of Memory in Human Decisionmaking Models", [9].

[93] Levis, A H; "Information Processing and Decisionmaking Organizations: A Mathematical Description", [9]

[94] Kahn, C, Kleinman, D., and Serfaty, D, "Distributed Resource Allocation in a Team", [13].

[95] Taylor, J G, *Lanchester Models of Warfare*, Vols I and II, ORSA, Arlington, VA, 1983.

[96] Wozencraft, J M., and Moose, P.H, "Lanchester's Equations and Game Theory", [9].

[97] Wozencraft, J.M, and Moose, P.H, "Characteristic Trajectories of Generalized Lanchester Equations", NPS 62-87-014, U S. Naval Postgraduate School, Monterey, CA, June 1987.

[98] Shubik, M., (Ed.), *Mathematics of Conflict*, Elsevier Science Publishers, New York, NY, 1983.

[99] Protopopescu, V., et al, *Combat Modeling with Partial Differential Equations*, Oak Ridge National Laboratory Report ORNL/TM 10636, Nov. 1987.

[100] Karr, A F., "Lanchester Attrition Processes and Theater-Level Combat Models", in *Mathematics of Conflict*, Elsevier Science Publishers, New York, NY, 1983.

[101] Tavantzis, J., Rosenstark, S., and Frank, J. "A Stochastic Model of Lanchester's Equations", [11].

[102] Shultz, G. W., and Tsokos, C.P., "Stochastic Modeling- Lanchester's First Law", in Tsokos, C.P., and Thrall, R.M., "Decision Information", Academic Press, New York, NY, 1979.

[103] Dockery, J.T., and Santoro, R.T., "Lanchester Revisited. Progress in Modeling C^2 in Combat", SIGNAL, July 1988.

[104] Isaacs, R., *Differential Games*, John Wiley, New York, 1965.

[105] Shubik, M, (Ed.) "Game Theory. The Language of Strategy?" in *Mathematics of Conflict*, Elsevier Science Publishers, New York, NY, 1983.

[106] Davenport, W.B., and Root, W.L, *Random Signals and Noise*, McGraw-Hill, New York, NY 1958.

[107] Gallager, R.G., *Information Theory and Reliable Communication*, John Wiley and Sons, New York, NY, 1968.

[108] Wozencraft, J.M., and Jacobs, I.M., *Principles of Communication Engineering*, J. W. Wiley, New York, NY, 1965.

[109] Van Trees, H.L., *Detection, Estimation and Modulation Theory, Vol. I*, J. W. Wiley, New York, New York, 1968.

[110] Van Trees, H.L., *Detection, Estimation and Modulation Theory, Vol. III*, J. W. Wiley, New York, New York 1971.

[111] Helstrom, C.W, *Statistical Theory of Signal Detection*, Pergamon, London, 1960.

[112] Berkowitz, R.S, *Modern Radar. Analysis, Evaluation and System Design*, J.W. Wiley, New York, New York, 1965.

[113] Proceedings of the Tri-Service Data Fusion Symposium, Vol I, John Hopkins University Applied Physics Laboratory, Laurel, MD, 9-11 June 1987.

[114] Shannon, C.E and Weaver, W, *The Mathematical Theory of Communication*, The University of Illinois Press, Urbana, Ill, 1949.

[115] Augustine, N.R, *Augustine's Laws* Viking, New York, NY, 1986

Section II

Theory and Models of C^3

Working Group Chairman:

Dr. Richard P. Wishner
Advanced Decision Systems

The C^3 Reference Model (C^3RM), Recent Developments

by

Israel Mayk# and Izhak Rubin*

#US Army CECOM, Center for C3 Systems, Fort Monmouth, NJ

*IRI Corporation, Tarzana, CA and UCLA, Los Angeles, CA

ABSTRACT

The subject reference model [1] describes a framework for the evolution of a coordinated and detailed definition of a C^3 discipline addressing complete C^3 systems, their resources and inherent interactions. It includes generic and analog extensions to the International Standards Organization (ISO), Open System Interconnection (OSI) Reference Model (RM) [2, 3]. As such, it requires reinterpretations and generalizations which go far beyond the scope of the ISO OSI RM. Any constraint or feature which may be present within the ISO OSI RM should not in any way be assumed to be automatically present, relevant and applicable throughout the C^3RM. The C^3RM includes the ISO OSI RM by adopting it for the communications types of interactions. In parallel, layers of three other complementary types of interactions also provide services to the application layer. The C^3RM embraces anologous architectures for all the key types of physical interactions and utilizes the application layer to provide command and control over all types of interactions in an integrated fashion. The goal of C^3RM is also to provide a framework to guide the development of a coherent set of standards, specifications and implementations of interoperability and to protect extensive investments in modular reuseable technologies.

BACKGROUND

The development of the C^3RM has been motivated by the fact that a multitude of paradigms for C^3 have been and are continuing to be developed in an *ad hoc* fashion. Such a proliferation in the long run may have an adverse effect of stifling the research and development of C^3 as a discipline by increasing the state of confusion about C^3 rather than enriching it as the case might be with a variety of well coordinated complementary perspectives. The C^3 is often hidden in war stories, doctrine and even technological products which compete for their opportunity to be fielded. In effect, however, C^3 may be found individually in any resource involved in a conflict and collectively in all resources teamed as a coherent fighting force. C^3 may have infinitely many dimensions corresponding to infinitely many perspectives. As a result, C^3 research and development efforts need to be multidisciplinary and require a significantly high level of collaboration to reach a common understanding. Facing this challenge, members of the Joint Directors of Laboratories (JDL) Technical Panel for C^3 (TPC^3) through the C^3 Research and Technology (C^3R&T) Program [4] are chartered to sponsor the development of the C^3RM. As a result of deliberations by the Basic Research Group(BRG) of JDL TPC^3 C^3 Research and Technology Program, the C^3RM Subgroup was formed to further develop and provide a focal point for the C^3RM as proposed in reference [5]. The progress made to date is the result of coordination with and feedback from many individuals, most of whom are identified in the acknowledgement section of this paper. It was coincidental and perhaps an indication that the time was right for the emergence of the C^3RM, that a similar structural framework was proposed at the same time, at the same conference but in a different working group session [6]. It is the diversity of perspectives on C^3 which beckons a common reference model and associated terminology to facilitate the evolution toward a consolidated framework for understanding C^3. The primitive notions, definitions and the entire structure of the C^3RM adopted to date are subject to change, pending future developments as required to achieve a greater common understanding, to be coordinated by the BRG C^3RM Subgroup with the support of the C^3 community at-large.

APPROACH

Regardless of the stage of technological development, systems comprising of interacting personnel and materiel utilize fundamental building blocks. These blocks may be subjects or objects which also may be comprised of interacting materiel and personnel. They may be functional or physical. Functional descriptions may be incorporated into physical realizations and conversely, physical descripions may be reducible to functional modules. The characteristic of the blocks, the possible local and global associations within and among the blocks and their means of interactions constitute the C^3 architecture and imply the potentials of all derivative systems. The initial stage of developing the C^3RM is directed towards conceptual explications and elucidation of generic and technology independent principles. Conceptual developments will be aided by and stimulate the further development of common formal descriptive techniques. The approach is comprehensive in the sense that it allows for exploiting a wide spectrum of theories which may be based upon the phenomenolgy of macroscopic behavior as well as basic principles of meta-, meso- and microscopic nature. The scope of the general architectural principles required for layering and sublayering the blocks is very broad and should lead to a universal framework for the stepwise unfolding and progressive adaptation and integration of capabilities. It is left up to each C^3 organization to study the C^3RM, extend it, if needed, and apply it to itself, its subordinates, its superiors and other C^3 organizations of importance to motivate and obtain greater insight where needed.

SCOPE

The C^3RM describes a framework for modularizing interoperability among resources which must be networked to comprise C^3 systems. As such, it establishes a framework for coordinating the development of existing and future standards as well as specifications for the interoperability of resources and is provided for reference by those standards and specifications. The C^3RM does not specify actual services and protocols for interoperability. It is neither an implementation specification for C^3 systems, nor the basis for appraising the conformance of implementations. It is intended only to provide uniformity of guidance with respect to the need of a coherent set of complementary standards for C^3 systems. The scope of the architectural principles required for modularizing interoperability, is applied in the broadest sense possible to embrace all key physical and logical interactions required for resources of C^3 systems. It is concerned with physical interactions involving not only communications, but transportations, identifications, and inflictions, as defined herein, which may take place between resources of the same, friendly, adversarial or neutral C^3 systems. It is equally concerned with the logical interactions which result in and from the physical interactions. The reference to the ISO OSI RM for communications does not by itself constitute an endorsement of the ISO OSI standards for any C^3 system.

INTRODUCTION

Command is typically associated with exercising "authority" where as control is associated with exercising "direction" over assigned forces in the accomplishment of missions. The definitions for "Command and Control (C^2)" usually imply the combination of command and control in an additive fashion. Definitions of "Command and Control System", however, typically include the facilities, equipment, communications, procedures and personnel essential to the commander for planning, and controlling operations of assigned forces pursuant to the missions assigned. Note that the force and the commander are generally excluded from being a part of the "C^2 system", yet both the commander and the C^2 system are included in the Table of Organization and Equipments (TO&E) of force units. The above definitions are insufficient to resolve among others the following key potential ambiguities: Is the commander a part of the "C^2 System"?, If not, what is the name of the system that includes the commander? What is the name of the system that in an integrated fashion includes not only the commander but his forces as well? What is the difference between "C^2" and "C^3" and between "C^2 System" and "C^3 System"? To overcome this problem of divergent interpretations, organizations charged with the mission to advance the state-of-the-art in different areas related to C^3 must collaborate to establish and support the evolution of a common framework for the C^2 discipline, via a C^3 *reference model*. The C^3RM incorporates certain primitive notions and definitions as a basis for its description of C^3 architectures. The notions defined are established as part of the C^2RM and technically should not be confused with the notions conveyed by the same words which may be found in other documents of other disciplines or as a part of lay language.

Three generalized canonical dimensions of C^3 architectures form the basis for constucting C^3 systems as represented by the various C^3 paradigms. These three dimensions involve a) the Resources X_i, b) the interactions Y_i, and c) the conflicts Z_i. No description of C^3 is complete without interrelating elements from the aforementioned canonical dimensions. The physical dimensions of space and time are implicit within each of the generalized dimensions. The resources X_i are layered to provide services. The interactions Y_i are layered in a corresponding fashion to enable interrelated identifications, communications, transportations, and inflictions. Finally, the conflicts C_i are also layered to allow the transition from *global* warfare to individual use of armaments. The hierarchical and layered nature of C^3 suggests a similar structure for describing C^3 countermeasures and C^3 counter countermeasures.

Note that the environment is responsible for triggering observations. A subsequent action may then impact the environment. Define a *system* F as a collection of *resources* X^F_i, *interactions* Y^F_i and their underlying dynamics driven by as well as generating new *conflicts* C^F_i. A primitive resource is a collection of inter-related *ingredient items* which *form* a *physical subject* able to *perform* a useful set of *processes*, $W_j(X^F_i)$. Physical and logical processes generate and receive physical and logical *objects* in the course of interaction with other resources. A logical object is a block of data, providing information or knowledge about physical objects. The key ingredient items are the physical as well as the logical *assets* which are crucial in characterizing the performance of the resource. The physical assets are capable of expending the potential energy available to a process when instructed to generate the kind of results desired by the resource. Such results, as a minimum, express any redistribution of physical assets as well as any other pertinent state variable as a function of time. In particular, the state of preparedness and readiness of the distributed assets is of crucial importance. Note that, ultimately, all processes are embodied by the physical assets.

Individual resources may be linked and networked to create large-scale, compound, collective, aggregated or functional processes spanning many resources. Inversely, processes from different resources may be linked and networked to create large-scale, compound, collective, aggregated or functional resources. This reality dictates a combined layered resource/ C^3 process architecture which should prove useful in development as well as in research. The process aspects of the model allow for conceptual visualization whereas the resource aspects of the model allow for design and implementation of the C^3 system.

The C^3 process aspects of the C^3RM have been expanded a shown in Figure 1. Each C^3 system must continuously go through the main cycle as shown. It is a multi-sided paradigm. In Figure 1, however, only one side is shown to interact through the environment. In general, as shown in Figure 1a, resources are involved in performing three types of *fundamental* processes *observation*, *decision* and *action*. A *subsystem* is a subset of system resources including their imbedded dynamics. The observation subsystem is the collection of all resources which are involved in making observations with regard to other resources and the environment. The action subsystem is the collection of all resources which are involved in executing actions which may impact other resources and the environment. The decision subsystem is the collection of all resources involved in making decisions, i. e., those resources responsible for deciding about conflicts with other resources and how to best utilize the observation and action subsystems to accomplish the missions evolving from the conflict. These subsystems may then be decomposed into specialized subsystems for different phases of observation, decision and action which may occur sequentially or in parallel as shown in Figures 1b and 1c. The complete paradigm may be imbedded, nested and/or layered to convey multiple levels of analogous processes as shown in Figure 1d. Note that in nested architectures, lower level resources become assets of higher level resources. As shown in Figure1b it is possible to add feed-forward connections which use the latest observations for adjustment of decisions to provide *a priori* predictive actions and thereby speed up the main C^3 cycles. As shown in Figure 1c, it is also possible and often desired or required to incorporate feed backward connections for fast corrective actions which use previous decisions as desired states

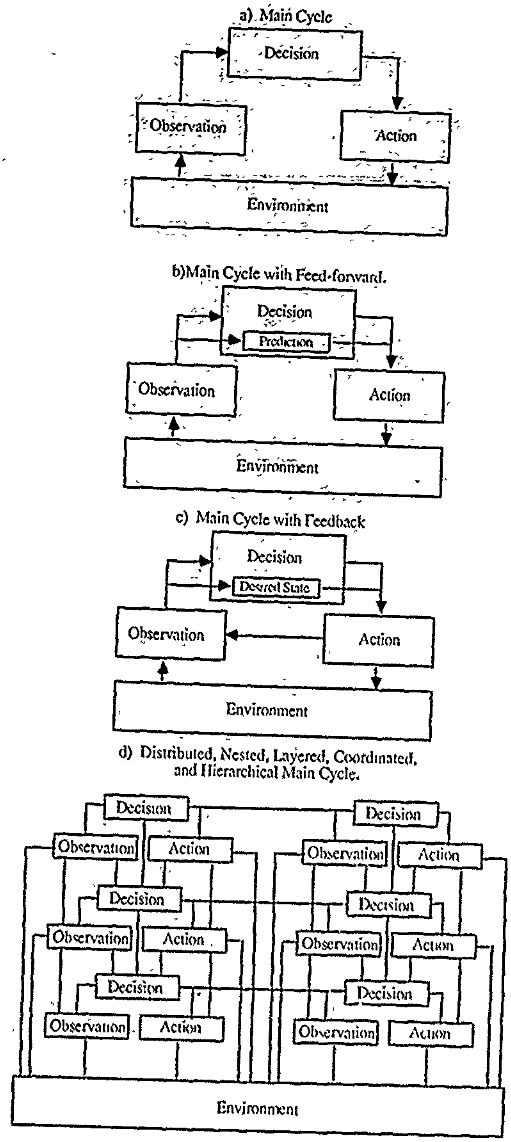


Figure 1. The General Structure of C^3 Paradigms.

to react *a posteriori* to unpredictable changes in the environment, and other systems.

The environment E is depicted as a separate factor. Given the universe as the totality of all physical objects, potentially adverserial C^3 systems, F,G,H,... and the environment E are disjoint sets of physical objects which together form the universe. Note that independent C^3 systems structure and dynamics are generically identical. In addition, physical elements depicting land, air, sea, space and weather are considered as part of the environment. Since E also includes *environmental assets* (natural resources) which C^3 systems require for habitat and consumption, a primary concern for any C^3 system is its ability to survive and thrive in peace within E. *Peace* is the state of a space-time region in E which is void of conflicts between and among co-existing C^3 systems. Competition for natural resources, however, as well as other apparently rational or irrational C^3 system requirements may lead to the development of space-time regions of conflict. As shown in Figure 2, each space-time region of conflict must be supported by a number of varied resources or processes which depends upon the scale of the conflict. The highest region of *conflict* is a space-time region characterized by the state of *war* nested in the region of *peace*. In addition to the region of war, the region of conflict consists of a set of five other hierachically nested regions with layered relationships. The structure of the environment, E, and its associated regions are also formally defined as an integral part of the C^3RM. As defined by the C^3RM, any discontent may lead to conflicts of concerns, interests, influences, maneuvers, interactions and the impact of weapons which are nested hierarchically to characterize the space-time regions of war, campaign, battle, combat, engagement and armament, respectively.

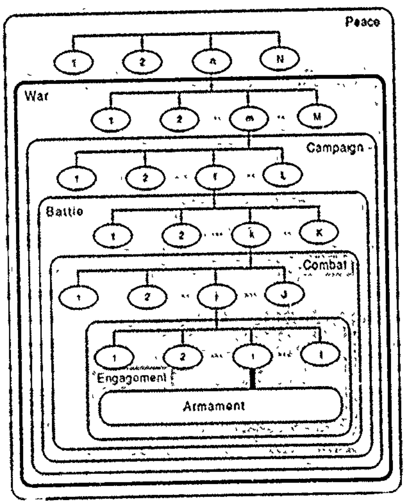


Figure 2. Hierarchical Levels and Nested Regions of Conflict for C^3

The commander is responsible for every C^3 system unit of action, force or power used in peace or in conflict. Conversely no unit of action, force or power should act without the authority of its commander. The authority of the commander may be formalized but must always retain the flexibility to take initiatives as required to meet unexpected conflicts and exploit opportunities which enhance the survivability posture of the C^3 system. Thus, the system which executes commands is strongly coupled to the system which generates commands and, as such, both are treated in an integrated fashion as one C^3 system.

The *mission* of the C^3 system is a primitive notion which defines the *goal, aim, objective, purpose, intent, decision requirement, function*, or *desired state* of the C^3 system. The mission must be derived from the conflict in which the C^3 system is involved. A mission may be decomposed into a series of submissions. At the system level, the mission is supported by layered *services* global to all resources and associated *process entities*, which are local to each resource. The service process entities in turn, may be modularized and refined, stepwise, in a formally descriptive manner, into a series of *activities* and *functions*. Note that whereas the generic structure of C^3 system is common and independent of echelon, the meaning of its specification will vary depending upon the C^3 system frame of reference.

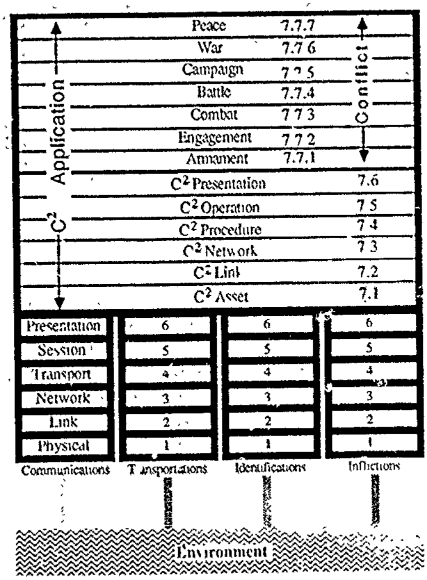


Figure 3. An Autonomous Multi-Interaction Integrated Resource

THE C^3RM STRUCTURE

As shown in Figure 3, the resource structure of the C^3RM is a generic and analog extension of the ISO OSI RM development [2]. Such an extension, however, is non-trivial since it requires significant reinterpretation and generalizations far beyond the present scope of the ISO OSI RM. The C^3RM includes the ISO OSI RM by adopting it for the communications types of interactions. In parallel, layers of three other complementary types of interactions, as defined herein, also provide services to the application layer. The C^3RM embraces anologous architectures for all the key types of physical interactions and utilizes the application layer to provide command and control over all types of interactions in an integrated fashion. The four fundamental types of interactions are identifications, communications, transportations and inflictions. Identifications is an interaction which directly results in the recognition of objects in the environment. Identification is used to determine the stages, phases and targets required for each layer of conflict. Communications is an interaction which directly results in an exchange of information. Communication is used to command, control and coordinate between and among the resources. Transportations is an interaction which directly results in the motion of objects. Transportation is used to carry, supply, strengthen, equip and/or load the resources with the necessary personnel and materiel. Finally, inflictions is an interaction which directly results in the destruction, damage, degradation or disruption of objects. Infliction is used to destroy, damage, degrade, disrupt the capabilities of the targets involved in the conflict. Each resource involved in a conflict, as a minimum, must be capable of communications. The capability for any one or more of the remaing types of interactions depends on the specialty of the resource. For example, weapon resources must be capable of infliction, sensor resources must be capable of identification and logistic resources must be capable of transportation. The autonomous resource depicted in Figure 3 includes the physical assets to interact with other resources using interactions of all four fundamental types in an integrated manner. Note that the different fundamental types of interactions are depicted by the differently textured thick lines connecting the resource with the environment.

Much like the ISO OSI RM Security Architecture [3], security considerations must be also applied at every (sub)layer of every resource and for every type of interaction. Ideally, any breach of security at a particular (sub)layer of a resource should be isolated by the (N-1)(sub)layer and/or the (N+1)(sub)layer and prevented from further propagation. Otherwise, the (N)(sub)layer of the other resources must be able to isolate the defective resource to prevent it from contributing to any further potential compromise. Consider the C^3 system. Every process and every resource is important. Information between and among processes is continually being exchanged explicitly or implicitly. Thus, communications in general is critical in coupling between and among processes which are either co-located or distributed, within a given resource or across a number of resources. In addition, processes may be coupled strongly or weakly, depending upon the need for physical separation between the resources and the potential interactions involved. The processes in any given resource are organized in a layered tree graph fashion to perform the global services required of each global (sub)layer. An (N)process is associated with an (N)entity which performs an (N)function to support a part of the global (N) service called (N)facility. A part of the activity of an (N)entity may be directly governed by a set of global rules called (N)protocols. These rules, depending upon the level of implementation may also be known as logic, laws, policy, tactics, strategy or doctrine. As the system is stimulated by a set of external *events*, the rules available within each resource are applied in succession and/or in parallel to make an "optimum" decision which, in turn, initiates a set of desired actions. Due to uncertainty in system reliability and differences between observations and reality, any one action may result in a variety of *outcomes*.

THE C^2 APPLICATION LAYER

At the highest level of abstraction, the application layer of the ISO OSI RM is called the C^2 Application Layer of the C^3 system. It consists of seven sublayers, as shown in Figure 4, such that each sublayer provides services to the sublayer directly above. Conversely, the services defined at each sublayer rely

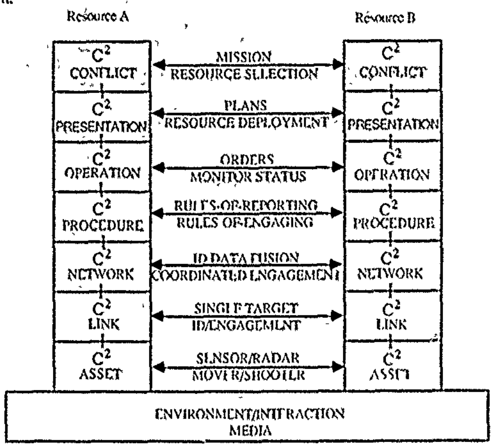


Figure 4. Sublayer Interactions of the C^3RM C^2 Application Layer.

upon the services provided by the sublayer directly underneath The third "C" in "C^3" stands for communications of all contextually meaningful information objects necessary to accomplish the mission of the C^3 System. The media is typically electronic, photonic or audio. Since leadership is a quality of motivation which may be conveyed only through communications, it is imbedded in the C^3RM and the meaning of communications is thereby broadened beyond the mere technical aspects of transmitting and receiving bits of information as layered for the communications port (ISO OSI RM Layers 1-6). The communications aspects relevant to C^3 reside in the C^2 Application Layer and entail logical communications with respect to and via all possible means of interactions needed to be ready for or to resolve a given conflict. Topologically, transportations, identifications and inflictions interactions between and among nodes in a network have isomorphic properies to communications networks. It should therefore be possible to define layered structures analogous to the physical and logical communications interactions as described by the ISO OSI RM. Thus, the C^2 Ai "cations Layer includes application services involving the individual interactions and the interrelationships of all four of the aforementioned types of interactions. For ease of reference, the sublayers of the C^2 Applications Layer are called the layers of C^2 and the sublayers of the highest C^2 layer are called the layers of C^2 conflict. Horizontal communications (peer level) between and among (sub)layers at the same level represent the interoperability of C^3 system resources. Vertical communications represent intra-operability of C^3 system processes within resources. Common services required by each of the layers of C^2 for database, display, and communications capability relevant to the data objects associated with each sublayer are provided for by the C^2 Asset Layer 7.1. Note that the general sublayer structure depicted in Figure 4 is embedded explicitly in the general layered structure of the resources shown in Figure 3. The following sections present an up to date detailed description of the functions and services performed by each layer.

The C^2 Conflict Layer 7.7

Conflict application services provide for the management, supervision and execution of missions. They are primarily concerned with the overall conflict facing the C^3 system, generating, updating and monitoring the mission to ensure that its accomplishment will end the conflict as constrained and bounded by the commander. They ensure that a) the mission is communicated and understood by all resources of the system to the maximum extent possible, b) the appropriate resources can be dedicated, activated or in reserve and are adequately supplied and equipped to carry out the mission related interactions and c) the plans developed by the presentation layer cover all aspects of the mission in an integrated fashion. The mission includes the full definition of success, failure and stalemate. The services provide for the general allocation of functions and resources to perform the functions as well as for the demonstration of leadership and dedication of the resources to the system The main products of this layer are *mission* data objects.

Sublayer Structure of C^2 Conflict Layer

The conceptual layering of adversarial relationships is shown in Figure 5. C^3 systems must be operational in peace and ready for war. War characterizes a space-time region of conflict which is layered hierarchically and nested to serve the general mission of peace. Any space-time region of peace which is threated becomes a region of concern subject to war. Threats may be realized by the prospects of takeover, destruction, disruption, damage and degradation of living conditions resulting from the imposition of general constraints upon the freedom and survivability of the C^3 system From the perspective of the C^3RM C^2 Conflict Layer, war is undertaken to achieve or regain peace in a limited or broadened environment. The mission must also describe any lower layer of conflict in which the C^3 system is to be involved Ultimately, the importance of accomplishing the mission may be motivated as necessary to bring peace, for

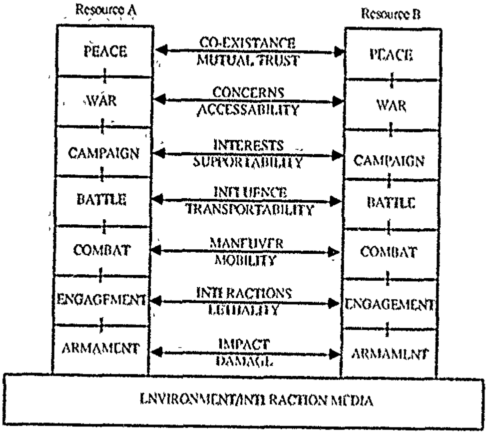


Figure 5. Sublayer Interactions of the C^3RM C^2 Conflict Layer.

the short-term and/or the long-term. Each C^3 system must define its own peace and related conflicts in which it may become involved. Once a conflict is identified it may be resolved or dissolved at any layer. Resources may be positioned in various states of readiness corresponding to each layer of conflict. The reaction of friendly, neutral and adversarial C^3 systems will depend to a large extent on observing and analyzing the state of readiness at each layer of conflict. Thus a resource may be war-ready but not campaign-ready. Another resource may be combat-ready but not engagement-ready. The ability to diffuse or eliminate a conflict at any layer above the armament layer is called deterence. If a resource is said to be ready at a lower layer of conflict, it is automatically ready at any of the higher layers. In the general case, all layers of conflict are needed to characterize a conflict. A conflict cannot be characterized without peace. In various simplifying cases, however, a higher layer of conflict may be degenerate with a lower layer of conflict. The perception of peace and the threat of a conflict is relative to the C^3 system F. C^3 systems G and H may have different perceptions. Regardless of the relative perceptions, the generic characterization of the layer of peace and supportive layers of conflict are common to all C^3 systems.

Each layer of conflict may be staged and phased in time. Staging involves positioning of resources for optimum deployment for that layer of conflict. Phasing invloves windows of opportunities for optimum execution associated with that level of conflict. Each conflict may also be characterized by its intensity. The intensity of the conflict is characterized by the number of resources which are made ready for the conflict at a given layer. Resources may transition from one layer of conflict into another as a result of physical and associated logical interactions. The general sublayer structure depicted in Figure 5 is embedded in the general layered structure of the resources shown in Figures 4 and 3.

The C^2 Peace Layer 7.7.7

The peace layer is responsible for maintaining the null-state of conflict characterizing a space-time region of content and void of conflicts. It is responsible for the missions of identifying potential conflicts, resolving them without a conflict and as a last resort, entering into conflict when peaceful means fail to deter potential adversary resources from launching into conflict. C^3 resources are said to be in peace if and only if they co-exist with mutual trust and support. The products of this layer are *peace mission* data objects.

The C^2 War Layer 7.7.6

The war layer of conflict is responsible for the missions of protecting the concerns essential for the friendly or own resources to win a peace. War type missions may launch a number of campaigns to be conducted in parallel or sequence essential to achieve freedom of access to and from the space-time regions of concern. Mobilization and maintaining an adequate inventory of supply are critical to the success of a war. Campaigns are selected by analyzing their accessibility from the region of concern to the region of interest. The products of this layer are *war mission* data objects.

The C^2 Campaign Layer 7.7.5

The campaign layer of conflict is responsible for the missions of securing interests to win a war. Campaign type missions may launch a number of battles to be conducted in parallel or in sequence to enable to reach the space-time region of interest to be secured. Prepositioning and sustainment of supplies to be made available for ensuing and on-going battles are critical to the success of a campaign. Battles are selected by analyzing their supportability in the space-time region of influence from the space-time region of interest. The products of this layer are *campaign mission* data objects.

The C^2 Battle Layer 7.7.4

The battle layer of conflict is responsible for the missions of achieving influence to win a campaign. Battle type missions may launch a number of combats to be conducted in parallel or in sequence to enable to reach the space-time region of influence. Resupplying and reinforcement of combat-ready resources are critical to the success of a battle. Combats are selected by analyzing the transportability of combat resources from the space-time region of influence to the space-time region of maneuverability. The products of this layer are *battle mission* data objects.

The C^2 Combat Layer 7.7.3

The combat layer of conflict is responsible for the missions of maneuvering to win a battle. Combat type missions may initiate a number of engagements against multiple targets, conducted in parallel or in sequence to enable the reaching of the space-time region of all four types of interactions identified in Figure 3. Inflictions include the firing of explosive projectiles, rockets and missiles and the radiation of the electomagnetic energy. Identifications are derived from observations of sensors and radars. Communications allow for the exchange of information for command, control and coordination. Transportations interactions consists of supplying and replenishing any consumables or perishables that may be needed by the resources. The mobility of engagement-ready resources is critical to the success of combat. Engagements are selected by analyzing the mobility of engagement-capable resources from the space-time region of maneuverability to the space-time region of interactions. The products of this layer are *combat mission* data objects.

The C^2 Engagement Layer 7.7.2

The engagement layer of conflict is responsible for the missions calling for interactions to win a combat. Engagement type missions may launch a number of armaments to be delivered in parallel or in sequence against individual or aggregated targets to enable to reach the space-time region of impact as generated by infliction type of interactions. A single engagement is relative to a single target. Engagements may be composed by the superposition of fundamental topologies for direct, direct support, support reinforcement, general support engagement missions. Note that two fundamental interactions are required to characterize the simplest type of engagement, *i. e* direct engagement. The support of direct engagements requires communications between the friendly resources to command, co‑‑ol and coordinate the more complex engagements. The implicit distinction between general support and direct support is borne out when considering engagements of multiple targets simultaneously. Targets subject to direct support engagements may or may not be subject to general support engagements depending upon the availability of general support resources which span a much wider region of space for engagement. The effective range of the sensors and the delivery range of the armaments are critical to the success of an engagement. Armaments are selected by analyzing the target as a threat potential and the lethality of armament-capable resources as projected from the space-time region of interactions to the space-time region of impact. The products of this layer are *engagement mission* data objects.

The C^2 Armament Layer 7.7.1

The armament layer of conflict is responsible for the missions of inflicting physical and/or logical damage, permanent or temporary, to win an engagement. Any number of armaments may be launched against a single target depending upon its survivability. The accuracy of sighting and delivery as well as the lethality of the impact are critical to the success of an armament. The destructive or disruptive envelope of the armament define the space-time region of damage. The space-time region of damage is a subset of the space-time region of impact. The products of this layer are *armament mission* data objects.

The C^2 Presentations Layer 7.6

Presentation services provide for the management, supervision and execution of plans. They are primarily concerned with understanding the conflict and corresponding mission well enough to draft, coordinate and finalize plans to support various potential missions of contingent conflicts. Planning consists of two phases. The first phase is concerned with the adequacy of existing resources, assets and their organization. At this layer, services may reorganize subordinate resources, their distribution of assets and justify the need for any changes in the quantity and quality thereof. The products from such services are called organization plans. The second phase is concerned with plans for a coordinated sequence of operations with a given organizational structure. Such plans are known as operations plans. Operational plans set schedule constraints to achieve key milestones toward accomplishing a given mission. They also provide guidance for maneuver and thresholds for expenditures of resources and their assets. The main products of this layer are *plan* data objects.

The C^2 Operations Layer 7.5

Operations services provide for the management, supervision and execution of staff functions concerned with generating clear and concise orders to a wide variety of subordinate resources and assets. Orders are generated and updated to select appropriate procedures and synchronize their execution sequentially or in parallel. Operations services are highly heuristic and typically call for the initialization of concentration or relief of effort by available resources at given times and given places. They may also undertake to adapt, modify and rehearse any existing set of procedures. Operational and higher level services evolve directly as a result of the mission in a particular scenario. The main products of this layer are *task* data objects.

The C^2 Procedures Layer 7.4

Procedural services provide for the management, supervision and execution of well-defined, pre-established procedures selected to implement orders derived by operational services. Procedural services may be established scientifically through theoretical reasoning and experimental calibrations. The same procedural services may be called upon to support operations derived from a wide range of missions. For example, procedures include all well defined tactical maneuvers, target reporting and weapon engagement selection disciplines which are incorporated in training as part of doctrine. Any improvisation in procedure should be well coordinated to ensure that potential conflicts are avoided or minimized. Any well rehearsed operation may be instituted at the procedure layer. Procedure services are generally scenario independent and allow for the internetting of complementary resources bridging transportation networks, communication networks, logistic networks, sensor networks, weapon networks and other types of network assets. Procedural and lower level services are known and prepared well before the mission is generated for a given scenario. The main products of this layer are *job* data objects.

The C^2 Networks Layer 7.3

Network services provide for the management, supervision and execution of multilateral, collective interaction of supplementary resources (with similar assets) to create a synergistic effect when compared with individual and link level independent services. Network services include any network of interacting resources such as communications networks, transportations networks for logistics and manuever, sensor networks, intelligence networks, weapon networks and navigation networks. Network services ensure that end-to-end interaction are supported by resources capable of indirect contributions. The main products of this layer are *assignment* data objects.

The C^2 Links Layer 7.2

Link services provide for the management, supervision and execution of one-to-one and one-to-many interactions. Services include communications links, transportation links, logistics links, sensors links, weapons links, sensor-target detections, weapon-target engagements, sensor-weapon coordination and relative navigation. Link interaction response times, throughputs, failures and errors as well as failures and errors recovery are of primary concerns. The main products of this layer are *package* data objects.

The C^2 Assets Layer 7.1

Asset services provide for the management, supervision and execution of individual assets and associated physical processes available within each resource. These assets are more directly involved with any one type of interaction, i.e., communications, transportations, identifications and inflictions. Assets include as a minimum supplies, equipments, and carriers. Supplies include any combination of ammunition, fuel, oils and lubricants, rations and spare parts. Equipments include any combination of weapons, sensors, and communications. Carriers include any type of personnel, vehicle, vessel, air or space craft. Asset services must ensure that individual assets attain the highest possible state of preparedness and operational readiness. In addition to transaction database, display and communications facilities, services at this level are concerned with a) weight, size, power and other compatibility requirements necessary to sustain and interconnect the resources b) maneuverability and navigation necessary to respond to marching orders, and c) assessment of physical destruction, disruption, interruption and/or damage which may be incurred or imparted in the course of any given interaction. Note that the actual performance of management, supervision and execution associated more directly with each individual physical asset is carried out by layers 1 through 6 corresponding to each interaction as shown in Figure 3. The main products of this layer are *transaction* data objects

THE ASSET PORT LAYERS

The framework for interactions of each type is also layered in an analogous fashion to the interactions of the communications type as described by the ISO OSI RM [2] Thus, the layered structure of the C^3 Application Layer defines not only the sublayers of the application layer of the ISO OSI RM for communications but the extension of these same sublayers of the application layers of the isomorphic reference models, yet to be defined, for transportations, identifications and inflictions as well.

Layered Identifications (1-7.1)
Signals retrieved from the environment results in the recognition of objects in the environment. The identification port interactions are subject to further study. Physical, link, network, transport, session and presentation characteristics of recognizing targets employing algorithms ranging from signal /image processing through data fusion are to be described in a common framework to be developed and coordinated herein or by reference.

Layered Communications (1-7.1)
Signals transmitted and received from other resources allow for the exchange of information through the environment. The communication port interactions are typified in Figure 6 based upon the ISO OSI RM [2, 3]. Physical, link, network, transport, session, and presentation characteristics of communications employing techniques ranging from modulation to database and display format and update algorithms are described in a common framework as developed and coordinated for the ISO OSI RM.

Layered Transportations (1-7.1)
Vehicles, manual or motorized, provide for the translation and rotation of physical objects as well as the movement of resources through the environment. The transportation port interactions are subject to further study. Physical, link, network, transport, session, and presentation characteristics of transportations employing techniques ranging from packaging of physical objects to unpacking for expenditure, consumption or coordinated maneuver are to be described in a common framework to be developed and coordinated herein or by reference.

Layered Inflictions (1-7.1)
Armaments provide for the destruction, disruption, degradation or damage of physical as well as logical objects through direct or indirect impact on target resources. The infliction port interactions are subject to further study. Physical, link, network, transport, session and presentation characteristics of inflictions employing techniques ranging from single shots of isolated targets to coordinated engagements are to be described in a common framework to be developed and coordinated herein or by reference.

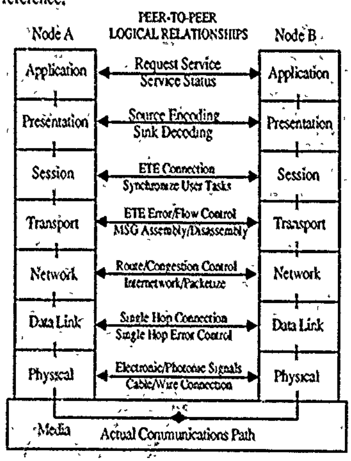


Figure 6. The ISO OSI RM for Communications

GENERIC EXAMPLE
As shown in Figure 7, individual resources, depicted by the seven layers of C^2, may roam about the environment independently carrying out autonomous missions or they may be grouped to complement and supplement each other's effort in an aggregated manner as shown in Figure 8.

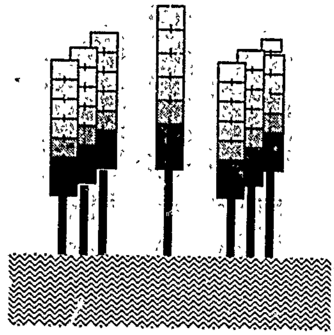


Figure 7. Independently Layered, Individual Resources

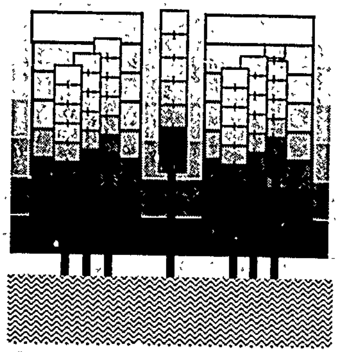


Figure 8 Hierarchically Layered, Aggregated and Distributed Resources.

In Figure 9, the potential interactions among resources are explicitly distributed to depict a more complex three tier C^3 system structure. At the highest tier, denoted by a, one command resource is responsible for establishing the mission, generating, updating and/or selecting plans and monitoring the resulting operations as generated by the subordinate resources, aa and ab. At the middle tier, control resources, aa and ab, are responsible for establishing the operations, generating, updating and/or selecting procedures and monitoring the resulting network interactions as activated by their respective line resources, aaa and aba. The resources, aaa and aba, at the lowest tier, are responsible for establishing a cohesive network of multilateral interactions which utilize and expend the assets at their disposal in support of the system level mission as communicated through channels from the command resource, a. The C^3 system shown in Figure 9 may represent a wide range of C^3 paradigms. In particular, a comparison with Figure 1 may be used to map several possibilities. For example, resources grouped according to {(aaa) (aba) and (a, aa, ab)} or {(aaa; aa) (aba; ab) and (a)} may constitute the observation, action and decision subsystems, respectively. Following the formal description provided thus far and as typified in Figure 9 a more specific example is illustrated.

An Illustrative Example

Consider the topological positioning of resources and their mutual physical interactions for a hypothetical but illustrative example as shown in Figure 10a. Resource C is the commander of System F. Resource S is a sensor of System F which includes its own controller. Resource W is a weapon of System F which also includes its own controller. Resource G belongs to system G. Thus, system F is a simplified version of the C^3 system shown in Figure 9 whereby Resource a corresponds to Resource C, Resource aa and aaa have been integrated or aggregated to represent Resource S and Resource ab and aba have been integrated or aggregated to represent Resource W. Systems F and G are involved in a mutual conflict. More specifically, the conflict has deteriorated down to the level of engagement (Conflict Layer 7.7.2). Note that Resources C, S, and G are fixed sites whereas Resource W is mobile as indicated by the textured arrow for transportation. As shown by the other textured arrows, Resource S is within identification range of Resource G as a potential target. Resources C and S, as well as Resources C and W are within communications range. Finally, Resource W is within range to inflict damage upon Resource G. Having identified the types of interactions which are key to a given resource in a given scenario, each resource may be expanded in terms of the layers which correspond to each interaction type and the application sublayers which couple among the interactions across resources. In the following scenario the proposed layers involved for each proposed service are identified in parethesis.

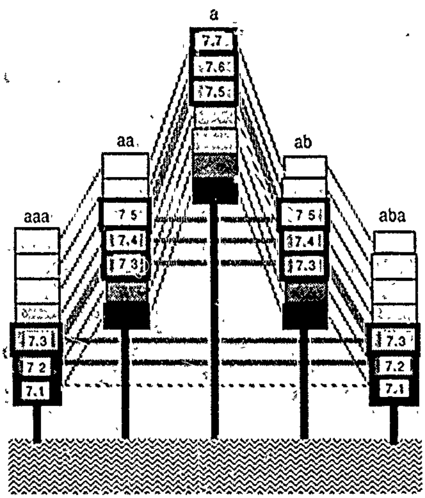


Figure 9. An Abstract C^3 System Structure

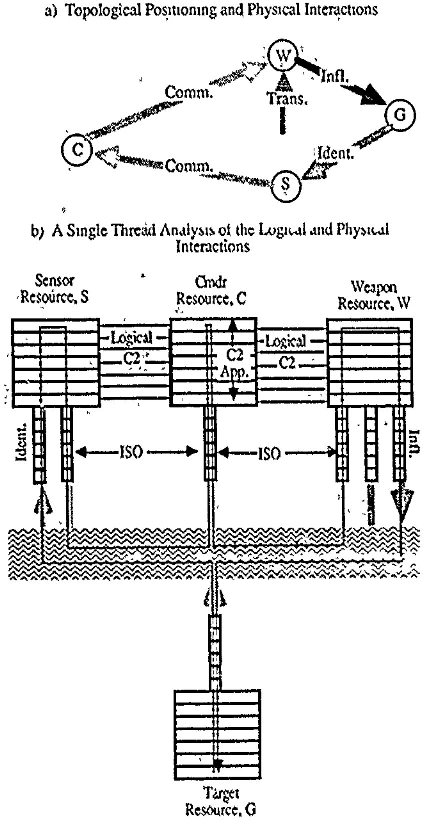


Figure 10. A Single Thread Analysis of Interactions for Illustration

Consider a single thread analysis, as shown in Figure 10b, of only one of an infinite number of sequences of events which could occur from detection by Resource S to the firing of a round by Resource W upon Target Resource G. Using binoculars as its physical asset for the identification port, Resource S detects (Identification Layer 1/S) an activity on top of an adjacent hill. Resource S tracks (Identification Layer 2/S) the activity for a short time period (Identification Layer 3/S-5/S) and notices a large antenna. Resource S performs additional correlation (Identification Layers 3/S-6/S) to determine that Resource G at location-time (x,y,z,t) is an observation post. At Layer 7.1/S, the target information is stored and displayed as a transaction data object. At Layer 7.2/S, the target information is packaged to enable it to be transferred immediately to the commander, Resource C, using its available communications port (Communications Layers 6/S-1/S) and/or for a local assessment of its relevancy with respect to any one of the several available assignments generated at Layer 7.3/S. As an unexpected target of opportunity, an assignment cannot be made by Resource S and Layer 7.4/S must decide if any procedures have been authorized for handling the target as a part of any one of several on-going jobs. Assuming that tasks activated at Layer 7.5/S did not anticipate this type of target at the given location, the plan generated at Layer 7.6/S must be reviewed to see if indeed the recognized target is of interest to the mission derived at Layer 7.7/S for any resource of System F. Assuming that the mission is sufficiently broad, Resource S decides to report on Target Resource G as a target of potential interest to Resource C. Resource S reinterprets, revises or regenerates a plan (Layer 7.6/S), task (Layer 7.5/S), job (Layer 7.4/S), assignment (Layer 7.3/S) and package (Layer 7.2/S) in that order to enble a communications transaction (Layer 7.1/S) to be motivated and formated down through the communications port (Communications Layers 6/S-1/S). The resulting message is transmitted through the environment to Resource C. Resource C demodulates and decodes the message (Communications Layers 1/C-6/C) for presentation to Layer 7.1/C.

Stored and displayed by Layer 7.1/C, the transaction is made available to Layer 7.2 /C for repackaging. Repackaging, even if trivial, is essential for cross-coupling transactions between any two resources. At this point, Resource C may ask for more information. In this example, however, time is of the essence and the commander must decide immediately whether the target is a threat to his immediate mission or some future mission. Being caught by surprise, the transaction which is automatically repackaged for potential infliction is pre-empted by a review process for consistency and practicality with respect to currently active assignments (Layer 7.3/C), jobs (Layer 7.4/C), tasks (Layer 7.5/C), plans (Layer 7.6/C) and missions (Layer 7.7/C) as understood by Resource C. Once consistency is established as the target is processed not only with respect to its immediate and future threat capability but also with respect to the consequences which contemplated levels of infliction might have on the success of resolving the current and future conflicts, Resource C decides to select (Layer 7.7/C) Resource W to engage the target G. An engagement mission is generated (Layer 7.7.2/C-7.7.1/C), a plan is redrawn or revised (Layer 7.6/C), a task is re-issued (Layer 7.5/C), a job is reinitiated (Layer 7.4/C), and an assignment is remade (Layer 7.3/C) to repackage (Layer 7.2/C) the target information in a manner suitable for a transaction (Layer 7.1/C) to Resource W. The repackaged information about Target Resource G is processed by Layer 7.1/C to create a communications transaction which follows the ISO OSI RM services down through the communications port of Resource C (Communications Layers 6/C-1/C), through the environment and up through the communications port of resource W (Communications Layers 1/W-6/W).

Stored and displayed by Layer 7.1/W, the transaction is made available to Layer 7.2/W for potential repackaging. Repackaging (Layer 7.2/W) allows for cross-coupling of communication, transportation and infliction transactions. At this point, Resource W must decide (Layer 7.3/W) whether it is close enough to cause the required damage or whether it should move closer to optimize its probability of impact. Once an assignment is contemplated for infliction (Layer 7.4/W) it is presented higher up to be scheduled as part of an on-going job. The candidate job is then subject to review for priority, consistency, and urgency with respect to existing tasks (Layer 7.5/W), plans (Layer 7.6/W) and missions (Layer 7.7/W). Having decided to respond with a hasty fire mission (Layer 7.7/W), an overall deployment plan is made available (Layer 7.6/W) to check and make sure that Resource S is a safe distance away from possible fratricide (Layer 7.5/W). Having approved the job as minimum risk (Layer 7.4/W), the assignment is approved for infliction (Layer 7.3/W). The target information available from storage (Layer 7.1/W) is repackaged with supplemental timing constraints (Layer 7.2/W). It is then reprocessed to create an inflictions transaction (Layer 7.1/W) which is serviced by the inflictions port ranging from a survey of the weapon's own position to loading the armament and pulling the trigger (Inflictions Layers 6/W-1/W). The armament is launched through the environment and may penetrate up through any port of Target Resource G which may become damaged. In this example, for hypothetical completeness, Resource W is able to destroy the antenna of Target Resource G.

This example may be easily modified and expanded to illustrate many other exan[illegible]s of interest to specific R&D, operations and tests, educatio[illegible] and training organizations. The reader is challenged to come up with his own example to be considered for future reference and as a test for both understanding command and control and this C^3RM.

CONCLUSIONS AND RECOMMENDATIONS

The C^3RM continues to be a viable framework for combined research and development of C^3 systems. It provides a flexible structure to be able to define, describe and evaluate organizations of C^3 systems at any echelon, their measures of performance and measures of effectiveness and potential improvements upon their design in a highly interopetrable yet modular fashion. By following in the footsteps of ISO, it is recommended to be used as a common framework for the development of interoperability standards. It is particulary suited to describe military systems which are layered hierarchically, their mission-oriented resource allocation structures, their interfaces, interactions and integrated operations. Its adoption as a model of choice should prove useful in practical, large-scale, distributed C^3 simulations, developments and operations.

ACKNOWLEDGEMENT

Based upon previous disseminations to many potential contributors, written contributions, endorsements, concerns and constructive comments were incorporated from many individuals. It is hoped that this list will grow as this model is proliferated and the opportunity to use it avails itself. The authors wish to acknowledge their interactions with the following individuals. Eli Beane, COL Gary Q Coe, Herbert Cohen, Frank Effenberger, Dr. Robert J. Fallon, MAJ Patrick L. Gandee, Richard Goodbody, Dr. Irwin Goodman, COL Dickey Hitchcock, Dr Lester Ingber, Dr Carl Jones, NPS, Dr. Theodore Klein, Dr Dirk R. Klose, Clark Mollenhauer, Mike Mudurian, Jack Plant, David Seltzer, Dr Mike Sovereign, Chadwick H Dennis, Dr Earl Craighill, Philip Feld, Dr. Michael S Frankel, Jerry Hines, Dr. Milt Lavin, M. J. Lockwood, Larry Low, Dr. Donald S.

Marshall, Dr. Frances Niedenfuler, Dr. Carol Niznik, Brian O'Leary, Theodore Pfeiffer, Lou Schreier, Dr. Nachum Shacham, George Stringfellow, Dr. Robert Tenney, Bill Walker, Mike Walter.

REFERENCES

1. The Command, Control and Communications Reference Model (RM), Draft Solicitation for Comment #4, Joint Directors of Laboratories Technical Panel for C^3, C^3 Research and Technology Program Document #43-88-R04, June 22, 1988.

2. ISO 7498(Draft) Information Processing Systems - OSI Basic Reference Model, 1984.

3. ISO 7498(Draft) Information Processing Systems - OSI proposed Draft addendum 2 to ISO 7498 on Security Architecture, 30 January 1986.

4. V. Monteleon, " Distributed C^3 Test Bed and User Interactions ", Signal Magazine, pp. 85-91, March 1987.

5. I. Mayk, I. Rubin, "Paradigms for Understanding C3, Anyone?" Science of Command and Control:Coping with Uncertainty, Edited by S.E. Jhonson, and A. H. Levis; AFCEA International Press, Washington, DC, pp. 48-61, 1988.

6. J. E. Holmes and P. D. Morgan, "On the Specification and Design of Implementable Systems" Science of Command and Control:Coping with Uncertainty, Edited by S.E. Johnson, and A. H. Levis, AFCEA International Press, Washington, DC, pp. 93-99, 1988.

ARMY MANEUVER PLANNING: A PROCEDURAL REASONING APPROACH*

Gregory Stachnick and Jeffrey M. Abram

Advanced Decision Systems
1500 Plymouth Street
Mountain View, CA 94043
(415) 960-7300

ABSTRACT

Army corps maneuver planning is the problem of constructing a battle plan that accomplishes a specified mission, given commander's guidance, descriptions of the battlefield environment (terrain and weather), friendly and enemy situations. This paper discusses the Maneuver Planning Expert System (MOVES), which is a component of the DARPA/ARMY AirLand Battle Management (ALBM) program that is being developed to assist corps and division staffs to reduce the length of the planning cycle, thus enhancing the capability for a corps to act faster than the enemy. In addition, we endeavor to generate high quality plans for execution. In particular, we describe the manual planning process, compare that to the system's planning process model, discuss the Procedural Reasoning System (PRS) that is used to represent and execute the planning process, and describe how the resulting plan is represented.

INTRODUCTION

Sun Tzu observed that "a victorious army wins its victories before seeking battle." A goal of the AirLand Battle Management (ALBM) program is to assist the U.S. Army in its victories by providing a collection of cooperating expert systems to support maneuver and fire support planning at both corps and division level. The focus of this paper is one of the ALBM expert systems - the corps level version of the Maneuver Planning Expert System (MOVES).

MOVES is an interactive, knowledge-based, expert system that generates and evaluates courses of action (COAs) for an army corps. The inputs to the system are a mission from echelon above corps, guidance from the corps commander, and data objects containing descriptions of the terrain and weather in the area of operations, the projected enemy situation, and the projected friendly situation; the output is the maneuver portion of an operation order and the COA sketch. The problem, then, is to generate and evaluate multiple COAs that each accomplish the given mission and conform to the commander's guidance for the situation at hand, eventually selecting a single COA from which an operation order is produced. One additional requirement on the system is that it be capable of supporting mixed initiative, i.e., the user controls the degree of autonomy given to the system by inputting in advance as much or as little of a COA as he desires, by modifying decisions made by the system, and by exercising some control over the order in which reasoning processes are performed.

* This research was supported by the Defense Advanced Research Projects Agency (DARPA) and the U.S. Army Ballistic Research Laboratory (BRL) under subcontract 7A40B0820M from Lockheed Missiles & Space Company, Inc., Austin Division.

To design and build a system that can solve the problem described above, it was necessary to study and answer the following fundamental questions:

1. How does the U.S. Army plan today?

2. How does one engineer software to perform maneuver planning?

3. What software technology should be used to capture the maneuver planning process?

4. How does one represent a maneuver plan in machine understandable form?

These four questions are answered in the remainder of this paper.

THE MANUAL MANEUVER PLANNING PROCESS

MOVES is an expert system, and as such, its reasoning processes are modeled after those used by human maneuver planning experts. As a starting point, MOVES is modeled after the doctrinal planning process that is described in various Army field manuals [1,2] and taught at the Command and General Staff College (CGSC) at Ft. Leavenworth, Kansas. This process, called the command estimate, is depicted in Figure 1.

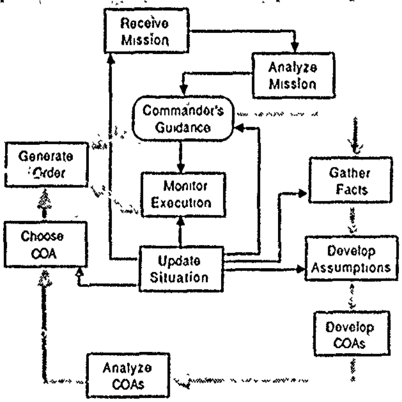


Figure 1: *The Manual Maneuver Planning Process*

The process is initiated by the receipt of a mission. The mission is analyzed to generate a list of tasks to be performed and a mission statement. The commander can then provide the staff with guidance in areas such as courses of action (COAs) that should or should not be considered, the number of COAs that should be generated, where risk may be acceptable, and command and control arrangements. The staff then gathers together the data that will be needed to support the planning process and makes assumptions, where necessary, as substitutes for unknown critical facts.

The steps performed to this point in the process have set the stage for the first major reasoning task: Develop COAs. In this step, which is performed once for each COA to be generated, a skeletal COA is produced. (The COA is fleshed out further during the Analyze COAs step, which is described subsequently.) The subtasks performed here are:

1. Analyze relative combat power - compare enemy and friendly combat power to infer which types of operations are feasible.

2. Develop scheme of maneuver - choose missions for subordinate units, determine how much force is required for each of these missions, allocate units to missions and array them on the battlefield.

3. Determine command and control means - allocate major subordinate command (MSC) headquarters to the arrayed forces and determine control measures such as MSC boundaries, phase lines, objectives and assembly areas.

4. Develop COA statements and sketches - produce written and pictorial descriptions of the COA.

In analyze COAs, each COA is evaluated via wargaming to refine or modify the COA based on the expected outcome, determine the required time sequencing of the various tasks in the COA, estimate the resources required to support the COA, and identify advantages and disadvantages of the COA. This step in the process is highly iterative; wargaming often proceeds until a weakness in the COA is discovered, then the COA is modified using reasoning processes similar to those employed in the Develop COA step, then further wargaming occurs, etc. After all of the COAs have been analyzed, they are compared with one another to identify the COA that will be recommended to the commander

The commander chooses one of the COAs proposed by the staff or asks for modifications to a proposed COA, the COA is expanded into an operation plan or order, and the plan is monitored as it is executed The last task in Figure 1, Update Situation, is a continuous process to collect information from the field to support the information requirements of the command estimate.

THE MOVES PLANNING PROCESS

The reasoning process employed within MOVES is shown in Figure 2 First, an intelligent user interface provides an environment for the corps commander or his operations officer to perform the mission analysis and provide guidance to the system, the results of which are represented internally as tasks, guidance, and constraints Mission and guidance components that specify portions of the solution (resulting plan) are instantiated into the plan object, which is the machine representation of the COA and which is described later in this paper . ne initial fragment of the plan object is expanded with an island growing technique to identify any additional mandatory tasks that must be incorporated into the plan. The resulting plan fragment becomes the common seed from which all COAs will be grown.

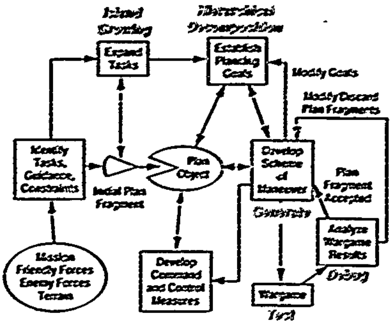


Figure 2: *The MOVES Planning Process*

The system then hierarchically decomposes the mission and intent into sets of requirements for the principal elements of the battle: deep, security, main, reserve, and rear battle. The different ways of decomposing the overall mission into the goals to be achieved per battle element, each decomposition leading to distinct guidelines for the COA generation process, provides one of the mechanisms in MOVES for producing multiple COAs.

After constructing the initial planning goals, the remainder of the MOVES planning process is an iterative constructive approach consisting of three major subprocess components generate, test, and debug.

Problem constraints are employed to conduct a controlled search through plan generation space to construct a plan fragment containing a proposed scheme of maneuver for a selected component of the battlefield. This plan fragment is then evaluated with a heuristic or an analytical wargame If the results of the wargame are unfavorable, an attempt is made to debug the current plan fragment to compensate for unexpected shortfalls identified by the evaluation. If the fragment cannot be debugged, the it is not the right solution within the current planning context, it is discarded and a new fragment is generated. For example, there might not be enough force left when the reserve battle is planned to perform a counterattack Once a satisfactory plan fragment for the battle element has been found, this process iterates over the remaining battle elements.

After completing the development of a scheme of maneuver, command and control arrangements and command measures are generated Multiple COAs for the same force laydown can be generated by considering alternative means of control A good example of this would be to generate one COA that has the corps retain centralized control of the covering force battle and a second COA that decentralizes the control of covering force units to the units in the main battle

CAPTURING THE MANEUVER PLANNING PROCESS

Because the maneuver planning process is highly procedural in nature, the underlying technology that is used in MOVES to capture the process is procedural reasoning [3]. In the following subsections, we describe the Procedural Reasoning System (PRS) that was developed at Advanced Decision Systems in support of the MOVES development effort, then discuss how PRS is used to represent maneuver planning knowledge.

THE PROCEDURAL REASONING SYSTEM

PRS provides a method of encoding procedural logic in a visual representation that is understandable and modifiable by domain experts, e.g., Army maneuver planners, and compilable into efficiently executable code. It supports a goal-directed method of programming that encourages a modular, hierarchical decomposition of goals and subgoals. PRS was inspired by the work of Georgeff and Lansky [4] on plan representation and execution.

PRS combines features of several programming paradigms. As in PROLOG, it encodes inferencing strategies, is goal directed, and features backtracking and retraction of data changes. As in ADA, it has declarative semantics, packaging, and exception handling. As in LISP, it is a symbolic language and facilitates rapid prototyping. As in rule-based systems, PRS also provides a frame system for data representation. Unlike any of these other languages, PRS represents a program visually and includes a graphical editor for program creation and modification.

The fundamental element of a PRS program is called a procedure. A PRS procedure is developed by a knowledge engineer, or possibly by a trained domain expert, and can then be augmented by a software engineer to add declarations and data flow specifications that enable the procedure to be compiled into modular, efficiently executable code. Execution speed of the PRS program is enhanced by performing the bulk of the necessary searching and pattern matching during compilation, at which time the visual representation is translated into executable COMMONLISP code. The visual nature of the language allows a domain expert who is not computer literate to participate more actively in the construction and critiquing of the software than is possible with a standard computer language.

PRS DEFINITIONS

A procedure is a network representing a set of instructions for accomplishing a specified goal. As seen in the sample procedure in Figure 3 (which is a screen dump of an actual PRS computer display), boxes represent actions or goals, circles represent states, and directed arcs represent the reasoning flow

There are a number of different actions that can be used along the arcs of a procedure:

1 Achieve ! effect-name parameters - Find a match for the specified effect in the achievement base. If no match is found, execute each procedure that can achieve the effect until one succeeds or all fail.

2 Achieve (no test) - !! effect-name parameters - Execute each procedure that can achieve the effect until one succeeds or all fail

3. Test ? effect-name parameters - Find a match for the specified effect in the achievement base. If no match is found, backtrack.

4. Perform - @ procedure-name parameters - Execute the specified procedure directly.

There are also actions for asserting affects and for evaluating LISP expressions.

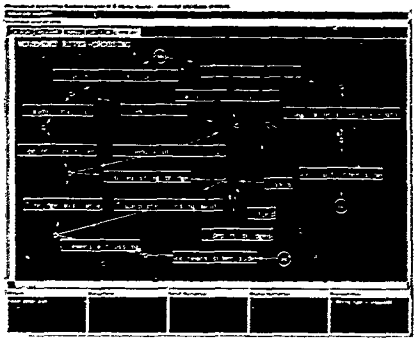


Figure 3: *A Sample PRS Procedure*

There are a number of different node types that are supported within PRS. A start node is the normal entry point for a procedure. End and Break nodes represent successful completion of a procedure, the Break node allowing reentry to a procedure during backtracking. Raise and Exception nodes allow abnormal exit from a procedure, similar to raising exceptions in ADA.

EXECUTING A PROCEDURE

Any path through the network from the Start node to any End node represents a possible successful execution of the procedure and is called a line of reasoning. A procedure is searched in an ordered depth first, left first manner, thus the order of search is inherent in the topological structure of the procedure.

Backtracking occurs whenever an action fails or a goal fails to be achieved. Execution backs up and attempts the next untried branch. Automatic retraction of assertions and assignments occur simultaneously with backtracking, thus, only actions along a successful path or line of reasoning affect the final results.

To illustrate the execution of a procedure, we now step through the example that was shown in Figure 3. Assume that the procedure, named RIVER-CROSSING, has been invoked either by name or by requested achievement of the effect named "reach other side." Before execution begins, the precondition "crossing type = unopposed" is checked, and only if the precondition is true will the procedure be executed. One can have several procedures capable of achieving the same effect. Some can be general purpose, others can be specific to certain situations. The precondition is one method of controlling which procedures can be considered in a given situation. Assume that the precondition is satisfied

The first arc executed from the Start node is an achieve action to find all feasible crossing locations, which fails only if there are no feasible locations. If it succeeds, the next arc executed is a test to see if there are many locations to chose from. Suppose this test fails. The search backtracks to the previous node and executes the other outgoing branch, which is a test that there are only a few lc ations to consider. The system next attempts to achieve the effect called "schedule units crossings", to make efficient use of the limited number of available crossing sites. If this arc fails, execution backtracks again and the arc to achieve a wider area of interest is tried next. Assuming success and returning now to the Start node, feasible crossing locations for the broadened area of interest are found. This time, perhaps there are many locations to chose from, and the next arc prioritizes them. If the test for "bridge available" succeeds, a procedure is invoked to execute the crossing, the system asserts the effect "reach other side," and this procedure is exited with successful completion.

HOW PRS REPRESENTS THE MANEUVER PLANNING ROCESS

The maneuver planning process is encoded within a hierarchy of PRS procedures. High-level procedures encode the knowledge required to solve large subcomponents of the planning problem at a high level of abstraction; low-level procedures encode the fine grained knowledge required to solve small, narrowly scoped subproblems.

At the highest level of the MOVES hierarchy is a procedure called MOVES-TOP, shown in Figure 4. This procedure captures the entire maneuver planning process at a very high level of abstraction. Tracing through the procedure, one sees the steps of developing multiple plans, recommending one COA to the commander, refining the COA to respond to feedback from the commander, expanding the COA into a full plan, and notifying the force control component that the plan is ready to be processed by a block text generator to produce the operation order.

Two levels below MOVES-TOP in the hierarchy, beneath the arc labeled "!! develop-plans," is the DEVELOP-COA procedure, depicted in Figure 5. This procedure captures the process of generating a single COA. The steps are to define the close battle area (which also defines the other battle areas relative to the close battle area), incrementally develop plans for the close battle, deep battle, and reserve battle, allocate any units that have not been committed to any battle, and then determine the command and control measures.

Two more levels down, exploring the "!! determine-reserve-effort" arc, is the procedure "DEFEAT-SECOND-ECHELON." This procedure develops a plan for defeating the enemy second echelon forces with the reserve battle. As seen in Figure 6, the procedure first explores the possibility of counterattacking the entire second echelon (in our current scenario, this is two Soviet tank divisions), first with nominal probability of success, then with some additional risk accepted. If there is not enough force available for the reserve to counterattack, the system will attempt to achieve a separation of the elements of the second echelon so that they can be dealt with individually. If the separation can be achieved, the system tries to counterattack the lead elements of the second echelon and defend against the follow-on elements. If the separation cannot be achieved, the system attempts to plan a defend mission for the reserve against the entire second echelon. If the system does find a mission that seems to be feasible for the amount of force available, it produces a plan fragment, evaluates the fragment with a wargame, analyzes the wargame results to see if the desired effect was achieved, and either accepts an adequate result or backtracks to try another mission if the result was inadequate.

It may be the case that too much force was allocated to the close battle, leaving insufficient force for the reserve to even defend. In this case, the DEFEAT-SECOND-ECHELON procedure would fail after having tried all of its known methods for achieving the defeat, control would return to the calling procedure with failure as the result, and reasoning processes higher up could choose between removing forces originally

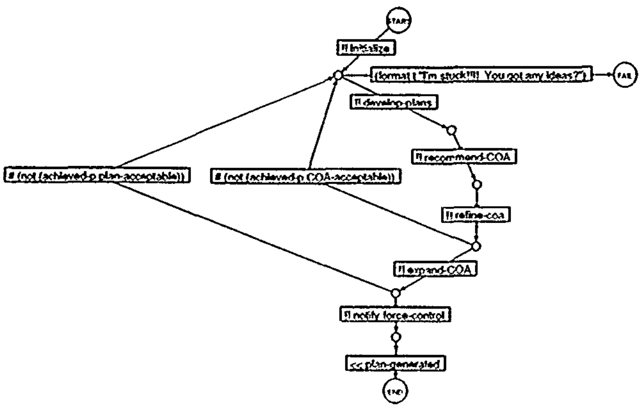


Figure 4 *MOVES-TOP Procedure*

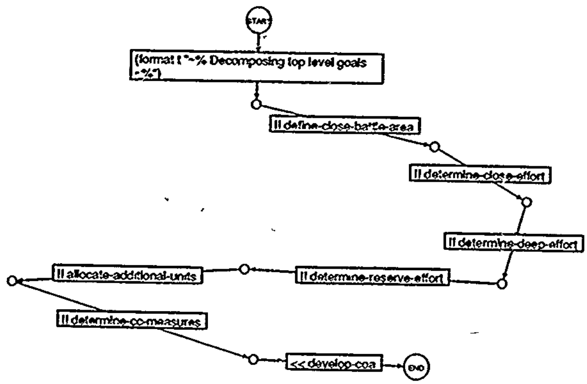


Figure 5: *DEVELOP-COA Procedure*

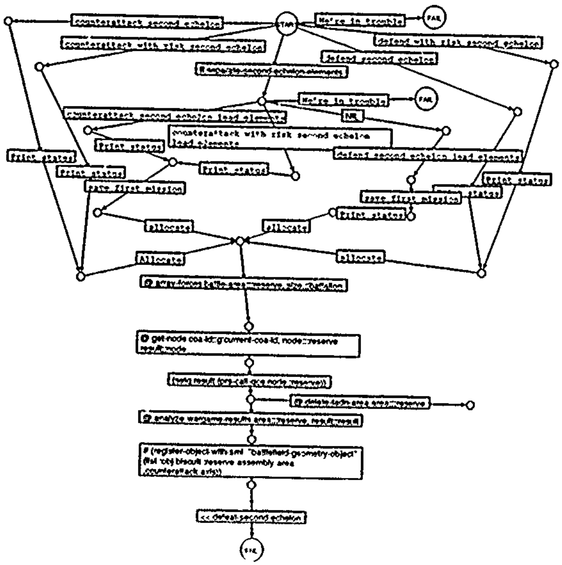


Figure 6 *DEFEAT-SECOND-ECHELON Procedure*

committed to the close battle to give to the reserve or changing the reserve goal to something that could be accomplished with less force than the original goal of defeating the second echelon.

Below the DEFEAT-SECOND-ECHELON procedure are a number of further layers that perform actions like determining a counterattack objective, calculating how much friendly force is required to counterattack the given enemy force, arraying the required force on the battlefield, etc. As one proceeds deeper into the hierarchy, the problem being addressed by each procedure becomes smaller and the reasoning knowledge represented becomes more detailed. At the lowest level, there are primitive procedures whose arcs can invoke LISP code, but cannot invoke subordinate procedures.

REPRESENTING THE PLAN

The results generated by the ALBM planning process must be stored in a representation that is sufficiently complex to capture the complete description of the tactical actions that compose the plan and the relationships that exist between various tasks and physical areas on the battlefield. The plan object consists of four major components.

- o Task Organization
- o Battlefield Geometry
- o Fire Support Plan
- o Tactical Action Dependency Network

TASK ORGANIZATION

The Task Organization (TO) component describes the operational command structure for the plan. It identifies how the various major subordinate commands are structured as a function of the different phases of the plan. It reflects the attach and opcon (operational control) relationships that have been derived by the planning process to account for shortages or extras in the organic unit compositions.

BATTLEFIELD GEOMETRY

The Battlefield Geometry (BG) component of the plan object, represented in Figure 7, contains the information describing terrain related control measures. This information includes the designated areas of operations for each of the major subordinate commands, assembly areas, attack and counterattack objectives, engagement areas, and axes of advance. Certain coordination information is also included in this schematic in the form of phase lines. A particular friendly or enemy unit crossing a phase line can be the trigger for termination of some current activity and initiation of the next. The BG object also contains information describing how the various battle elements (security, main, deep, rear, reserve) are spatially defined for the course of action. Part of this component is filled in with inputs provided by echelon above corps, e.g., the corps area of operations and perhaps the Forward Edge of Battle Area (FEBA) or covering force coordinating points. The remainder of the information is either provided by the system user or is determined as part of the maneuver planning process.

FIRE SUPPORT PLAN

The fire support component of the plan object describes which fire support assets are retained under corps control and which assets are assigned to the major subordinate commands. The specific sets of fire support tasks that support the maneuver missions are incorporated into the tactical action dependency network, described below.

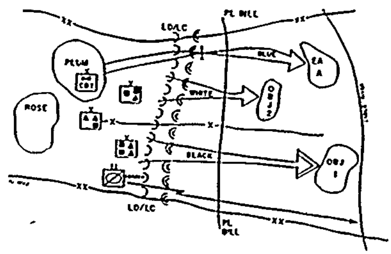


Figure 7: *Battlefield Geometry Component*

TACTICAL ACTION DEPENDENCY NETWORK

The Tactical Action Dependency Network (TADN) is the most important and most complicated of the plan representation components. This component captures the tasks that have been assigned to the corps and its subordinates, and the various dependency relationships between tasks that, taken together, describe the sequence of activity making up the plan.

The TADN is a network of task and conditional nodes, with the arcs representing hierarchical task decomposition, temporal, and tactical proximity relationships between the nodes. Figure 8 illustrates the plan object from the task decomposition perspective. The TADN begins with a top-level reference node representing the task which corresponds to the given corps mission. The network is hierarchically organized according the decomposition of tasks into subtasks. Thus, nodes deeper in the structure are more detailed and represent units at lower echelons. In Figure 8, the first decomposition would represent the corps tasks for the deep, close, rear, and reserve battle elements. The specific decomposition of the corps close battle element task could be found in the division level segment of the diagram.

Each TADN node contains a task object and sets of parent links, child links, temporal links, and tactical proximity links. The task object stores basic information about the kind of tactical action that is to be performed on the battlefield. The task object contains the following:

Who - The unit performing the action
What - The activity to be performed
What Direct Object - Object to which the activity applies
Where - The location
When - The time
Why - The purpose to be achieved
Why Direct Object - Object of the purpose
How - The specific form of maneuver or another decomposition

For example, consider the following task:

On order (when) the 208 ACR (who) attacks (what) the 79 Tank Regiment of the 27 Division (what direct object) at objective blue (where) to force deployment (why) of the 9 Tank Regiment of the 27 Division (why-direct object).

In this example the "how" was not given, indicating that the way in which the mission is accomplished is implicitly described in the task objects of subordinate nodes in the network.

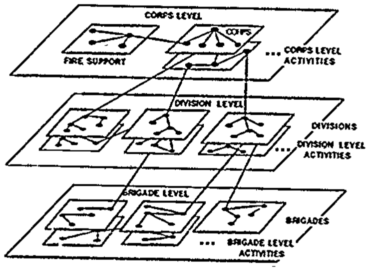


Figure 8. *TADN Task Decomposition*

Additional information about the task is also available for determining whether the task is a principal maneuver task or a support task. Some tasks have special invocation and terminations that are not derivable from the temporal relations. For example, when the 208 ACR is pushed back over Phase Line Bill, it transitions from a "defend" mission to a "delay" mission. Crossing the phase line is the termination condition for the "defend" mission.

Whenever the maneuver planner determines that a particular temporal ordering of two tasks is required, the relationship is represented as a temporal link connecting the corresponding task nodes There are thirteen possible values that describe the way in which tasks can be temporally related, the temporal relationships being based on Allen's theory of time [5] These relations have been defined as follows:

before (A,B).	A ends before B starts
after (A,B):	A starts after B ends
overlaps (A,B):	A starts before B starts, ends after B starts, and ends before B ends
overlapped-by (A,B).	A starts after B starts, ends after B ends
equal (A,B):	A and B have same start and end
starts (A,B):	B starts when A starts
started by (A,B).	A starts when B starts
ends (A,B):	B ends when A ends
ends by (A,B):	A ends when B ends
meets (A,B):	A starts when B ends
met-by (A,B)	B starts when A ends
during (A,B)	A starts after B starts, and A ends before B ends
contains (A,B)	A starts before B starts, and A ends after B ends

Allen uses these relationships to describe the possible temporal relationships that can exist between two time intervals A temporal link between two TADN nodes has a stronger connotation: the link represents a requirement that a particular temporal relationship exist between the two nodes. The information contained in the "when" slots of two nodes can be examined to determine the temporal relationship that exists between the two nodes, but the nodes could be temporally independent and the relationship could be purely coincidental However, if a temporal link connects the two nodes, then the temporal relationship must be maintained, and changing the time interval associated with one of the nodes may cause the time interval of the other to be adjusted accordingly.

TACTICAL PROXIMITY

When allocating the command and control measures after forces have been arrayed, it is important to know which tasks in the network strongly influence other tasks in the network Such tasks could be considered as being pa of a single battle. This relationship between nodes is called tactical proximity. The values for this relation are indicated by tactical proximity links. The acceptable values are:

Tactically-close (A,B):	strong influence
Tactically-distant (A,B):	negligible influence
Tactically-intermediate (A,B):	moderate influence

The concept of "tactically close" does not depend on physical distance only. For example, two defend activities on adjacent avenues of approach may be "tactically distant" if a significant obstacle exists between the avenues If two tasks are "tactically close," there is strong motivation to assign both tasks to the same major subordinate command.

CONDITIONAL NODES

A single set of tactical actions may not always be adequate to respond to sets of differing enemy actions. In this case the plan representation must have the capability to incorporate a branch. A branch is indicated by a conditional node. A conditional node consists of sets of parent, child, and temporal links, but in addition it has a list of "control forms" Each control form consists of a LISP predicate and an accompanying list of task nodes to be executed if the predicate evaluates to true during plan execution. This predicate corresponds to a test against the world states that result from execution of the plan. For example, the predicate might test whether enemy units move to the north or south at a particular intersection of major roads If the predicate indicating enemy goes north is true, the nodes corresponding to that branch of the plan are executed as required. These sets of predicates also help to identify requirements for what kind of data needs to be collected about the battle as events unfold.

CONCLUSIONS

This paper has discussed the ongoing development of the Maneuver Planning Expert System (MOVES), which is a major component of the AirLand Battle Management program. We have designed, implemented, and demonstrated the first prototype of a system that generates corps manual decision-making process. Both the manual process used in the U.S Army today and the process model built into the system are highly procedural in nature, leading to the conclusion that the Procedural Reasoning System (PRS), developed at Advanced Decision Systems, is an appropriate technology with which to implement the MOVES system.

PRS has a number of features that make it an appropriate software development environment for MOVES development

1. It is designed specifically to address the problem of representing procedural knowledge.

2. Its graphical representation of both domain knowledge and inference control mechanisms make the inherent reasoning processes visual to the software developer and, more importantly, to the computer illiterate domain expert, who can then interact with the developer directly through the code, rather than through the developer's explanation of what the code is supposed to do.

3. The PRS programming paradigm encourages the development of modular, hierarchically structured software It supports the ability to develop the reasoning knowledge only down to a certain, desired depth, with the logic below that level easily "stubbed" out until it is

desired to fill in more low-level knowledge. The PRS development environment supports an incremental approach to software development through rapid prototyping.

4 Much of the search and pattern matching is performed at compilation time, providing excellent performance at execution time, unlike most expert system development environments.

PRS is used to represent the maneuver planning reasoning processes, but is not used to represent the plan that is dynamically generated by the MOVES system. A complex plan object has been developed to represent the actions that make up a plan, the various dependency relationships in effect between these actions, and the measures in place to control the plan.

Over the remainder of the project, MOVES will evolve as we continue to expand the knowledge encoded in the system in the form of PRS procedures, adding more breadth to the types of missions for which the system can generate plans, and adding more depth to the knowledge already captured.

ACKNOWLEDGMENTS

The authors wish to express their gratitude for the helpful comments provided by James P. Marsh and James R. Greenwood, ADS, and by Morton A. Hirschberg, US Army Ballistic Research Laboratory, APG, MD. We also wish to acknowledge the efforts of our ALBM project co-workers.

REFERENCES

[1] Department of the Army, FM 100-5, *Operations*, 1986.

[2] U.S. Army Command and General Staff College, Student Text 100-9, *The Command Estimate*, 1986.

[3] Georgeff, Michael P., and Lansky, Amy L., "Procedural Expert Systems," in *Proceedings of the Eight International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, 1983.

[4] Georgeff, Michael P., and Lansky, Amy L., "A System or Reasoning in Dynamic Domains: Fault Diagnosis on the Space Shuttle," Technical Note 375, SRI International, 1986.

[5] Allen, James F., "Towards a General Theory of Action and Time," *Artificial Intelligence*, vol. 23, 1984.

HEURISTIC ROUTE OPTIMIZATION PROGRAM
An Extendable Route Planning Concept

Robert J. Kruchten

Rome Air Development Center
RADC/COAD
Griffiss AFB, NY 13441-5700

ABSTRACT

A number of automated tools have been developed to find suitable paths for aircraft penetrators in a surface-to-air missile environment. Unfortunately, the techniques used by the existing tools force them to ignore many factors (speed, altitude, EW, Threat alert status, number of penetrators, etc) that are so relevant to the problem. The data bases of the existing tools are also difficult to modify for moving threats. RADC developed an in-house concept (Decision Aid for Threat Penetration Analysis, DATPA) that was meant to overcome many of these shortcomings. DATPA simplified the process through a combination of heuristics and object-oriented programming. These techniques allow the program to encompass EW, saturation, or other tactics so that the path recommended is more survivable. These improvements can be realized because the heuristics dramatically lower the number of potential routes and the object-oriented representation creates robust data structures that are easily manipulated and duplicated.

INTRODUCTION

"For centuries, successful national military strategies have been based on principles of war learned in equally as many centuries of military experience. Those lessons came hard; and at great cost in lives and gold, and in national power...(These) principles of war... have been successful for more than 2500 years. We ignore these lessons at our peril."

General Curtis LeMay (1962)

General LeMay's concerns are valid and must be considered as we begin to automate our planning processes. This paper discusses a way to include the "Principles of War" in our automated planning systems. The Air Force identifies the "principles of war" in AFM 1-1. They include such fundamental concepts as: objective, offensive, mass, economy of force, surprise, maneuver, etc. Unfortunately, as we begin to introduce automated tools into the planning process, we have sometimes avoided these principles. This is especially true in the field of route planning.

RADC developed a concept of route planning that uses some unique techniques to improve the route planning process. This concept was described in the Spring 1986 NATO Advisory Group for Aeronautic Research and Development as the Decision Aid for Threat Penetration Analysis (DATPA). Although DATPA was originally developed as an alternate approach to route planning that could be extended, its basic ideas lent themselves to a system that can more effectively create a plan utilizing the traditional principles of war. This paper will discuss a concept of automated planning that can accommodate these principles of war using the techniques found in DATPA. Properly applied, the tool described here could serve for planning individual sorties and, more importantly, could suggest major theater level tactics.

PROBLEM

Although the principles of war mentioned above apply to all levels of command, they are most important at the higher levels. Route planning, as used in this paper, refers to the path used by sorties to accomplish their objectives. Obviously routes must be planned at the unit level for each individual sortie in order to actually fly the mission, but route planning should also be a major factor in theater planning. In the terms of war principles; the decision of where to mass forces, how to achieve surprise, the size of force necessary (economy of force), etc., are all integrally intermixed with the routes flown. Thus, many routes may converge to achieve a critical mass against a defense or certain routes may be chosen to deceive a defense and enhance surprise. Ideally, all automated route planning systems should encompass these principles. Unfortunately, this is not the case.

Most existing automated tools for route planning are limited to minimizing the threat exposure for a single sortie. This is usually accomplished by building an array representing the threat lethality/time and then using an algorithm that minimizes the sum of moves through the array. This results in a path (route) that minimizes the exposure to threats. Various techniques are used to speed this process and limit the search. Tools that use these methods are incapable of fully accommodating the principles of war mentioned above. The tools are limited because they use a "static" representation of the threats and they are computationally intensive. Thus the routes generated are insensitive to massed penetrators, deceptive tactics, surprise, etc.

Ideally the automated tools should suggest not only individual routes, but also the best plan that exploits basic principles of war. To accomplish this, the automated tools must be sensitive to factors such as massed penetrators, surprise, deception, etc.. This implies a tool that represents threats in a more dynamic manner. Real threats do not have a fixed, unchanging capability against all penetrators. Instead, real threat characteristics change as a function of the penetrator type, the threat alert status, the number of threats, the local terrain, the munitions status of the threat, etc.. Some of these factors can get quite complex. For example, if two threats are somehow in communication with each other; flying past one automatically alerts the other and increases its probability of kill. This means a threat's capability may be a function of the route flown (i.e. threats passed). This becomes a sort of incestuous type of problem. The route should be a function of the threat probability of kill and the threat probability of kill may be a function of the route.

TECHNIQUES

Three techniques, integrated together, could be used to create an automated tool that could plan individual routes and could also recommend specific theater tactics that would satisfy basic principles of war. The first of these techniques is the use of heuristics.

Heuristics or 'rules of thumb' can be used to simplify problems and to speed search processes. For route planning, one simple heuristic is to avoid threats where possible. Threats can be avoided by going around them. Two paths can be created to go around any threat (clockwise and counterclockwise). Thus, one simple heuristic is that upon encountering a threat attempt going around it clockwise and counterclockwise. Using this heuristic, the search space in a dense threat area is limited to the edges of threats. This dramatically reduces the number of potential paths. Other heuristics can include tactical actions taken upon encountering a threat (i.e. jam the threat, deploy chaff/flares, maneuver, kill the threat, change altitude/speed, etc.). The application of these heuristics can be viewed as a means of creating alternate paths as threats are encountered. Thus, as a threat is encountered the heuristics may create clockwise paths at different speeds, paths that expend chaff, paths that kill the threat, etc.. Each of these generated paths can be evaluated for distance, lethality, and practicality with impractical paths rejected and the best (distance/lethality) extended. The heuristics used to create the alternate paths can be collectively evaluated by an expert system as threats are encountered. Using an expert system allows the heuristics to be easily added and modified. Most importantly, the expert system allows the heuristics to be captured as simple statements that are easily understood rather than as complex mathematical algorithms.

The second technique is "object-oriented" programming. Object-oriented programming is a way of programming that develops the program as a series of objects. An object is something that has behavior characteristics. That is, the object behaves in some stated manner. The behavior is caused (invoked) by sending an object a message. Objects can inherit characteristics from other objects and objects can send messages to other objects. For route planning, the threats can be treated as objects. Each threat type can be described as an object. The description would contain the basic characteristics of the threat (capability against various penetrators, susceptibility to attack \$EW, or othert, sensor types, physical cooperativeness, munitions, etc.). The description would also include behaviors such as how to go around the threat, threat doctrinal considerations, etc.. Threat behaviors would also include the procedures for the threat to maintain its current status. Thus the threat would automatically keep track of its alert status, munitions availability, communications, etc.. This status could easily change as a function of time and events. As instances of the threat were created, information specific to the instance would be constructed. This includes the local geographical information, connectivity to other threat instances, current munitions loading, current alert status, etc.. It would be possible to send a message to a threat instance and have the threat instance create a path(s) around itself. The message could easily contain arguments that would affect the paths created. Examining the incestuous problem of interconnected threats described above; it is easily possible to pass as an argument the threats already encountered. The threat could then use this argument to determine whether to change its alert status and thus its probability of kill. Object-oriented programming has a side benefit in that it can be very concisely coded. This means a very complex program is both easy to develop/test and it is relatively small.

The last technique is a branch and bound search with dynamic programming. A branch and bound search is a search that progressively constructs paths and links them to previously created paths. The branch and bound algorithm first constructs a series a of paths from an initial location and selects the least cost path, saving the remaining possible paths. Next additional paths are constructed from the chosen path location and the sum total of those paths are compared with the previously rejected paths. The lowest cost total path is next extended. This process is continued until the goal is reached. If a location that had been previously reached is reached some later path, the later path is deleted since best paths have always been extended first. As paths are created important statistics can be retained on each path. Thus the remaining fuel, current altitude/speed, aircraft configuration, consumable (chaff, etc.) status can all be associated with a path. Therefore two routes flying the exact same course, but at different speeds would be treated as two separate paths. The statistics allow paths that exceed capabilities to be rejected (e.g. a path that consumes too much fuel). This search technique has the advantage of yielding a mathematically optimum solution without examining all possible alternatives. The primary disadvantage of this technique is that it can involve many alternative paths. The heuristics described above can be used to limit the number of paths examined.

SYSTEM CONCEPT

The techniques described above can be used for an individual (single sortie) route planning system.

The system would have a module for the threat information. This module would contain a high level threat object. This object would contain the methods and procedures common to all threats. The module would also contain objects for each threat type. The threat type objects would inherit characteristics from the threat object. The module would contain additional objects for key terrain features, enemy units, weather, or targets that may influence the route chosen. These objects could be similar to the threat objects; containing procedures/methods for navigating around them or through them. The basic objects would be an integral part of the system when used in any application. The information contained in these object descriptions would change infrequently (i.e. when new threat types are discovered). Instances of the various objects would be created in the module to correspond to a particular tactical situation. As the instance were created, the information specific to that instance would be constructed. This may include information such as the terrain masking at selected altitudes, communications networks, etc.. As the instance was created the instance could ensure that the paths avoiding it were flyable. This can be done via a set of heuristic rules and a knowledge of the features around the object. The object instances would be expected to change frequently as the tactical situation evolved. Since most of the information resides in the object description rather than the instance, it is relatively easy to change/modify the instances. The name of each object instance would be posted on a map corresponding to the area affected by the instance. Thus, given a particular map location, messages can be sent to all objects capable of affecting routes in that area. The objects themselves can reply giving alternative routing possibilities.

Another module in the routing system would contain the branch and bound search and dynamic programming algorithm. This module would construct a tree type of search; building new paths on the tree as new objects (threats) are encountered. The algorithm would start at that origin point for the sortie and would attempt to create a path toward the objective (target). As object were encountered, the algorithm would send messages to the objects requesting alternative paths in relation to the object. The alternatives would be created using the heuristic system. Thus one alternative might be to follow a path around the threat at the current altitude/speed; another might be to jam the threat and continue on the current course; another might be going around the path at a different speed (and fuel consumption); etc. The number of alternatives created depends on the characteristics of the heuristics, but the possibilities are virtually endless and limited only by the eventual processing time to evaluate the alternatives. As each object answered the messages with an alternative path, it would also provide the expected lethality and distance of the path. The ability of objects to send messages with each other plays a part also. If a clockwise path around a threat encounters another threat, that path can immediately be modified to reflect the additional lethality of going through the new threat. In addition new paths would be created by the new threat. For example, a counterclockwise path around the new threat and through the current threat would be created and a clockwise path around the new threat would be created. The lethality and distance of these new paths could be calculated immediately because all information on both threats could be determined via messages. Therefore the algorithm could evaluate and extend the best route while eliminating dynamically unflyable routes. A key consideration in this process is the creation of alternative paths only as new objects are encountered. This dramatically lowers the search alternatives to compare. High path resolution is maintained since the path generated by the threat can be any resolution desired.

A final key module in the routing system would contain the heuristic rules and an inference engine to evaluate the rules. Thus module would interact with the algorithm described above. Although many heuristics could be incorporated, it may be best to limit the heuristics to avoid long processing times.

Additional modules would be needed for the man-machine interface, interface with threat data base, rule base editing, and object editing. Ideally the system will ultimately automatically interface with some outside source for threat update and will allow the user full freedom to revise the doctrine/tactics that would be embedded in the rule base.

The architecture has been described above as an individual route planner. As such, it offers significant advantages over current approaches. Not only would the planner provide a single altitude, single speed route; it would also be capable of suggesting a route with varying speeds/altitudes plus when to use various tactics (jamming, chaff, etc.). This happens because the search algorithm was actually creating and comparing routes with legs at various speeds/altitudes/tactics. The ultimate route recommended would represent the best for all factors involved. More importantly, perhaps, is the capability of the architecture to go beyond single sortie planning.

Using the same basic architecture the system could recommend corridors to saturate defenses or deception tactics. It also can recommend where to use scarce force assets such as standoff jammers or threat suppression systems. This can be done by exploiting the dynamic representation of the threats in the planner. The system has an inherent capability to evaluate paths involving multiple sorties with the threat capabilities changing appropriately. A key consideration is that the best force plan is not merely the sum of individually optimum plans. Individual planning precludes the synergistic impact between the sorties. The object-oriented threat representation allows for this synergistic effect to be properly accounted for. Thus the planner penalizes the second or third sortie past a threat (due to a higher threat alert state) and rewards the saturation of a threat (due to C2 or munition limitations).

CURRENT STATUS

As described above, RADC developed the Decision Aid for Threat Penetration Analysis (DATPA) as an in-house program programmed in Zetalisp on a Symbolics Processor. DATPA uses simple heuristics and object-oriented programming to find a path through a series of threats. DATPA can also indicate where to best use limited Hard-kill capabilities against threats. However, DATPA was meant only to demonstrate the concepts and it does not include all of the characteristics described above. DATPA also does not use realistic threat representations, realistic aircraft polynomials or provide complete, flyable routes. Thus, DATPA shows the value of the basic concepts, but does not prove the total system feasibility.

RADC is in the process of starting a program that will use the techniques described here, plus realistic data to prove the system feasibility. This program, Heuristic Route Optimization (HERO), will build a system capable of planning single and multiple sortie routes. The routes created will be a function of all the factors described above. In particular, the routes will develop routes that apply the basic, historic principles of war.

RECOMMENDATIONS

If HERO does fully prove these techniques, the HERO architecture should be included in both unit level planners and in theater level planners. It would be especially helpful as an adjunct to a system performing Air Tasking Order (ATO) generation such as the Tactical Expert Mission Planner (TEMPLAR) program. Whereas TEMPLAR properly applies the necessary constraints for weapon system employment, HERO can provide the large scale tactics for that employment. That is, HERO can suggest entry/exit corridors, where to use EC assets, possible deception tactics, etc. In summary HERO uses new programming concepts that allow historical planning considerations (Principles of War) to be introduced into automated systems.

MEASURING FORCE MULTIPLICATION

Colonel Gary Q. Coe, USA
Chief, Modeling and Analysis Division
J6, The Joint Staff, Room 1D834, Pentagon
Washington, D.C. 20301

ABSTRACT

Demonstrating value of command and control systems against weapons hardware has been a particularly difficult challenge for combat modelers. The article contends that the battlefield is a nonlinear environment which the analytical community persists in replicating using linear means. Even when this inconsistency is realized, modelers often read nonlinearity as "impossible to model because of the scarcity of nonlinear mathematical tools".

In response to this notion, the article discusses the emerging science of complex systems and the potential it offers in dealing with this challenge. The goals are to articulate exactly what we mean when we describe command and control as a "force multiplier" and to use this in permitting us to choose between the more effective systems - command and control or weapons hardware.

To this end, the article explains how to effectively simulate command and control issues using notions from complex systems theory in an environment of real world measures.

INTRODUCTION

How does one measure force multiplication? The answer to this question is fundamental in evaluating the value of C3I to force capability. With it, we can analyze the tradeoff value of C3I systems to weapons systems or address the balance of manpower, weapons and control within force structure. Without it, the defense community will continue to depend on anecdotes rather than rigorous analysis in relating command, control, communications, and intelligence programs to combat outcomes.

GENERIC MEASURES

What we need first is an agreed upon set of measureables for command, control, communications and intelligence. The literature is packed with measures and we could do an exhaustive survey of these but we will not. For brevity, we can hypothesize that there are four which are shown in the following figure.

MEASUREABLES

ENTITY	EXAMPLES	MEASURES*
MESSAGES	COMMUNICATED FACTS	TIME TO RECEIPT
MESSAGE PATHS	ORAL, ELECTRICAL PROCEDURES, DOCTRINE	CAPACITY**, SPEED ACCURACY
DECISION PROCESS	SEE DECIDE ACT	GOODNESS OF RESOURCE APPLICATION
RESOURCES	MANPOWER, EQUIPMENT &	NET USAGE RATE

* ALL HAVE TIME RELEVANCE PURPOSE OF COMMAND IS TO APPLY RIGHT RESOURCES AT RIGHT PLACE AND RIGHT TIME

FIGURE 1

Messages are simply communicated facts which can be measured by time to receipt. Messages are most relevant to their operational need. Those without operational need are also relevant since they may increase the noise level, further increasing time to receipt of the former.

Message paths are routes of information from one node to the next and are often the most measured measureable since measures are performance specifications which can be easily observed.

Decisions are most correctly measured by time windows of opportunity, but these are often not precisely known. Therefore we are dependent on the goodness of resource application as an observable measure of their effectiveness.

Net usage rate of resources is the best overall measureable, linking the combined measures of messages, paths and decisions. These are often used without linkage to information flow, causing them to be misused.

SYNERGISM

Force multiplication is synonymous to synergism - the idea that the whole is larger than the sum of its parts. This gives us a "back door" key to quantify force multiplication. We can measure capability of individual weapons systems by physical output. These can be added together to give a "the sum of the parts" measurement of force capability. Mathematically this is the relationship of linearity.

NONSYNERGISM

FORCE ELEMENTS: x & y (e.g. INFANTRYMEN, SHIPS, AIRPLANES, TANKS OR ORGANIZATIONS OF THESE)

RULES: O&F ASSIGN VALUE TO FORCE ELEMENTS.

MEASURES: O(x) AND F(y) ARE THE VALUES ASSIGNED TO x BY O & F.

ARBITRARY NUMBER: "a"

LINEARITY: IF $D(ax+y) = aD(x) + D(y)$ FOR ALL "a" GIVEN x & y.

FIGURE 2

Conversely, it must follow that "larger (or smaller) than the sum" is nonlinearity.

SYNERGISM

ENGLISH: WHOLE IS LARGER THAN SUM OF PARTS

MATHEMATICS: $D(ax+y) \neq bD(x) + D(y)$

GIVEN x & y

FOR SOME $a \neq b$

(NONLINEARITY)

FIGURE 3

Figure 3 shows how we can derive a definition of nonlinearity where the factor "b" defines mathematical parameters for a force multiplier. (i.e. If "b" is greater than "a" then the measure of force is greater than the sum of its parts.) What remains is first to validate this concept against other notions of force multiplication, then we need methods to calculate it if it is to be useful.

OPERATIONAL IMPERATIVES

The issue of nonlinearity in combat is is embedded both in our doctrine and in our investment strategy each of which support the objective of defeating a larger force. The doctrine is based on historical evidence and the investment strategy is based on a notion that our nation has a technology to manage information on the battlefield in such a way that it can make a dramatic difference.

Focus upon operational concepts of combat. Clausewitz discussed three which may interest us.[1] These are center of gravity, lines of operation and culminating points.

The center of gravity is defined as "the hub of all power and movement, on which everything depends."[2] If combat were seen as a biological system, it could be characterized as the most vital of organs in a most complex organism. If it were damaged, imbalances to the entire structure would result in a cascading deterioration in cohesion and effectiveness of the force certainly leaving the force vulnerable to further damage if not resulting in its complete destruction. This is nonlinearity and is a property we must capture in combat models if we are to adhere to doctrine which states that our forces can defeat numerically larger forces. The doctrine is supported by an investment strategy which premises using high technology to create force multipliers. At the tactical level, the center of gravity could be a crucial piece of terrain. At the operational level, it could be the boundary between two of its major combat formations, a vital command and control center, or perhaps its logistical base or lines of communication. At the strategic level, the center of gravity could be a major economic resource or locality, strategic lift or a vital part of the homeland.[3]

Lines of operation connect the force base or bases of operation of a force with its operational objective. A force is said to be operating on interior lines when its operations diverge from a central point and on exterior lines when its operations converge on a central point. Interior lines benefit a weaker force by allowing it to shift the main effort laterally more rapidly than the enemy and this is another example of nonlinearity.

Attacking forces consume energy. Unless strategically decisive, every offensive operation therefore will sooner or later reach a point where the strength of the attacker no longer exceeds that of the defender, and beyond which continued offensive operations therefore risk overextension, counterattack, and defeat. In operational theory, this point is called the culminating point. The art of attack at all levels is to achieve decisive objectives before the culminating point is reached. Conversely, the art of defense is to hasten the culmination of attack, recognize its advent, and be prepared to go over to the the offense when it arrives.[4] This also is nonlinearity.

Use of nonlinearity (particularly catastrophic affects) as a battlefield concept requires extensive knowledge of the enemy's organizational makeup, operational patterns, and physical and psychological strengths. It also requires an appreciation for the dynamic nature of the operational concept, to include changes such as a major shift in operational direction, the replacement of a key commander, or the fielding of new units or weaponry. Clearly there are implications to be considered in both the art and science of operational design. Our interest, however, will be limited to the science.

OPERATIONAL CONCEPTS

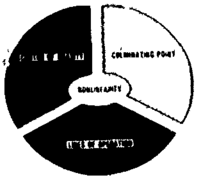


FIGURE 4

The idea of operational concepts being nonlinear is all well and good but eventually we need to face up to the reality of relating these concepts to technically specific procedures. Our emphasis will be on the interaction of information with operations. Indeed, this is the essence of command and control.

COMBAT MODELS

Consider how information is modeled in most combat models now. Simplistically, they have scenarios which provide pre-scripted messages to a weapons allocations module which drives a combat effectiveness module which measures success or failure in terms of attrition, force ratio, or FLOT movement. Such models ignore the effect of message processing to include the role of a decision maker. They fail to consider the degradation of weapons effectiveness because of maneuver. There is no uncertainty and fire fights are treated as continuous functions. Such models are adequate for the trench warfare of World War I, but unacceptable for modern warfare.

THEORY OF COMBAT

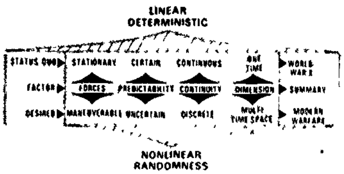


FIGURE 5

In order to correct the paradigm we need to insert a module which converts information from the scenario to messages which are processed into decisions allocating resources with the intent of providing these at the right time and place for decisive outcome on the battlefield. It is the effect of feedback and feedforward loops constrained by time opportunities that causes that complex behavior of the battlefield which interests us.

THE DECISION CYCLE

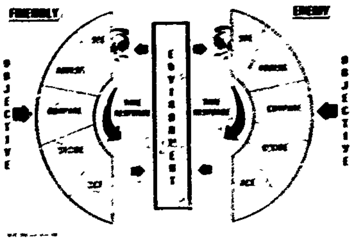


FIGURE 6

One needs to conceive of how information will be used in order to envision any concept of force multiplication. The warfighter has the operational challenge of controlling the battlefield. He follows the paradigm of seeing the environment, deciding on a course of action, then implementing action. Within each of these processes there may be sub-processes, but more importantly, the paradigm is repeated in discrete time steps, since the user is in competition with at least one opponent who is following a similar process, albeit perhaps following a different strategy. Therefore, time is a critical element of the paradigm and the competition drives the dynamics of the process.

Replication of this paradigm on a computer to gain a competitive edge may cause counterintuitive results. This follows from the complexity of the high tech environment.

What is complex? One measurement of complexity is the amount of information an observer requires to describe the object of complexity - the environment or a system. Since this information requirement may exceed what can be put in code for a computer, discrepancies between the computer's and the decision maker's view of the world naturally exist. While the error may be small in discrete calculations, continuous calculations which occur from frequent interation, may cause an explosion in the size of successive errors. Other measurements include predictability (probability of selecting correct future events), entropy (a measurement of disorder created by trying to order other issues), and computability (the amount of computations or time necessary to answer the observers questions).

Another issue with complexity is the importance of perceptions to human observers in making decisions. As Jane Austen once said, "Seldom, very seldom, does complete truth belong to any human disclosure; seldom can it happen that something is not a little disguised, or a little mistaken..."[5]

Never do decision makers have perfect information. It is always incomplete against some measurement. Its incompleteness always involves a time factor - it is always late. Invariably we try to fix these deficiencies by providing more data quicker, only to be faced with the fusion dilemma - creating more data but less information.

The solution to this dilemma may come from shifting focus. People who design information systems seem to measure improvement by the number of bits and bytes of data which can be communicated or stored relative to some operation. Operators focus on plans and resources. However, few focus on the interaction between operations and information, leaving out the element of competition and the role of time in the process.

Constructed to illustrate the dynamics of chaos, Edward Lorentz's classical chaotic waterwheel[9] provides an excellent analog of the relation between information and operations.

WHEEL OF CONFUSION

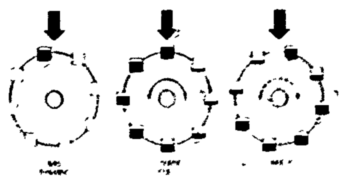


FIGURE 7

In this model, information is similar to the flow of water and operations are similar to the workings of the wheel. The wheel is designed so that buckets fill up from the flow of water and their weight drives revolutions on a descent to the bottom of the wheel.

In order that the drag of the buckets on their ascent do not exceed the drive, they have holes drilled [illegible] the bottom to permit water to drain. From this design we can observe three cases categorized by the flow rate of the waterfall.

In the first case, water will drain from the buckets faster than they fill, consequently the wheel will not revolve - it will not be dynamic, illustrating a decaying quality in the importance of information in our analogy. In the second case, (there is some rate of water fall where the flow rate will exceed the drain rate and "normal" dynamics will occur. In the case of information we need a sufficient amount for dynamics to normally occur, introducing a new concept which we call essentiality. In the final case, we see revolution speeds so high that buckets do not drain fast enough. Consequently, the drag will exceed the drive and the wheel will reverse itself. Further, the wheel will continue to reverse itself in nonperiodic ways (i.e. unpredictable). This is nonlinearity and it may well describe what we have in mind when we say "force multiplicity". Significant is that while this nonlinearity defies intuition, it can be modeled without too much difficulty.

Linearity is normal business for our analytical culture. It suits our human intuition since it means that we can describe events in terms of a succession of activities where output is proportional to input. We have a tendency to believe that if a problem is not linear, it cannot be solved.

But linearity is often the wrong approach in modeling complex problems because it involves a finite number of independent variables and interactions. Linear problems can be sub-divided into sub-problems to be sub-optimized into solutions which can be summed to a global solution. Further, these solutions are time reversible.

Complex problems often involve unlimited information where interactions and outputs are often disproportional to inputs. Such problems are not reducible in that sub-problems cannot be solved within themselves. They require the output of other sub-problems on a real time basis for correct solution. Therefore, sub-optimal solutions cannot be related to a global solution. It becomes clear that the issue evolving is how to solve the nonlinear description of combat.

INFORMATION FLOW ANALYSIS

One way to succeed in this endeavor is to incorporate information flow analyses in combat models. Current models feature a scenario which drives some sort of combat outcome module thru physical performance factors of weapons systems, often by measuring success by comparing attrition, FLOT movement or combat ratios.

An information flow simulation additionally includes a system of links thru which messages are passed and nodes where information is processed. Processed information results in decisions for the allocation of resources. The time value of modeling scarce resources against time opportunity windows is truly the value of what we call command and control. Scenarios provide the script for the initiation of messages. These are entered thru nodes where they are synthesized into information which is assessed, decided upon and implemented thru other messages passed thru links to other nodes for subsequent processing. Certain nodes are linked with resources (close air support (CAS), helicopters, logistics, artillery, other forces) which can be directed to specific points on the battlefield to engage in operations, where results are measured as in current models. In this type of simulation, messages can be lost or late, resulting in wrong perceptions and consequently wrong decisions.

The simulation described above includes feedback and feedforward loops. When designed with sufficient detail to include single sensor nodes and Red side information flow it becomes a complex system in itself. The challenge of exploiting the power of such a model becomes one of describing nonlinearity in layman terms.

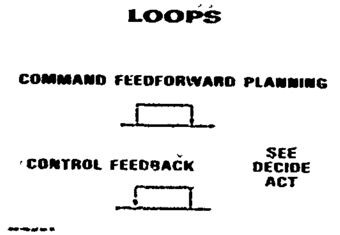


FIGURE 8

Nonlinearity occurs when information is fedback or fedforward in the system faster than a human who thinks he is control can process that information and act upon it.

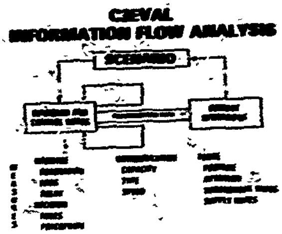


FIGURE 9

What information flow analysis does for us in quantifying force multiplication is give us data points for the value of force capability against whatever force mix we wish to consider. Given these data points we need only substitute them into the relationship shown in figure 3 to calculate "b" - the force multiplier.

NONLINEAR METHODS

While an information flow analysis may give us a measurement of force multiplication without dealing directly with the computational challenges of nonlinearity, a word about breakthoughs in this area is appropriate.

In order to correct the deficiency of using Lanchester first order linear differential equations of one variable (time) to measure the change of resources on the battlefield, a new analytic model based on coupled nonlinear partial differential equations has been proposed to describe the temporal and spatial evolution of opposing forces in combat.[1] These have been generalized further to include a stochastic term permitting the value of random effects to be considered.

The Joint Staff is currently sponsoring work to develop mathematical models using cellular automata.[2] This approach features a modeling environment in which complex phenomena cab be generated from very simple and computationally efficient interaction rules. In short, it is a "top-down" approach where the model operates as a system rather than as a sum of pieces.

One area of focus with cellular automata or any nonlinear system is on the set of attractors which bound the dynamical "flow" of the system. A particularly interesting set of attractors are known as fractals for "fractionally dimensioned".

Fractals can be thought of as nature's attractors since they appear so frequently in natural settings. They have recently gained some publicity for data compression as a high speed alternative for facsimile but they have other interesting properties. One such property is self-similarity which is interesting to command and control research since the decision cycle tends to be self-similar at different levels of organizational hierarchy. The dimension of a fractal is another property of interest. The value of this property is as an indicator of the relative chaos of the system they describe. Conversely they may be a good measure of control in a system. Finally, it has been shown that data relating to fractals tends to be distributed according to the Power Law. This is a useful fact in statistically analyzing fractal structure.

Another interesting field of nonlinear study is catastrophe theory which by its name suggests sort of an ultimate view of nonlinearity. Work done in this area has led to some rudimentary decision aids in the areas of command and control and intelligence analysis.[3] A most useful result of this work is in reducing a number of local variables, largely irrelevant to a particular problem to a few global relevant ones.

SUMMARY

In summary, what we have done in this article is suggest that what we mean when we mention force multiplication is actually nonlinearity. On one hand we observed that war, thru its complexity is nonlinear. This observation is embedded historically in concepts of operational art. On the other hand, we noted that we traditionally have modeled war as a linear, deterministic form and by so doing neglect those properties which are most relevant to the idea of force multiplication. Finally, we introduced the concept of information flow analysis to simulate the role of information and decisions on the battlefield and showed how results from such simulations would permit us to actually calculate force multiplication for specified force mix.

FOOTNOTES:

1 Clausewitz, On War

2 Clausewitz, On War

3 FM 100-5, p 180.

4 Ibid., pp. 181-182.

5 Jane Austen, Emma (1916), Edited by Stephan M. Parrish, p420, W.W. Norton & Company Inc., New York 1972.

6 Edward Lorentz, "Deterministic Nonperiodic Flow," Journal of Atmospheric Sciences 20, (1963) pp 130-41.

7 Protopopescu, R.T. Santoro, J. Dockery, R.L. Cox and J.M. Barnes, Combat Modeling With Partial Differential Equations, ORNL/TM-10636, Nov 1987.

8 Dr. A.E.R. Woodcock, Research into the Applicability of Cellular Automata Mathematics to the Modeling of Combat with Embedded C2, Interim Technical Report, WR-87-TA-00, 22 February 1988.

9 Dr. Alexander E.R. Woodcock, Decision Aids Based on Catastrophe Theory Manifolds, Final Technical Report, WR-86-LP-00, 22 Sep 1987.

THE ELECTRONIC WORKBENCH: AN ENVIRONMENT FOR TESTING THEORIES AND MODELS OF COMBAT

Dr. Alexander E. R. Woodcock [1,2,3,4]

Chief Scientist, Synectics Corporation,
10400 Eaton Place, Fairfax, Virginia 22030

ABSTRACT

The electronic workbench is a prototype computer-based interactive graphics system which can be used by military analysts and decision-makers for combat modeling, situation assessment, decision-making, and battle management involving the application of tactical command and control (C2). Combat parameters and decision threshold conditions are selected by the user, and time-dependent changes in adversarial force strengths are calculated by the system. Computed combat outcomes are displayed in conjunction with the decision space, a structure whose properties are derived in part from catastrophe theory.

1. ELECTRONIC WORKBENCH FACILITIES

The electronic workbench is based on research in mathematical modeling, cognitive engineering, computer graphics, and data acqusition (Figure 1). It can support a wide range of user-selected combat modeling and analysis activities aimed at determining combat outcomes and revealing those circumstances under which non-linear and/or counter-intuitive behavior can occur. If not identified properly, such non-linear behavior can lead to highly misleading and even totally incorrect analyses of a situation of interest. The workbench makes it possible to undertake a range of "thought experiments" in order to explore possibilities and suggest new approaches to solving a range of different types of problems in a computer environment that makes minimal mathematical demands on its users. The workbench can also serve as a testbed for developing and testing theories and models of combat and as a prototype for a fully operational, combat-related, decision-aiding facility.

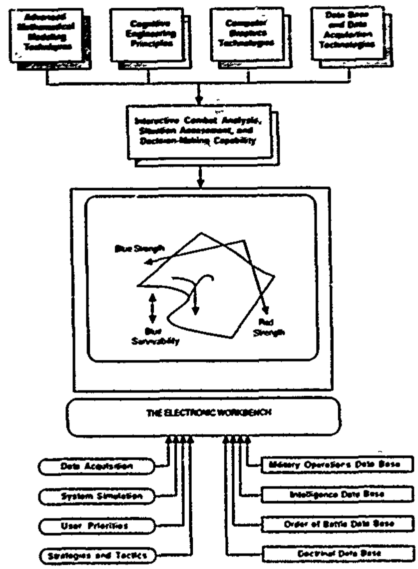


Figure 1: The Electronic Workbench

1.1 MENU-DRIVEN SELECTION

The electronic workbench system uses a menu facility and a series of windows to permit access to a selection of combat models and decision rules and thersholds involving combat termination. They also permit the input of parameter values representing specific combat attrition and reinforcement activities and processes. Available choices include:

1. The names of the adversaries (such as "blue" and "red" forces).
2. The initial troop strengths.
3. The nature of the combat attrition process, which can include the following: aimed firepower, area firepower, and special (smart) weapons firepower (Woodcock and Dockery 1988).
4. The kill uncertainty of each force.
5. The rates of reinforcement.
6. The required duration of the simulation

Force strength computations involving deterministic, stochastic, and probabilistic models are made available by menu-based selection, as described below. During use of the electronic workbench, time-dependent, adversarial-force-strength values are computed on the basis of the input information. Results of this computation are presented on the decision space graphics display. Use of these different combat models and a series of decision strategies can provide an enhanced understanding of the nature of the combat process and the role played by command and control in determining combat outcomes.

1.2 COMBAT MODELS

The electronic workbench permits selection of up to three Lanchester-type attrition models (modern warfare, area fire, and smart weapons fire) and associated attrition and reinforcement coefficients to compute adversarial force strengths as a function of time (Lanchester, 1914; Taylor 1983; and Woodcock and Dockery, 1988). Both deterministic and stochastic versions of the Lanchester-type models are available to the user of the workbench and the results obtained from either of these versions are available on demand.

1. *Modern Warfare:* Lanchester-type modern warfare (or aimed firepower) is described by equations (1) and (2).

$$dx/dt = - a_{mw} y \qquad (1)$$

$$dy/dt = - b_{mw} x \qquad (2)$$

where x and y are the strengths of the opposing forces and a_{mw} and b_{mw} are the modern warfare attrition coefficients. Stochastic versions of the modern warfare attrition equations can be written in the following manner:

$$dx = - a_{mw} y\, dt + e_1\, dw_1 \qquad (3)$$

$$dy = - b_{mw} x\, dt + e_2\, dw_2 \qquad (4)$$

where the terms $(e_1\, dw_1)$ and $(e_2\, dw_2)$ describe the stochastic component of the modern warfare attrition process.

2. *Area Fire:* Area (or unaimed) fire is represented by equations (5) and (6).

$$dx/dt = - a_{af} xy \qquad (5)$$

$$dy/dt = - b_{af} xy \qquad (6)$$

where x and y are the strengths of the opposing forces and a_{af} and b_{af} are the area fire attrition coefficients. Stochastic versions of the area fire attrition equations can be written as:

$$dx = - a_{af} xy\, dt + e_1\, dw_1 \qquad (7)$$

$$dy = - b_{af} xy\, dt + e_2\, dw_2 \qquad (8)$$

where the terms $(e_1\, dw_1)$ and $(e_2\, dw_2)$ describe the stochastic component of the area fire attrition process.

3. *Smart Weapons Firepower:* The model of smart weapons (or special) firepower is based on the assumption that the rate of attrition for one force is proportional to the square of the strength of its adversary (Woodcock and Dockery 1988). Deterministic smart weapons attrition is represented by equations (9) and (10):

$$dx/dt = - a_{sw} y^2 \qquad (9)$$

$$dy/dt = - b_{sw} x^2 \qquad (10)$$

where x and y are the strengths of the opposing forces and a_{sw} and b_{sw} are the smart weapons attrition coefficients. Stochastic versions of the smart weapons fire attrition equations can be written as:

$$dx = - a_{sw} y^2\, dt + e_1\, dw_1 \qquad (11)$$

$$dy = - b_{sw} x^2\, dt + e_2\, dw_2 \qquad (12)$$

where the terms $(e_1\, dw_1)$ and $(e_2\, dw_2)$ describe the stochastic component of the smart weapons attrition process.

1.3 THE DECISION SPACE

The electronic workbench provides a framework to support the activities of analysts and decision-makers through the graphical display of system inputs and outputs in relation to a construct called the decision space (Figure 2). The initial force ratios and changes in these values caused by attrition and reinforcement described by the selected combat models are displayed within the decision space window. The decision space also displays a series of lines drawn to represent the selected combat-termination, decision rules and thresholds.

The pattern of movement on the decision space represents the "ebb and flow of combat" and crossing a decision line indicates necessary conditions for combat termination. Visual inspection of the pattern of combat provided by the decision space can alert the user to the need to modify force deployment strategies or to select different decision methods and/or thresholds, for example.

The decision space is based on a version of the control space of catastrophe theory (Poston and Stewart, 1978; Thom, 1975; Woodcock and Davis, 1978; Woodcock and Poston, 1974; Zeeman, 1977; and Zeeman and Trotman, 1976). This space provides a representation of the strength of two opposing (blue and red) forces in terms of a position on a plane specified with reference to the two conflicting factor axes of a catastrophe function.

Catastrophe theory has been used to develop models of several military systems and its use in the workbench is a logical extension of these activities. Two- and four-control-factor models of military behavior, which are based on the cusp and butterfly catastrophes, respectively, have been developed by Woodcock and Dockery (1984, 1988). Following this work, Dockery and Chiatti (1986) have used a statistical program developed by Cobb (1978, 1981, 1983) to fit simulated combat data to the surface of the cusp catastrophe manifold. Woodcock, Langendorf, and Cobb (1988) and Langendorf and Woodcock (1988) have developed a new methodology, based on statistical catastrophe theory (Cobb, 1978, 1981, 1983) to analyze and interpret the perceptions of military threat by intelligence analysts.

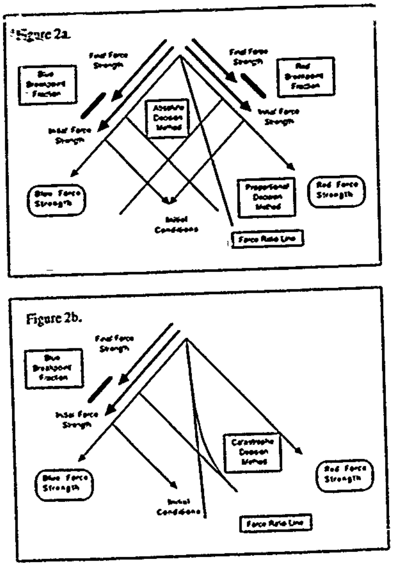


Figure 2 The Decision Space

1.4 DECISION RULES

The electronic workbench supports the selection and graphical display of different types of decision rules and thresholds to reflect different types of user-defined combat strategies and dogmas. Three basic classes of rules (called absolute, proportional, and catastrophe decision rules) are available.

1. *Absolute decision rules* specify that combat termination will occur after a given percentage reduction in the strength of one force (the "blue" force, for example) without reference to the changes in the strength of the opposing ("red") force and thus are "one-sided decision rules" (Figure 2a).

2. *Proportional decision rules* specify that combat termination would take place when the ratio of the opposing ("blue" and "red") forces reaches a particular value and thus are "two-sided decision rules" (Figure 2a).

3. *Catastrophe decision rules* capture features of both absolute and proportional decision rules and are represented graphically by a line which is topologically equivalent in shape to the shape of the bifurcation set of the cusp catastrophe (Figure 2b).

Absolute, proportional, and catastrophe decision methods and numerical thresholds for combat termination for each of two ("blue" and "red") adversarial forces can be selected. Similar decision rules for both force combat termination conditions (such as: catastrophe-catastrophe, absolute-absolute, or proportional-proportional) reflect the use of similar combat strategies or dogmas by the "blue" and "red" force decision-makers. However, selection of different rules (proportional-catastrophe; absolute-catastrophe; or catastrophe-absolute, for example) reflects the use of different strategies by these decision-makers.

Selection of particular sets of decision rules can have a profound impact on the nature of the combat outcome, as we will now demonstrate. The Initial Forces-type of display (see below) represents the initial and final combat conditions as a point and an ellipse, respectively. The lengths of the axes of the ellipse represent the degree of kill uncertainty for the two forces so that the ellipse can be thought of as a form of "error ellipse" (Figure 3). We will refer to the ellipse as the *kill uncertainty ellipse*. The likelihood that a red force decision-maker, for example, would recommend breaking off an engagement is proportional to the ratio of the areas A and B (area A / area B) on either side of the red force decision line. Relatively larger values of this ratio reflect a higher likelihood of combat termination.

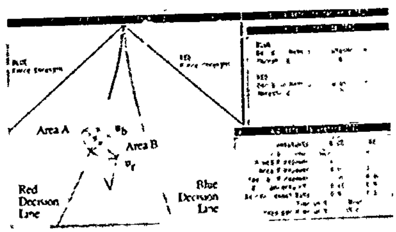


Figure 3 Initial Forces Display

In Figure 4a, both decision-makers have chosen catastrophe-type decision rules while in Figures 4b and 4c both of them have chosen absolute-type and proportional-type rules, respectively. The information presented in these figures suggests that the choice of catastrophe-type decision rules by both forces, would produce combat termination likelihood values for the red and blue forces of about thirty and zero percent, respectively (Figure 4a). By contrast with absolute-type rules, the likelihood of combat termination for the red forces would increase to some seventy percent while remaining at zero percent for the blue forces (Figure 4b). With proportional rules, the likelihood of combat termination is some twenty- and zero-percent for the red and blue forces, respectively (Figure 4c).

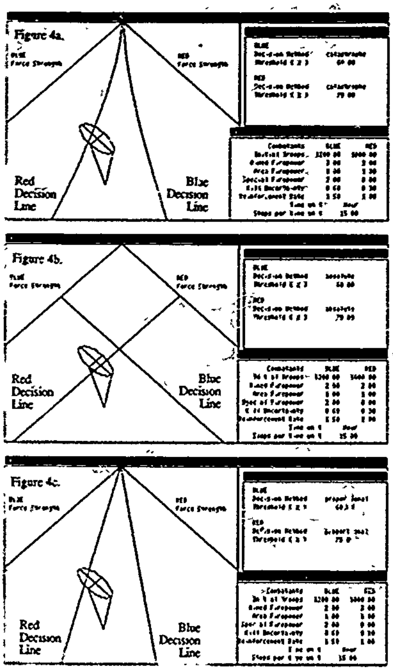


Figure 4: Similar Decision Rules

The impact of selecting a mixture of combat termination decision rules is shown in Figure 5. Blue uses a catastrophe rule and red an absolute rule in Figure 5a while blue uses an absolute rule and red a catastrophe rule in Figure 5b. Selection of such mixed combat termination rules can reflect the adoption of different combat termination dogmas by the blue and red forces and it is obvious from an inspection of Figure 5 that such differences in decision-making methods can lead to marked differences in combat outcomes.

2. USING THE ELECTRONIC WORKBENCH

The electronic workbench is a highly interactive device which supports a range of combat analyses involving deterministic, stochastic, and probabilistic processes, as well

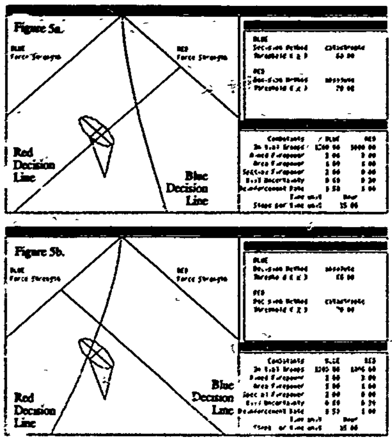


Figure 5: Mixed Decision Rules

as decision-making activities involving absolute, proportional, and catastrophe-types of decision rules. Available display options include the following: Non-Random Paths, Random Paths, Initial Forces, Cascading Paths, Probability Flow, and Vector Field displays, and are described below.

The decision space displays the results of the modeled combat process in terms of the position of coordinates representing the strengths of the opposing ("red" and "blue") forces. These coordinates trace out a path on the decision space as the combat analysis proceeds. Monitoring of the pattern of change reflected in the nature of such paths provides information on the changing combat environment and on the effectiveness of the opposing ces. Movement of the path towards one or other of decision lines provides a warning of the possibility of combat termination and other types of sudden changes so that the workbench can serve as the basis of an indications and warnings facility for military analysts.

2.1 NON-RANDOM PATHS

The Non-Random Paths display presents the results of integrating deterministic versions of the selected combat equations with the assigned force strength and attrition coefficent values for "red" and "blue" forces. The results of this integration are displayed as a curved line in Figure 6 which, in this example, crosses the red force decision line. Under such circumstances, a red force decision-maker would have opted for a termination of the simulated engagement. If such a termination did not occur, the engagement would have continued until the red force was totally destroyed while significant numbers of the blue force remained.

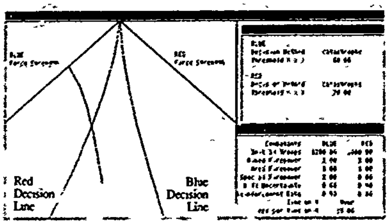


Figure 6 Non-random Display

2.2 RANDOM PATHS

The Random Paths display considers the impact of uncertainty on the combat process. It presents one hundred separate integrations of the combat equations based on the assumption of stochastic or random behavior related to the selected levels of kill uncertainty for each of the ("blue" and "red") forces (Figure 7). Consideration of the impact of kill uncertainty can have a profound effect on the combat outcome. Identical combat conditions to those described in Section 2.1, (in which the only result is a blue "victory") with the addition of the effect of kill uncertainty can lead to results in which the red force can overcome the blue force in a significant number of cases. Such a result was not predicted from the deterministic models used in the Non-Random Paths analysis and this study reveals the importance of developing combat models which include uncertainty in order to reduce the possibility of unexpected combat outcomes.

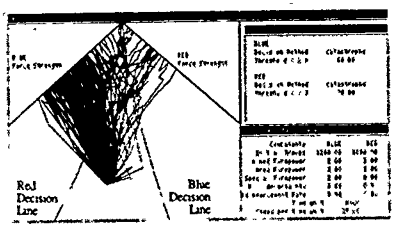


Figure 7 Random Display

2.3 INITIAL FORCES

In the initial forces type of display (Figure 3), the initial and final force strengths of the adversarial ("blue" and "red") forces are represented by the points a and b on the decision space diagram, for example. The kill uncertainty ellipse centered on the combat end-point (point b) is constructed to represent the uncertainty associated with the attrition of the "blue "and "red" forces with the length of the axes of the ellipse (u_b and u_r) proportional to the kill uncertainties of these two forces. The higher the level of uncertainty, the greater the area of the ellipse. As mentioned above, the kill uncertainty ellipse can be used to assess the likelihood of combat termination. Thus, the ratio of the areas of this ellipse located on either side of a decision line can serve as an estimate of the likelihood that a decision-maker would decide to terminate combat.

2.4 CASCADING PATHS

The Cascading Paths display is an extension of the Initial Forces display described in Section 2.3. It is generated by drawing the kill uncertainty ellipse for a single time step of the integrative process and using the positions of the ends of the axes of this ellipse as the starting conditions for a subsequent integrations. In this display, integration is carried on for ten iterations (Figure 8) and the tree-like structure that is produced can be considered to represent the "flow" of uncertainty associated with the simulated combat environment. Identical combat conditions to those described in Section 2.1 were used to construct both the Initial Forces and Cascading Paths displays presented in this paper. Combat termination with the blue force dominating the red force is the only outcome of deterministic combat with these selected parameters. The significant numbers of the paths which cross the blue force decision line (representing conditions for possible blue force combat termination) in Figure 8 is another manifestation of the impact of kill uncertainty on combat outcomes.

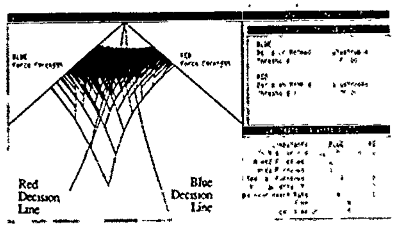


Figure 8 Cascading Paths Display

2.5 VECTOR FIELD

The Vector Field display (Figure 9) consists of a series of line segments drawn to reflect the result of combat attrition and reinforcement with initial force ratios represented by the starting points of these vectors. The magnitude and direction of these vectors represent the pattern of combat activity with these starting conditions. This type of display can provide a decision-maker with an estimate of the magnitude of the changes in force ratio that would be required to achieve a particular combat outcome, for example.

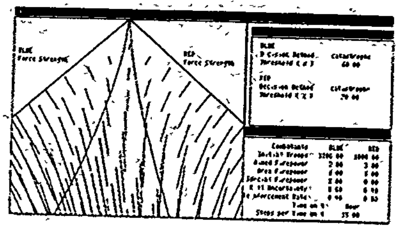


Figure 9: Vector Field Display

2.6 PROBABILITY FLOW

The Probability Flow display (Figure 10) represents the evolution of combat in terms of the pattern of movement of a series of points whose density represents the probability of the existence of a particular force ratio. A higher degree of aggregation of the points on the decision space represents a higher level of force ratio probability. At the start of the combat process, these points are located in a dense rectangle centered on a point on the decision space whose position is determined by the initial force strengths. Combat attrition and reinforcement activities lead to changes in the force ratio probabilities which are reflected in a change in the density of the points on the decision space. Points which cross the decision lines are considered to represent combat events which would be terminated due to the selected decision rules and thresholds. The percentage of points (representing probability distributions) attributed to the blue and red forces as the result of crossing these decision lines can serve as a measure of the relative success of these forces during the combat engagement.

Analysis of the Probability Flow display can be very instructive. Thus, with the combat conditions selected in Figure 10a, a red force achieves an early advantage of 17 percent (in favor of this force) to zero percent (in favor of a blue force) after some 0.0164 time units have elapsed (Figure 10b) and the blue force decision-maker might seriously consider terminating the engagement. However, if the simulation is permitted to continue toward completion, it becomes clear that the blue force is much more likely to be victorious as it eventually achieves an advantage of 77 percent to 17 percent (in favor of the red force) with some 6 percent of the combat engagements remaining undecided after some 0.2554 time units have elapsed (Figure 10c). Therefore, all the blue force attrition appears to have occurred during the initial part of the combat process and all subsequent attrition then occurs to the red force in this example.

3. SUMMARY

The electronic workbench consists of a computer-based interactive graphics system which can be used by military analysts and decision-makers to support such tasks as situation assessment, decision-making, and battle management which involve the application of tactical C2.

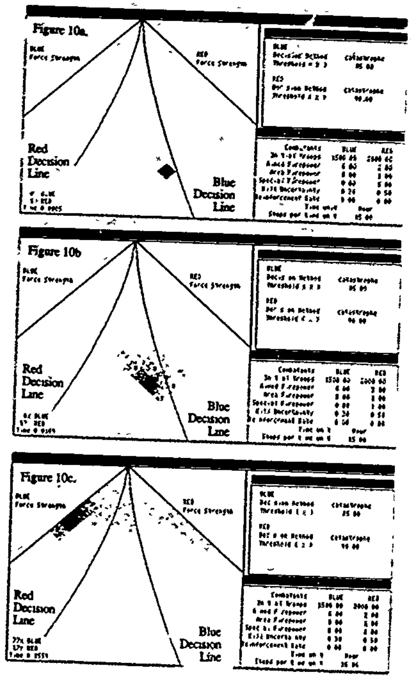


Figure 10: Probability Flow Display

The workbench permits the performance of a wide range of user-selected mathematical and other processing operations through the use of a menu facility.

In particular, the workbench permits the use of mathematical techniques to explore the potential non-linear nature of situations in order to reveal the possible occurrence of non-linear and/or counter-intuitive behavior. If not identified properly, such non-linear behavior can lead to highly misleading and even totally incorrect analyses of a situation of interest.

By providing an interactive graphics display the workbench permits the user to undertake a range of "thought experiments" in order to explore possibilities and suggest new approaches to solving a range of different types of problems. Furthermore, the workbench has been designed so that the advanced mathematics upon which it is based is transparent so that minimal mathematical demands are made on its users.

4. BIBLIOGRAPHY

Cobb, L., 1978. "Stochastic Catastrophe Models and Multimodal Distributions." *Behavioral Science*, 23: 360-374.

Cobb, L., 1981. "The Multimodal Exponential Families of Statistical Catastrophe Theory." In: *Statistical Distributions in Scientific Work*. Taille, C. et al. (eds.) 4: 67-90.

Cobb, L., 1983. "A Maximum Likelihood Computer Program to Fit a Statistical Cusp Hypothesis." The Hague, The Netherlands: SHAPE Technical Centre.

Dockery, J. T. and S. Chiatti, 1986. "Application of Catastrophe Theory to the Problems of Military Analysis." *European Journal of Operations Research*, 24: 46-53.

Dockery, J. T., and A. E. R. Woodcock, 1988, "Models of Combat with Embedded C2. II: Catastrophe Theory and Chaotic Behavior." *International C. I. S. Journal*. In Press.

Lanchester, F. W., 1914. "Aircraft in Warfare: The Dawn of the Fourth Arm; No. V:, The Principles of Concentration." *Engineering*. 98: 422-423.

Langendorf, P. and A. E. R. Woodcock, 1988. "An Application of Catastrophe Theory to C^3I Problems. In: *The Proceedings of the Second Joint Diorectors of Laboratories Command and Control Research Symposium*, Monterey, California. Washington, D.C.: National Defense University.

Poston, T. and I. Stewart, 1978. *Catastrophe Theory and its Applications*. London: Pitman.

Taylor, J. G., 1983. *Lanchester Models of Warfare*, I, II. Alexandria, Virginia: Operations Research Society of America, 1983.

Thom, R., 1975. *Structural Stability and Morphogenesis* Reading, Massachusetts: W. A. Benjamin.

Woodcock, A. E. R. and M Davis, 1978. *Catastrophe Theory*. New York: E. P. Dutton.

Woodcock, A. E. R. and J. T. Dockery, 1984. "Application of Catastrophe Theory to the Analysis of Military Behavior." The Hague, The Netherlands: SHAPE Technical Centre Consultants Report, STC CR-56.

Woodcock, A. E. R. and J. T. Dockery, 1988. "Models of Combat with Embedded C2. I: Catastrophe Theory and the Lanchester Equations." *International C.I S. Journal*. In Press.

Woodcock, A. E. R., P. Langendorf, and L. Cobb, 1988. "Perceptual Hysteresis: You Can't Get There From Here." *Signal*, April: 51-57.

Woodcock, A. E. R., and T. Poston, 1974. *A Geometrical Study of the Elementary Catastrophes. Lecture Notes in Mathematics*. 373. Berlin and New York: Springer-Verlag.

Zeeman, E. C., 1977. *Catastrophe Theory, Selected Papers 1972-1977*. Reading, Mass.: Addison-Wesley.

Zeeman, E. C., and D. Trotman, 1976, "The Classification of Elementary Catastrophes ≤ 5." In: *Structural Stability, The Theory of Catastrophes and Applications. Lecture Notes in Mathematics*. 525: 263-327. Berlin and New York: Springer-Verlag.

---

1. Supported in part by Contract Number MDA903-86-C-0366 to Synectics Corporation.

2. The author is very grateful for the continuing interest in and support of this work by Dr. John Dockery, The Organization of the Joint Chiefs of Staff (OJCS), The Pentagon, Washington, D. C.

3. The author would like to acknowledge the support provided by Dr. Loren Cobb (Medical University of South Carolina, Charleston, South Carolina) in the development of the electronic workbench.

4. The views, opinions, and findings contained in this paper are those of the author and should not be construed as an official OJCS position, policy, or decision, unless so designated by other official documentation.

DYNAMIC WEAPON-TARGET ASSIGNMENT PROBLEMS WITH VULNERABLE C^2 NODES

Patrick A. Hosein, James T. Walton and Michael Athans

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

In this paper we present a progress report on our work on the dynamic version of the Weapon to Target Assignment (WTA) problem and on the static version of the WTA problem in which vulnerable C^3 nodes are included in the formulation.

In the static WTA problem, weapons must be assigned to targets with the objective of minimizing the total expected number (or value) of the surviving targets. In the dynamic version, this allocation is done in time stages so that the outcomes of previous engagements can be used in making future assignments. We will show that, for the simple cases studied, there is a significant cost advantage in using the dynamic strategy. We believe that similar results will hold for the more general problem.

In the static defense-asset problem with vulnerable C^3 nodes the offense is allowed to either attack the assets themselves or to first attack the command and control system, and then the assets; if the C^3 nodes are destroyed then the defensive interceptors are assumed unusable. We first consider simple cases where assumptions are made as to offensive and defensive states of knowledge and kill probabilities. Strategies are then developed and optimal weapon allocations identified. These assumptions are then relaxed, and further examples demonstrate the ensuing complexity.

1. INTRODUCTION

Our long-range research objective is the quantitative study of the theory of distributed C^3 organizations. Our present research has been concentrated on certain aspects of situation assessment and resource commitment.

Situation assessment entails the use of sensors to detect and track the enemy and its weapons (i.e missiles, tanks etc.). These sensors are usually geographically distributed so that distributed algorithms are desirable. This problem can be formulated as a distributed hypothesis testing problem. Results on this research can be found in the paper by Papastavrou and Athans [1] in these proceedings.

The resource commitment problem deals with the optimal assignment of the defense's resources against the offense's forces so as to minimize the damage done to the defense's assets. If the battle is such that the defense has a single opportunity to engage the enemy then the problem can be formulated as a static resource allocation problem. If multiple engagements are possible (as for example in the Strategic Defense System (SDS) scenario) then better use can be made of the defense's resources by assigning them dynamically (i.e. observe the outcomes of some engagements before making further assignments). This is sometimes called a shoot-look-shoot strategy in the literature. In this paper we will provide some results on simple cases of the dynamic problem and make comparisons with the corresponding static problem.

The above resource allocation problem will typically be solved at a C^3 node and the results transmitted to the relevant resources. These C^3 nodes will therefore be of vital importance to the defense since their destruction will in effect paralyze the resources over which they have control. One approach, that can be used to increase the reliability of the system, is to replicate the C^3 nodes. In this way destruction of the primary C^3 node does not affect the defense's system since its function can then be performed by one of the "backup" C^3 nodes. We have formulated a model which includes these replicated nodes and will provide results on simple cases of the problem.

This paper is in effect a progress report of our work on the resource allocation problem and on the survivability issue mentioned in the previous paragraph. The models being used are rich enough to capture the nature of the mission (e.g. defense of assets), enemy strength (number and effectiveness of the enemy's weapons), defense strength (number and effectiveness of the defense's weapons) and strategy and tactics (preferential defense, shoot-look-shoot, etc.) It should be noted that basic research studies on these topics are virtually non-existent.

Our work is motivated by military defense problems, two examples of which are as follows. The first example involves the Anti-Aircraft Weapon (AAW) defense of the Naval battle group or battle force platforms. The assets being defended are aircraft carrier(s), escort warships and support ships each of which is of some intrinsic value to the defense. The threat to these assets are enemy missiles launched from submarines, surface ships and aircraft. These missiles may have different damage probabilities which depend on the missile type, target type, etc. The defense's weapons are different types of AAW interceptors launched from Aegis and other AAW ships. The kill probability of these weapons may also depend on the specific target-interceptor pair. The objective of the defense is to maximize the expected surviving value of the assets. The problem is to find which AAW interceptors should be assigned to each of the enemy missiles, when should they be launched and why. This formulation allows for a preferential defense where, in a heavy attack, it may be optimal for the defense to leave "low" valued assets undefended and concentrate its resources on saving the "high" valued assets.

The second motivating example for our work is the midcourse phase of the Strategic Defense System. In this case the assets are our (the defense's) Inter-Continental Ballistic Missile (ICBM) silos, military installations, C^3 nodes, populations centers, etc. The threat to these assets are enemy re-entry vehicles (RV's), surrounded by decoys. The defense's weapons are Space-based kinetic-kill vehicles (SBKKV's) and ERIS interceptors. The objective of the defense is the maximization of the expected surviving value of the assets. The problem is the determination of the optimal weapon-target assignments and the timing of the interceptor launches.

2. THE STATIC TARGET-BASED WTA PROBLEM

In this section we will present the static version of the target-based WTA problem. In this model, a number of missiles (the targets) are launched by the offense. The defense has a number of interceptors (the weapons) with which to destroy these missiles. The defense assigns a value to each of the targets based on factors such as target type, point of impact, etc. Associated with each weapon-target pair is a kill probability which is the probability that the weapon will destroy the target if it is fired at it. We will make the assumption that the action of a weapon on a target is independent of all other weapons and targets. The problem faced by the defense is the assignment of weapons to targets with the objective of minimizing the total expected surviving target value.

2.1 Mathematical Statement of the Static WTA Problem

The following notation will be used:

N = the number of enemy targets
M = the number of defense weapons
V_i = the value of target i
p_{ij} = probability that weapon j kills target i if shot at it

The solution will be represented by:

$$x_{ij} = \begin{cases} 1 & \text{if weapon j is assigned to target i} \\ 0 & \text{otherwise} \end{cases}$$

The optimization problem can now be stated as:

$$\min_{\{x_{ij}\in\{0,1\}\}} \quad J = \sum_{i=1}^{N} V_i \prod_{j=1}^{M} (1-p_{ij})^{x_{ij}} \qquad (2.1)$$

$$\text{subject to:} \quad \sum_{i=1}^{N} x_{ij} = 1, \quad j = 1,2,\ldots,M.$$

The objective function is the total expected value of the surviving targets while the constraint is due to the fact that each weapon can attack only one target.

2.2 Comments on the Solution of the Static WTA Problem

This problem has been proven, by Lloyd and Witsenhausen [2], to be NP-Complete in general. This means that polynomial time algorithms for obtaining the optimal solution do not exist. One must therefore resort to sub-optimal algorithms.

In the special case in which the kill probabilities are independent of the weapons, denBroeder et al. [3], have proposed an optimal algorithm for the problem which runs in polynomial time. This algorithm, which is usually called the Maximum Marginal Return algorithm, works by assigning the weapons sequentially with each weapon being assigned to the target which results in the maximum marginal return in the objective function.

In the special case in which the kill probability is the same for all weapon-target pairs and all targets have the same value, the optimal assignment is obtained by dividing the weapons as evenly as possible among the targets.

3. THE DYNAMIC WTA PROBLEM

The battle scenario for the dynamic problem is the same as for the static problem except that the weapon assignments are done in time stages each of which consists of the following steps:

(a) Determine which targets have survived the last engagement,
(b) Assign and fire a subset of the remaining weapons with the objective of minimizing the total expected value of the surviving targets at the end of the final stage.

We have looked at some simple cases of this problem to gain insight into the general problem and to help bolster our intuition.

3.1 The Two-Target Case

In this case we will assume that there are two targets ($N=2$), the kill probability is the same for all weapon-target pairs, the defense has M weapons and there are K time stages. With these assumptions we have proved the following theorem:

Theorem 3.1 *Under the assumptions given above, an optimal strategy for the present stage can be found as follows. Let* $T = \lceil M/K \rceil$. *Solve the corresponding static problem with* T *weapons and denote the solution by* $\{x_i\}$ *where* x_i *is the optimal number of weapons to be assigned to target* i. *The optimal assignment for the present stage of the dynamic problem is to assign* x_i *weapons to target* i.

If we further assume that $V_i=1$ and that M is divisible by 2K then it can be shown that the optimal target leakage J(K) is given by:

$$J(K) = 2K(1-p)^{M(1-\frac{1}{2K})} - 2(K-1)(1-p)^{M}. \quad (3.1)$$

Note that if K is large then the optimal leakage $J(K) \approx 2(1-p)^M$ while the optimal leakage for the corresponding static problem is $2(1-p)^{M/2}$. In other words, roughly *half* as many weapons are required for the dynamic strategy to obtain the same expected leakage as that of the static strategy. This expresses, in a convenient form, the value of the battlespace to the defense.

In figure 1 we have plotted the ratio of the K-stage leakage to the static leakage versus the kill probability p for the case of two targets and 16 weapons. We have plotted the cases K = 2, 4 and 8. First note that the leakage advantage of the dynamic strategy over the static one increases with the kill probability. This is due to the fact that as the kill probability increases, the information gained from the first stage increases. This implies that more effective use will be made of (costly) high accuracy weapons if the dynamic strategy is used. Next, note that the leakage advantage of the dynamic strategy increases with the number of stages. This is due to the fact that the information gained increases with the number of stages. Note, however, that most of the improvement is obtained after only a small number of stages. In other words, the curves rapidly converge as the number of stages increases. This implies that, even if the number of stages is small, the dynamic strategy offers a significant leakage advantage.

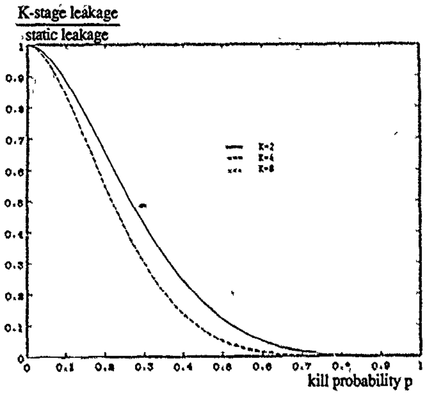


Figure 1 : J(K)/J(1) vs. p for the case M=16 and N=2

Figure 2 contains plots of the ratio of the two-stage leakage to the static leakage versus the kill probability p for the case of two targets. We have plotted the cases for which M = 4,8,12,16 and 20. Note that the leakage advantage of the dynamic strategy increases with the number of weapons used. Note also that the same leakage advantage can be obtained by either using a few high accuracy weapons or many low accuracy weapons.

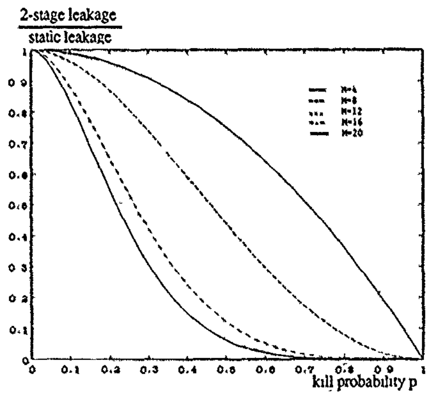


Figure 2 : J(2)/J(1) vs. p for the case K=2 and N=2

3.2 The Two-Stage, N-Target Case

In this section we will consider the two-stage dynamic problem with N equally valued targets (each of value 1), a single kill probability p (for all stages and weapon-target pairs), and M weapons. Unlike the two target problem, we were unable to find an analytical solution to this problem for the case N>2. However, we were able to derive useful properties of the optimal solution. The first property, which holds for the more general problem of K stages, is given as theorem 3.2.

Theorem 3.2 *The optimal solution to the problem given above has the property that the weapons to be used at each stage are spread evenly among the surviving targets.*

The above result simplifies the problem to be solved. Let us consider the two-stage problem. Let M_2 denote the number of weapons used in the first stage (i.e. 2 stages to go). The number of weapons that will be used in the last stage will then be $M-M_2$. Denote the corresponding cost of the assignment, in which weapons are spread evenly at each stage, by $J(M_2)$. The optimal solution can then be obtained by minimizing $J(M_2)$ over the set $\{0,1,...,M\}$.

If $J(M_2)$ happened to have a "convex" shape (i.e have the property that $J(M_2+1) - J(M_2) \geq J(M_2) - J(M_2-1)$) then the above minimization could be done efficiently. Unfortunately, we can show (by example) that this is not the case.

Let us now consider the case of K stages, N equally valued targets, a single kill probability and M weapons with the constraint that the number of weapons is less than the number of targets. Our intuition tells us that a dynamic allocation should not perform any better than a static allocation. This is in fact the case. This result is given in the following theorem.

Theorem 3.3 *Under the assumptions given above, a dynamic allocation cannot perform any better than a static allocation. Hence it is optimal to assign all of the weapons in the first stage.*

The above theorem is not particularly enlightening but it allows us to concentrate on the cases in which M>N. Let us now consider the case in which there are two stages, N targets each of unit value, a single kill probability p and M weapons with M > N. As before, let M_2 denote the number of weapons used in the first stage of a strategy.

Theorem 3.4 *Under the assumptions given above, the optimal assignment has the property that* $M_2 \geq N$.

We conjecture that the above theorem can be extended to the case of more than two stages.

Using the properties given above we computed the optimal solution to the two stage dynamic problem for various numbers of weapons and targets using a kill probability of p = 0.9. The optimal values of M_2 are given in table 1. The number of targets varies from 1 to 10 (the columns) and the number of weapons varies from 1 to 25 (the rows). In the cases where the solution is non-unique we have chosen values of M_2 which exemplify any patterns. Note that for the cases $N \leq M \leq 2N-1$ the optimal value of M_2 is N. We conjecture that this result holds in general but have so far failed to find a proof.

M	N=1	N=2	N=3	N=4	N=5	N=6	N=7	N=8	N=9	N=10
1	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
2	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
3	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
4	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
5	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
6	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
7	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
8	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
9	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
10	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
11	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
12	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
13	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
14	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
15	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
16	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
17	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
18	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
19	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
20	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
21	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
22	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
23	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
24	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]
25	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]

Table 1 : Optimal values of M_2 for N=1:10, M=1:25, p=0.9

Figure 3 contains a plot of the ratio of the 2-stage leakage to the static leakage versus the kill probability p with a 2:1 weapon to target ratio (i.e. M = 2N). We have plotted the cases N = 2,4,6,8 and 10. Here we see that as the size of the problem increases the leakage advantage of the dynamic model increases. This implies that, for the problems that concern us (i.e. large scale problems), the dynamic model has a significant leakage advantage over the static model.

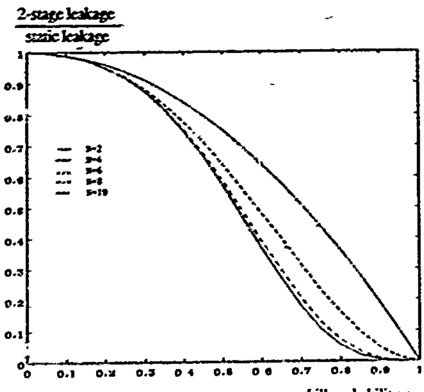


Figure 3 : J(2)/J(1) vs. p for the case K=2 with M=2N

Figure 4 contains a plot of the ratio of the 2-stage leakage to the static leakage versus the number of weapons M with a kill probability p = 0.5. We have plotted the cases N = 2,4,6,8 and 10. Note that the leakage advantage of the dynamic model increases roughly exponentially with the number of weapons. This implies that the dynamic model is significantly better even for relatively small weapon to target ratios.

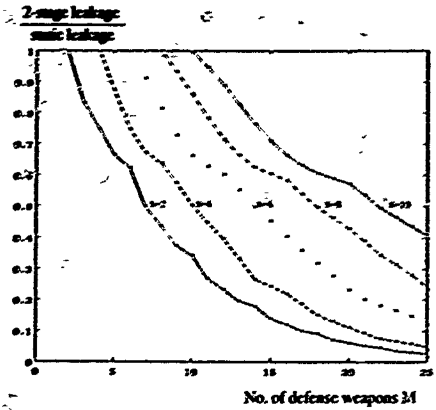


Figure 4 : J(2)/J(1) vs. M for the case K=2 with p = 0.5

4. DISTRIBUTION OF THE C^3 FUNCTION

4.1 Introduction

Although distribution of C^3 functions is generally regarded as desirable, little quantitative insight exists as to the resulting gains. Enhanced survivability of the BM/C^3 system is typically cited as the reason for such distribution. Here we develop a concept of vulnerability for the WTA function and consider its effect on allocation strategies. The underlying paradigm is the defense asset model, in which the defense defends a collection of differently valued assets against attack.

In such an engagement the objective of the offense is to minimize expected surviving defense asset value. This can be accomplished by either attacking the assets themselves or by first attacking the defense system, and then the assets. Although various components of the defense system are vulnerable we assume here that the objective is the command and control centers, the destruction of which renders the defensive stockpile unusable. A precursor attack increases the likelihood that the assets will be undefended and thus increases their vulnerability, making this a potentially attractive offensive strategy. Increased vulnerability, however, comes at the cost of a reduction in the offensive stockpile available to attack assets.

The overall objective of the defense is to maximize expected surviving asset value; BM/C^3 nodes exist and are defended only insofar as they further this. Unless some redundancy is planned, a destroyed BM/C^3 node causes the defensive weapons it controls to be useless. For the purposes of initial analysis, we assume that command and control is centralized, and replication of function will occur at this level. The defense has the potential to increase expected surviving value by allocating some portion of its stockpile to BM/C^3 defense, but in so doing reduces the number of weapons available for defending assets.

4.2 Problem Definition

We consider a set of Q defense assets each of value v_q to be defended, and a set of T identical BM/C^3 nodes to do so. Assets and C^3 nodes are far enough apart so that a successful attack on one does not affect any other. The defense can see which assets or nodes are being engaged in time to intercept the attack, if desired, but cannot adaptively change weapon assignments based on damage assessment.

The defense has a stockpile of M interceptors and the offense a stockpile of N missiles. Each side knows these quantities. Interceptors are in a central stockpile, and are thus capable of defending any C^3 node or asset. Initially we assume that all attacking missiles have the same probability of kill, p, and defending interceptors r. Thus the miss probability for an unintercepted target (e.g. attacking missile) is $(1\text{-}p)$, if intercepted by j defenders $(1\text{-}p(1\text{-}r)^j)$ and for i such intercepted targets directed at the same asset $(1\text{-}p(1\text{-}r)^j)^i$. Were kill probabilities dependent on weapons, targets and assets this would become (for a given asset q)

$$\prod_{i=1}^{N_q}[1\text{-}p_{qi}\prod_{j=1}^{M}(1\text{-}r_{ij})^{x_{ij}}] \qquad (4.1)$$

where $x_{ij} = 1$ if weapon j is assigned to target i and 0 otherwise, r_{ij} = kill probability for interceptor j fired at target i, p_{qi} = kill probability for target i against asset q and N_q indexes the set of targets aimed at asset q. The simpler version is

$$\prod_{i=1}^{N_q}[1\text{-}p(1\text{-}r)^{x_i}] \qquad (4.2)$$

where x_i is the number of defensive weapons assigned to the i-th target.

The objective function used is

$$U = \xi V_u + (1\text{-}\xi)V_d \qquad (4.3)$$

where ξ = probability that all C^3 nodes are destroyed, V_u = expected surviving asset value if undefended and V_d = expected surviving asset value if defended. The offense, then, seeks to minimize V and the defense to maximize it. In general,

$$V_d = \sum_{q=1}^{Q}V_q\prod_{i=1}^{N_q}[1\text{-}p_{qi}\prod_{j=1}^{M}(1\text{-}r_{ij})^{x_{ij}}] \qquad (4.4)$$

With the above assumptions about kill probabilities, this reduces to

$$V_d = \sum_{q=1}^{Q}V_q\prod_{i=1}^{N_q}[1\text{-}p(1\text{-}r)^{x_i}] \qquad (4.5)$$

The expected number of surviving assets, if undefended, is

$$V_u = \sum_{q=1}^{Q}V_q(1\text{-}p)^{N_q} \qquad (4.6)$$

The probability that all BM/C^3 nodes are destroyed, using simplified probabilities, is

$$\xi = \prod_{t=1}^{T}\{1 - \prod_{s=1}^{a_t}[1 - p_c(1\text{-}r)^{y_s}]\} \qquad (4.7)$$

where a_t and y_s are offensive and defensive allocations for the precursor attack, t indexes the C^3 nodes and p_c is the offensive kill probability for C^3 nodes. We assume that all BM/C^3 nodes must be destroyed to eliminate the defense.

The offense wants to minimize U, while the defense seeks to maximize it:

$$\max_{\{y_s,x_i\}}\ \min_{\{a_t,a_q\}}\ U \qquad (4.8)$$

A formal statement of the problem also includes constraints reflecting stockpile size and shot integrality. The offense allocation constraint

$$\sum_{t=1}^{T}a_t + \sum_{q=1}^{Q}N_q = N \qquad (4.9)$$

requires that the sum of targets directed at C^3 nodes and at assets sum to the offensive stockpile. The defense allocation constraint

$$\sum_{t=1}^{T}\sum_{s=1}^{a_t}y_s + \sum_{q=1}^{Q}\sum_{i=1}^{N_q}x_i = M \qquad (4.10)$$

similarly requires that defensive weapons sum to the interceptor stockpile size.

The utility function, however, is nonlinear, nonconvex and nonconcave. Minimal analytical insight has been developed, and initially we discuss small numerical examples.

4.3 Discussion

The best way to demonstrate the complexity of strategies for such problems is to first consider simple cases. Assume that the defense has the last move, and also that kill probabilities for targets against BM/C^3 nodes (p) and interceptors against targets (r) are unity; while targets against assets are less than one. Assets are taken to be of equal value and hence normalized to one. The offense, then, can either attack just the assets, or first the BM/C^3 nodes and then the assets. If the control nodes are attacked the defense will defend just one by matching each incoming missile, obviously choosing the most lightly attacked. As long as there are interceptors left they will be so used in the first stage, for otherwise they are useless in the second. If there are any left for the second stage, they will similarly be employed to successively defend the least attacked targets. Thus the offense will attack all C^3 nodes evenly, as it will also do for assets (though not necessarily at the same level for both). Note that this is true only when C^3 nodes are of equal value, the same applies to the asset attacks. The purpose of a precursor attack, with these kill probabilities, is to deplete the defender's stockpile rather than to actually destroy the control nodes.

An example from [4], with M=12, N=36, Q=12, p=1/3 (targets against assets), p_c=r=1, v_q=1 for all q and 1 radar (rather than C³ nodes), shows that the offense minimizes expected surviving value by exhausting the defensive stockpile in the first stage and then attacking the assets. [4] also give expected surviving values for different strategies for the same parameters when there are 2 radars, and find that the optimal offensive strategy is to ignore the C³ nodes and attack the assets with the entire stockpile. But with the unity kill probabilities used, it can be observed that the offense will never choose a two stage attack (regardless of the value of p) if there is more than one radar. Since the offense attacks radars evenly, the defense receives a target-to-interceptor exchange ratio equal to the number of radars, and this makes a precursor attack undesirable.

When the assumption of unity kill probabilities is relaxed, strategies become much more complex. Interceptors might no longer be singly matched to targets, and the destruction of C³ nodes loses its binary character. The offense and defense are able only to modulate the probability that nodes and targets are destroyed. The defense last-move case becomes a two-stage allocation problem, with two different defensive objectives: BM/C³ allocation strategies seek to maximize the probability of at least one node surviving, while asset defense tries to maximize the expected surviving value.

In [4]'s example the offense would not attack the BM/C³ system if there were more than one node. Another example shows that this no longer need hold when weapons are imperfect. For the case of Q=4, N=12, p=.9 (both assets and C³ nodes) and r ranging from .7 to 1.0, figures 5-8 show optimal strategies for M=4, 6, 8 and 10. Note that the optimal strategies were obtained by an intelligent enumeration: against candidate offensive strategies the defense employs that allocation which maximizes U; the offense then selects the strategy corresponding to the minima of these maxima. Only by simplifications regarding p, r and the v_i is such an enumeration feasible. In figure 5 (M=4) the offense allocates 4 of its 12 weapons to attack the C³ nodes when r is low, but increases to 8 when r is high; this corresponds to defensive allocations of at first 2 and then 3 interceptors in defense of the C³ nodes. Not surprisingly, the number of expected survivors is quite low.

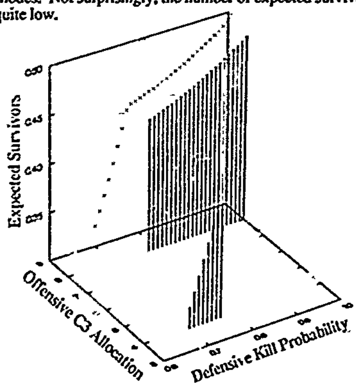


Figure 5 : P_c = .9, P_a = .9, Q = 4, N = 12, M = 4

In figure 6 (M=6), however, increasingly effective interceptors prompts the offense to decrease its C³ attack allocation from 4 to 0; expected survivors have increased significantly. This pattern continues in figure 7 for M=8, with the offense switching to an asset only attack at an even lower value of r. In figure 8, the offense will not attack the BM/C³ system even for r=.7. We conclude that more and more effective defensive interceptors discourage attacks on the BM/C³ system.

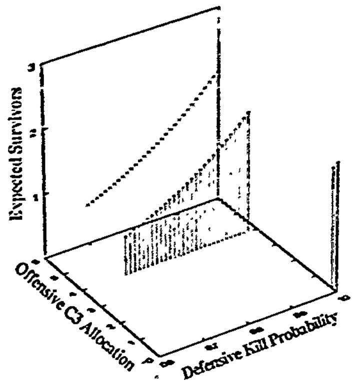


Figure 6 : P_c = .9, P_a = .9, Q = 4, N = 12, M = 6

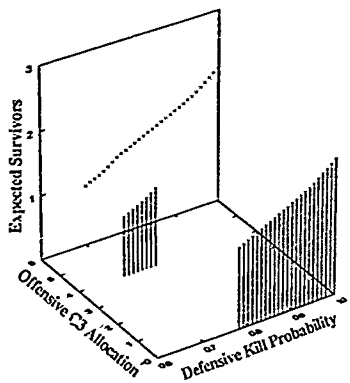


Figure 7 : P_c = .9, P_a = .9, Q = 4, N = 12, M = 8

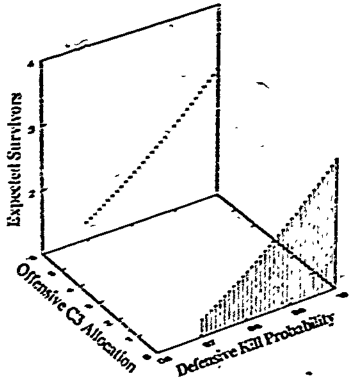


Figure 8 : $P_c = .9, P_a = .9, Q = 4, N = 12, M = 10$

In figures 9-12 we present the results when the C^3 nodes are softer targets than the assets: $p_c = .95$ and p_a (assets) = .9; other parameters are Q=2, N=12 and r from .5 to .7. When M=12 (figure 9) the offense attacks the C^3 nodes with 6 weapons for all values of r. When the defensive stockpile increases to 14, the C^3 attack level drops to 4 only for r close to 1.0 (figure 10). If increased to 16 or 18 (figures 11 and 12), however, the offensive C^3 allocation drops to as low as 2 for high r, and then switches to 4 and then 6 as r drops. It can be seen that softer C^3 nodes encourage attack, even for relatively large defensive stockpiles.

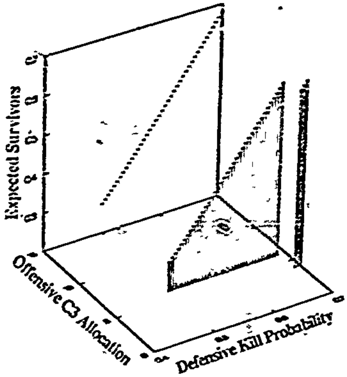


Figure 10 : $P_c = .95, P_a = .9, Q = 2, N = 12, M = 14$

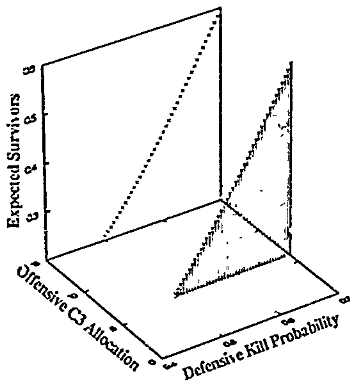


Figure 9 : $P_c = .95, P_a = .9, Q = 2, N = 12, M = 12$

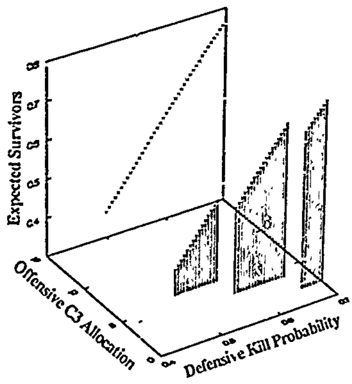


Figure 11 : $P_c = .95, P_a = .9, Q = 4, N = 12, M = 16$

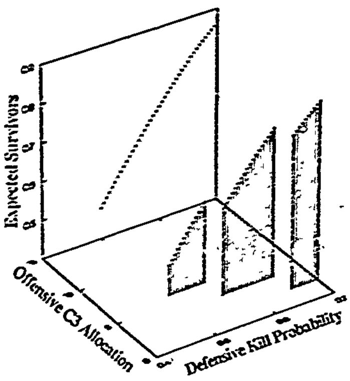


Figure 12 : $P_c = .95$, $P_a = .9$, $Q = 2$, $N = 12$, $M = 18$

Figure 13 shows the gain that replication of the command and control function provides. The ratio of the payoff for 2 C^3 nodes to that of 1 C^3 node is plotted against values of r that range from .70 to 1.00 and for defensive stockpiles of 4, 8 and 12.

Payoff Ratio for Defense with 2 C3 Nodes : 1 C3 Node

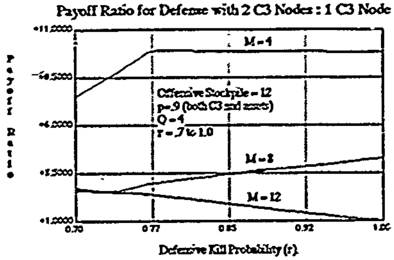


Figure 13

For small M (4), the gain is quite marked: it climbs from 7 to over 10. The plateau occurs at the point at which the offense switches from 2 to 4 and the defense from 2 to 3 weapons for the 2 C^3 node case. When M=8 the gain at first drops and then rises for increasing r, with the minima occurring where strategies for the one C^3 node case change. Strategies remain the same for all values of r when M=12, and the gain decreases monotonically - equalling one when r equals one. This is to be expected, since when the defense has 12 perfect weapons, it doesn't matter whether there are 1 or 2 C^3 nodes: all of the offense's weapons will be successfully intercepted. Figures 14 and 15 detail the strategies that lead to this result, both one and two C^3 nodes resulting in the same number of expected survivors when r=1, but with two C^3 nodes being more effective for lower values of r.

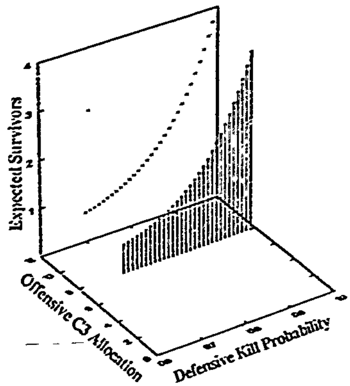


Figure 14 : P_c=.9, P_a=.9, Q=4, N=12, M=12, *1 C^3 Node*

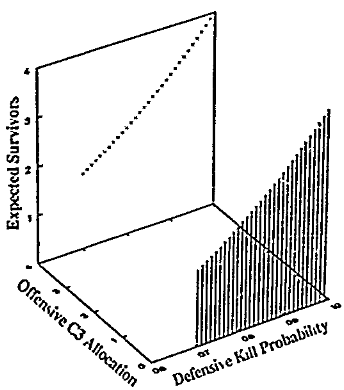


Figure 15 : P_c=.9, P_a=.9, Q=4, N=12, M=12, *2 C^3 Nodes*

5. CONCLUSIONS

The introduction of feedback, i.e. optimal "shoot-look-shoot-look-shoot-..." strategies, can significantly improve defense effectiveness. Some extensions to the present model, which we plan to consider, include stage-dependent target values and kill probabilities, as well as consideration of the dynamic version of the asset-defense problem. Unfortunately, this will mean dealing with substantially increased complexity.

In completing the study of the impact of vulnerable C^3 nodes, a limited domain of control for each node will be introduced. This begins to consider the effect that a distributed organizational structure has on the implementation of WTA strategies, and raises such questions as how many BM/C^3 nodes and of what type there should be. The issue of differing values for both C^3 nodes and assets must be addressed, since valuation will better enable us to quantify performance, and evaluate the tradeoffs that occur as distribution increases.

ACKNOWLEDGEMENT

This research was supported by the Joint Directors of Laboratories (JDL), Basic Research Group on C^3, under contract with the Office of Naval Research, ONR/N00014-85-K-0782.

REFERENCES

1. Papastavrou, J. and Athans, M., "Optimum Configuration for Distributed Teams of Two Decision-Makers", Proceedings of the 1988 Command and Control Research Symposium, Monterey, California, June 1988.

2. Lloyd, S.P. and Witsenhausen, H.S., "Weapons Allocation is NP-Complete", Proceedings of 1986 Summer Conference on Simulation, Reno, Nevada, July 1986.

3. denBroeder, G.G., Ellison, R.E. and Emerling, L., "On Optimum Target Assignments", *Operations Research*, Vol.7, pp. 322-326, 1959.

4. Eckler, A.R. and Burr, S A., Mathematical Models of Target Coverage and Missile Allocation, Military Operations Research Society, Alexandria, Virginia, 1972.

THE ANTI-SATELLITE MISSILE DEFENSE PROBLEM

R.M. James, J.J. Shaw

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

S.C. Chang

National Taiwan University
College of Engineering
Department of Electrical Engineering
Taipei, Taiwan, China

ABSTRACT

We present in this paper interceptor assignment algorithms for space-based weapon satellites conducting self- and mutual-defense against anti-satellite weapons. We demonstrate that the optimal solution to the self-defense problem can be readily obtained using a coordinate-ascent algorithm. We also show that the mutual-defense problem is a non-convex integer programming problem that is at least as complex as the general weapon-target assignment problem, which is known to be NP-complete. While coordinate-ascent algorithms can at best provide sub-optimal solutions, a Defense-in-Depth strategy that commits certain weapons to the defense of other weapons consistently improves the performance of a particular coordinate-ascent algorithm. Computational results presented here also demonstrate that mutual defense is consistently superior to self defense.

SECTION 1. INTRODUCTION

1.1 OBJECTIVE

In this paper, we examine the problem of defending space-based weapons for ballistic missile defense against an attack from co-orbital and direct-ascent anti-satellite (ASAT) missiles*. Understandably, the role of ASATs is to destroy the defense shield and thereby enhance the success of a subsequent ICBM attack. There are a number of responses the defense can use to counter an ASAT attack, including deploying decoys, maneuvering and firing back. We investigate the fire back option where the defense must decide how to best allocate its interceptors between ASAT and ICBM defense. Our objective is to present algorithms for the ASAT defense problem and to examine critical issues attending this special version of the weapon-target assignment problem.

1.2 OVERVIEW

We formulate the basic weapon allocation problem for mutual and self defense against an ASAT attack in Section 2. Implicit in this formulation is the assumption that the ASATs are launched in a single wave to create 'holes' in the defense prior to the ICBM launches. This assumption permits us to treat the ASAT and ICBM problems as two separate problems that are loosely coupled through values assigned by the defense to the interceptors on board every weapon surviving the ASAT attack. In practice, these values should reflect the likelihood that the individual weapons surviving the ASAT attack will have the opportunity to engage high-valued ICBMs in the future.

We develop algorithms for both the self defense and mutual defense ASAT weapon allocation problems in Section 3. The self defense problem can be solved optimally using a maximum marginal return (MMR) algorithm. The mutual defense problem, on the other hand, cannot be solved optimally using this algorithm. We present in this paper arguments for solving the mutual defense problem using a 'defense in depth' strategy where weapons are grouped by value with the lower valued groups assigned lead responsibility for defending the higher value groups. We develop an extension to the MMR algorithm employing the 'defense in depth' concept, and will demonstrate that this method offers significant improvements in performance over the standard MMR algorithm.

SECTION 2. PROBLEM FORMULATIONS

2.1 THE MUTUAL-DEFENSE ASAT PROBLEM

The objective of the mutual-defense ASAT problem is to maximize the value of interceptors on space-based weapons surviving an ASAT attack. We assume that each interceptor on weapon platform j has been assigned a value V_j by the defense. We assume that each weapon is under attack by a set of ASATs, S_j, with $P_{i,j}$ denoting the probability that an ASAT i in S_j surviving the defense will destroy weapon platform j. We further assume that the defense has on hand all information needed to compute these values.

The probability that a weapon j survives the ASATs targeted against it is:

$$\prod_{i \in S_j} \left(1 - P_{i,j} \prod_{j=1}^{M} (1 - \rho_{i,j})^{X_{i,j}} \right) \tag{2-1}$$

where the probability that an ASAT i destroys weapon j is discounted by the probability that it itself survives the defense. The expected value of a weapon following an ASAT attack is the product of the value of the interceptors remaining on board and the net probability that it survives the attack:

* The research described in this paper was conducted for Capt. Lawrence G. Bain at the Rome Air Development Center under contract F30602-85-C-0303

$$V_j\left(N_j - \sum_{i=1} X_{ij}\right)\prod_{i\in S_j}\left(1 - P_{ij}\prod_{j=1}^{M}(1-\rho_{i,j})^{X_{i,j}}\right) \quad (2\text{-}2)$$

Under the mutual-defense ASAT problem the objective of the defense is to maximize the expected value of the interceptors on board the weapons surviving the ASAT attack, leading to the following problem:

$$\max_{X_{ij}} \sum_{j=1}^{M} V_j\left(N_j - \sum_{i=1} X_{ij}\right)\prod_{i\in S_j}\left(1 - P_{ij}\prod_{j=1}^{M} (1-\rho_{ij})^{X_{ij}}\right) \quad (2\text{-}3)$$

subject to the interceptor availability constraints:

$$\sum_{i=1}^{N} X_{ij} \le N_j \quad (2\text{-}4)$$

and to the constraint that the $X_{i,j}$s be integer..

The mutual-defense static problem (MD-SP) is a <u>non-convex</u> integer programming problem that is at least as complex as the general weapon-target assignment problem, which is NP-complete [3]. Consequently, there is not evidence that efficient polynomial-time algorithms exist that can locate optimal solutions in all problem instances.

2.2 THE SELF DEFENSE ASAT PROBLEM

A special case of the ASAT weapon allocation problem is where each weapon defends only itself. This condition may arise when individual battle managers are forced to operate independently without recourse to a global battle assessment database. As in the mutual defense formulation, we assume that each interceptor on weapon platform j has been assigned a value V_j by the defense. We further assume that each weapon is under attack by a set of ASATs, S_j, with $P_{i,j}$ denoting the probability that an ASAT i in S_j surviving the self-defense system will destroy weapon platform j. Similarly, we let $\rho_{i,j}$ denote the probability that an interceptor from weapon j will destroy ASAT i.

The objective of the self defense (SD) problem for weapon j is to maximize the expected value of its interceptors surviving the ASAT attack:

$$\max_{X_{ij}} V_j\left(N_j - \sum_{i\in S_j} X_{ij}\right)\prod_{i\in S_j}\left(1 - P_{ij}(1-\rho_{ij})^{X_{ij}}\right) \quad (2\text{-}5)$$

subject to the interceptor inventory constraint:

$$\sum_{i\in S_j} X_{i,j} \le N_j \quad (2\text{-}6)$$

and to the integrality constraint as before.

While the objective function for this problem is nonconcave, the logarithm (SD') is additive and concave in $X_{i,j}$:

$$\text{MAX } \ln(V_j) + \ln(N_j - \sum_{i\in S_j} X_{i,j}) + \sum_{i\in S_j} \ln(1-P_{i,j}(1-\rho_{i,j})^{X_{i,j}}) \quad (2\text{-}7)$$

subject to Eq. 2-6 as before. SD' is a concave maximization problem, and its optimal solution can be obtained using a coordinate-ascent algorithm such as the Maximum Marginal Return (MMR) algorithm [1]. Furthermore, we note that the optimal solution to SD' is also the optimal solution to SD because SD = exp(SD') and exp(·) is a monotonically increasing function.

SECTION 3. SOLUTION ALGORITHMS

3.1 OPTIMAL ALGORITHM FOR SELF DEFENSE

We now present the MMR algorithm for finding the optimal self defense strategy for an arbitrary weapon j. We let $\underline{X}$ represent a column vector having length equal to the number of ASATs attacking the weapon, and e_i denote the column vector having a 1 in the i^{th} position and zeroes elsewhere. We also use $J(\underline{X})$ to denote the value of SD' when the interceptor-ASAT assignments are given by the values in $\underline{X}$. The MMR algorithm for self defense is presented in Figure 1.

```
X := 0;
Optimal_Not_Found := true;
while (Optimal_Not_Found) do
begin
  Maximum_Benefit_Found := 0;
  for i ∈ Sj do
    if Maximum_Benefit_Found < J(X + ei) - J(X) then
      begin
        Maximum_Benefit_Found := J(X + ei) - J(X);
        Best_Shot := i;
      end;
  if Maximum_Benefit_Found > 0 then
    begin
      Xi,j := Xi,j + 1;
      Nj := Nj - 1;
    end;
  if (Maximum_Benefit_Found = 0) or (Nj = 0) then
    Optimal_Not_Found := false;
end;
```

Figure 1. The MMR Algorithm for the Self Defense Problem

3.2 HEURISTICS FOR THE MUTUAL DEFENSE PROBLEM

The mutual defense problem is considerably more difficult than the self defense problem; the objective function for mutual defense is additive and cannot be transformed into a strictly concave function. This problem does have a particular structure, however, that motivates a heuristic based on a 'defense in depth' (DID) concept, where certain weapons take the lead responsibility for defending other weapons. We need to take a moment to introduce an important observation that supports the DID concept.

Let us suppose that a weapon k under attack from ASAT α can be defended by itself or by another weapon k'. Let us further suppose we have partially assembled a mutual defense solution, and define the following terms given the current partial solution assembled so far:

- $C_{k'}$: the probability that weapon k' survives the ASAT attack;
- $L_k(\alpha)$: the probability that weapon k survives all ASATs attacking it other than α;
- S_α : the probability that ASAT α will survive the defense

The cost of assigning an interceptor from weapon k' to defend weapon k from ASAT α is simply:

$$V_{k'} \cdot C_{k'} \tag{3-1}$$

while the benefit obtained by the defense in selecting this assignment is:

$$V_k\Big(N_k - \sum_{i=1} X_{i,k}\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k'} \tag{3-2}$$

Similarly, the cost of assigning an interceptor from weapon k to defend itself from ASAT α is:

$$L_k(\alpha)\Big[1 - P_{\alpha,k}\ S_\alpha\ (1 - \rho_{\alpha,k})\Big] \tag{3-3}$$

while the benefit obtained by the defense in selecting this assignment is:

$$V_k\Big(N_k - \sum_{i=1} X_{i,k}\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k} \tag{3-4}$$

Let us now suppose that it is better to use an interceptor from weapon k' to defend weapon k from ASAT α rather than have weapon k defend itself, implying the following inequality:

$$V_k\Big(N_k - \sum_{i=1} X_{i,k}\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k'} - V_{k'} \cdot C_{k'} >$$

$$V_k\Big(N_k - \sum_{i=1} X_{i,k}\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k} -$$

$$L_k(\alpha)\Big[1 - P_{\alpha,k}\ S_\alpha\ (1 - \rho_{\alpha,k})\Big] \tag{3-5}$$

or, re-arranging terms:

$$L_k(\alpha) - V_{k'} \cdot C_{k'} >$$

$$V_k\Big(N_k - \sum_{i=1} X_{i,k} + 1\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k}$$

$$- V_k\Big(N_k - \sum_{i=1} X_{i,k}\Big) L_k(\alpha)\ P_{\alpha,k}\ S_\alpha\ \rho_{\alpha,k'} \tag{3-6}$$

(Note: this inequality does not imply that the defense should commit an additional interceptor to the defense of weapon k, only that it prefers to do so using an interceptor from weapon k' rather than one from weapon k.) Suppose the defense does indeed assign one interceptor from weapon k' to defend weapon k against ASAT α. The probability of survival for ASAT α, S_α, is now reduced by a factor of $1 - \rho_{\alpha,k'}$, affecting only the righthand side of Eq. 3-6 and leaving the direction of the inequality the same as before. This observation can be summarized as follows:

> If, at some iteration in the development of the mutual defense solution, weapon k is best defended by weapon k' than by one of its own interceptors, then this preference will exist throughout the completion of the solution.

This observation is the motivation for mutual defense ASAT weapon allocation algorithms based on a 'defense in depth' (DID) concept, where certain weapons comprise the first line of defense against ASATs. The weapons comprising the first line of defense are assigned no survival value (i.e., the defense reaps no benefit if those weapons themselves survive the ASAT battle intact), while the defended weapons engage in self defense should the ASATs survive the first line of defense. The DID concept divides the mutual defense problem into three subproblems: (1) finding the optimal partitioning of weapons into the first and second lines of defense, (2) finding the best allocation of interceptors to ASATs for those weapons comprising the first line of defense, and (3) finding the best self-defense solution for the defended weapons should the first line of defense fail.

The first subproblem is NP-complete when the second and third subproblems are counted as one computation, and is therefore a nontrivial problem in its own right. Nevertheless, special characteristics of space-based weapons suggest some guidelines for partitioning weapons at the moment that an ASAT attack occurs. For example, weapons moving in a positive ascension towards the North Pole will not be in position to engage ICBMs that appear shortly after the ASATs, and therefore should be placed in the first line of defense. Similarly, weapons moving in the opposite direction will be in ideal positions to engage a subsequent ICBM attack, and should be placed in the second line of defense. We are presently examining several heuristics for solving this first subproblem, and will report on these in a later paper.

The second subproblem leads to the following preferential defense optimization problem:

$$\text{MAX} \sum_{j \in P} V_j \prod_{i \in S_j} [1-P_{i,j} \prod_{k \in D} (1-\rho_{i,k})^{X_{i,k}}] \quad (3\text{-}7)$$

where D and P denote the weapons comprising the first and second lines of defense, respectively. This problem is also nontrivial to solve [2, 4], suggesting that polynomial-time algorithms will yield sub-optimal solutions at best. Our current algorithm for this problem is based on the MMR algorithm we presented in Figure 1 for the self-defense problem; we report on its solution efficacy in the next section. Recently, Castanon, et al. [4] have developed several algorithms that are superior to MMR for solving this preferential defense problem. We plan to incorporate those algorithms into our analyses and report our findings at a later date.

3.3 A SIMPLE COMPUTATIONAL EXAMPLE

We use a simple example to compare solutions yielded by self defense, mutual defense and Defense-in-Depth mutual defense algorithms. We consider the 2 weapon/3 ASAT problem described in Table 1; Weapon 1 has 5 interceptors and is being attacked by 2 ASATs, while Weapon 2 has 2 interceptors and is being attacked by a single ASAT. The DID mutual defense strategy will place Weapon 2 in the first line of defense, commit its two interceptors to the defense of Weapon 1, and leave Weapon 1 to engage in self defense against any ASAT surviving the first line of defense. This strategy yields an expected interceptor survival value of 3.4.

The mutual defense strategy without DID will conclude when it selects its first interceptor assignment that Weapon 1 has less than a 10% probability of survival if a <u>single</u> interceptor is committed to its defense. Conversely, it will conclude that Weapon 2 has a good chance of surviving intact if a single interceptor from Weapon 1 is used for its defense. Consequently, the MMR algorithm is thrown off the path leading to the optimal solution at the very first iteration, and proceeds to commit all five interceptors from Weapon 1 to the defense of Weapon 2, yielding an expected interceptor survival value of 1.922.

The self defense strategy results in Weapon 1 assigning one interceptor apiece to the ASATs threatening it, and Weapon 2 not, and Weapon 2 not defending itself at all. This strategy yields an expected interceptor survival value of 1.6.

In short, the survival value from the optimal mutual defense solution is more than twice that obtained by the self defense solution, and 77% larger than the solution obtained by mutual defense without Defense-in-Depth.

SECTION 4. COMPUTATIONAL RESULTS

We now present performance results for the ASAT weapon allocation algorithms (self defense, mutual defense and mutual defense with DID) described above. These algorithms were tested on problem sets specially constructed to evaluate performance under conditions where weapons enjoy relative advantages over ASATs, and vice versa. Our purpose in constructing these problem sets was to determine whether or not certain problem instances defeat any one or all three of these algorithms.

The test cases are divided into four blocks. The objectives of Blocks I thru III are to compare the performance of the MMR algorithm for mutual defense without DID against the performance obtained by the MMR algorithm when the weapons engage exclusively in self defense. Here, we are comparing sub-optimal solutions obtained using a superior problem formulation (MMR for mutual defense) to optimal solutions obtained using an inferior problem formulation (MMR for self defense). It is our hypothesis that the real determinant of performance is not the choice of algorithm, but the choice of formulation: MMR for mutual defense should outperform MMR for self defense.

	WEAPON 1		WEAPON 2	
ASAT 1	0.4	0.9	0.3	0.0
ASAT 2	0.4	0.9	0.8	0.0
ASAT 3	0.4	0.0	0.8	0.5
INTERCEPTORS	5		2	

Table 1. Parameters for the 2 Weapon/3 ASAT Example (weapon-ASAT kill probabilities appear in the upper lefthand corners, ASAT-weapon kill probabilities appear in the lower righthand corners)

The objective of the Block IV test case is to compare the performance of algorithms for mutual defense with and without DID. The DID concept actually forces the defense to invest interceptors to the defense of high-valued weapons, even when the initial investments appear to offer little payoff. Given the <u>non concave</u> nature of the mutual defense problem, these initial investments can place the defense in a position where subsequent interceptor commitments reap very large payoffs in effectiveness. It is our hypothesis, then, that the DID concept should outperform mutual defense without DID.

4.1 PERFORMANCE RESULTS FOR THE BLOCK I TEST CASES

The Block I test cases were conducted for both the MMR mutual defense and MMR self defense algorithms for varying ASAT-to-weapon ratios. The tests cases involve 10 weapons with 15 interceptors apiece, weapon-ASAT probabilities of kill of 0.6, and ASAT-weapon probabilities of kill of 0.7. The performance results for this block are presented in Figure 2. Mutual defense is always superior to self defense, and confers a higher relative value to the defense against ASAT-rich attacks. This latter characteristic is easily explained: no one weapon has enough interceptors to successfully defend itself in an ASAT-rich environment, but one or more weapons acting in concert can successfully defend a single weapon.

4.3 PERFORMANCE RESULTS FOR THE BLOCK II TEST CASES

The objective in the Block II test cases is to compare mutual defense against self defense under varying weapon-ASAT probabilities of kill. Like the Block I experiments, these test cases involve 10 weapons with 15 interceptors apiece, and ASAT-weapon probabilities of kill of 0.7; the ASAT-to-weapon ratio is 3:1. The performance results for this block are presented in Figure 3. As in Block I, mutual defense is always superior to self defense, and confers its highest value to the defense when the defense cannot engage ASATs with a high probability of kill. This latter characteristic follows for the same reasons that self defense is inferior to mutual defense in an ASAT-rich environment.

Figure 2. Comparative Post-ASAT Attack Defense Value Surviving for the Block I Tests

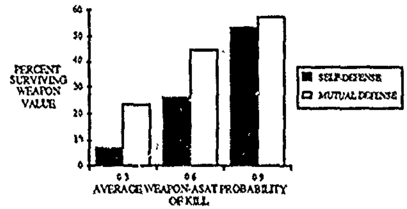


Figure 3. Comparative Post-ASAT Attack Defense Value Surviving for the Block II Tests

4.4 PERFORMANCE RESULTS FOR THE BLOCK III TEST CASES

The objective in the Block III test cases is to compare mutual defense against self defense under varying spreads in weapon value. Intuitively, the defense should prefer mutual defense when there is a clear distinction in weapon values, since low-valued weapons can be committed to the defense of high-valued weapons. These test cases involve 30 weapons with 15 interceptors apiece, weapon-ASAT probabilities of kill of 0.6, ASAT-weapon probabilities of kill of 0.9, and an ASAT-to-weapon ratio of 3:1. The performance results presented in Figure 4 confirm our intuition. While the absolute performance of the self defense strategy decreased with larger spreads in weapon value in these experiments, we believe this is an artifact of the Monte-Carlo methods used here. An examination of Eq. 2-7 reveals that weapon value does not influence the weapon allocation for self defense. The weapon probabilities of survival following the ASAT attack should be statistically the same for all weapons, and thereby leave the overall expected surviving value unchanged if the distribution of weapon values are not skewed about the mean value.

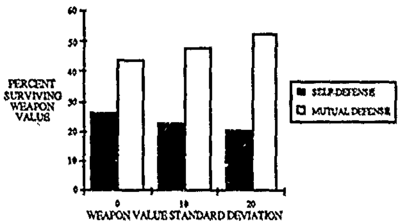


Figure 4. Comparative Post-ASAT Attack Defense Value Surviving for the Block III Tests

4.5 PERFORMANCE RESULTS FOR THE BLOCK IV TEST CASES

The objective of the Block IV test case is to compare the performance of algorithms for mutual defense with and without DID. Assuming a rationale offense, we would expect to see the ASAT attack concentrated around the high-valued weapons, leaving these weapons with low probabilities of survival if a defense is not taken on their behalf. This creates a misleading impression that the interceptors on board those weapons have a low expected value following the ASAT attack. Consequently, an unsophisticated mutual defense algorithm may conclude the defense has nothing to lose by letting those high-valued weapons engage in self defense. The DID concept, however, forces the defense to invest interceptors from one set of weapons to the defense of high-valued weapons, even when the initial investments appear to offer little payoff. Given the non-concave nature of the mutual defense problem, these initial investments can place the defense in a position where subsequent interceptor commitments reap very large payoffs in effectiveness.

The Block IV test cases involve a set of 15 weapons having interceptors worth 10 units apiece and a second set of 15 weapons having interceptors worth nothing; weapons in both sets each have 10 interceptors. With a value of zero, the weapons in the second set naturally comprise the first line of defense under a DID strategy. Letting P_1 and P_2 denote the average weapon-ASAT probability of kill for the weapons in the first and second sets, respectively, we examine mutual defense with and without DID for varying ratios of P_1: P_2. DID strategies offer no advantage over strategies not using DID when this ratio is equal to 0 since the weapons designated for the first line of defense are completely ineffective against the ASATs. We also expect DID strategies will offer little relative advantage when this ratio is equal to 1 since the mutual defense strategy will not require the special constraint implied by DID to conclude that low-valued weapons are best used in the defense of high-valued weapons.

The results presented in Figure 5 support our conjectures Surprisingly, however, mutual defense using a DID strategy outperforms mutual defense without DID when the P_1: P_2 ratio is as high as 0.8, meaning that the mutual defense strategy unaided by the DID constraint will fail to protect high-valued weapons using low-valued weapons that have favorable chances of intercepting ASATs.

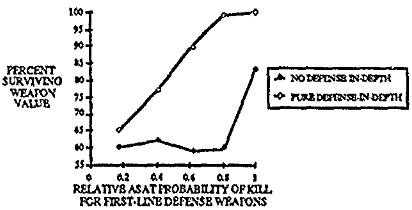


Figure 5. Comparative Post-ASAT Attack Defense Value Surviving for the Block IV Tests

SECTION 5 SUMMARY

5.1 CONCLUSIONS

We presented three algorithms for the ASAT weapon allocation problem in this paper, all algorithms are based on the principle of maximum marginal return (MMR). The self defense problem is actually a trivial problem to solve, and can be solved optimally using an MMR algorithm. The mutual defense problem is considerably more difficult, but it has a particular structure that motivates algorithms based on 'defense in depth' (DID) concept where certain weapons take lead responsibility for defending other weapons. The weapons comprising the first line of defense are assigned no survival value and the defended weapons engage in self defense should the ASATs survive the first line of defense. The allocation problem for the weapons comprising the first line of defense is quite a difficult problem; the allocation problem for the defended weapons engaging in self-defense is trivial in view of our comments above.

We presented a simple computational example suggesting both that self-defense is inferior to mutual defense, and that DID seems to be a preferred strategy for conducting mutual defense. We then corroborated this finding with additional computational results based on large scale problems. These results demonstrate that mutual defense is consistently superior to self defense over a wide range of scenarios. Self defense commits interceptors to too many ASATs to achieve an effective defense for the weapons. Mutual defense, on the other hand, lets one or more weapons act in concert to assure the survival of another weapon, and thereby focuses the use of defense resources to good effect.

Our results also demonstrate that defense in depth (DID) can be a very effective concept for mutual defense. Just as mutual defense focuses the weapon-ASAT allocations to specific objectives, DID does not commit high-valued weapons to self defense before exhausting all of the mutual defense opportunities offered by low-valued weapons. We know that the mutual defense problem is nonconvex, and has many locally optimal solution, therefore the choice of an initial solution in the neighborhood of good locally optimal solution is critical to algorithm performance. We have demonstrated that the Defense-In-Depth doctrine yields such a choice.

REFERENCES

[1] denBroder, G.G., R.E. Ellison, and L. Emerling, "On Optimum Target Assignment," Operations Research, Vol 7, 1959, pp 322-326.

[2] Castanon, D.A., et al., "Advanced Weapon-Target Assignment Algorithm - Quarterly Report," TR-353, ALPHATECH Inc., Burlington, MA, November 1987.

[3] Lloyd, S.P., and H.S. Witsenhausen, "Weapon Allocation is NP-Complete," AT&T Bell Laboratories, Whippany, NJ, December 1986.

[4] Castanon, D.A., et al., "Advanced Weapon-Target Assignment Algorithm - Quarterly Report," ALPHATECH Inc., Burlington, MA, April 1988.

TOWARD A GENERAL THEORY OF C^3 PROCESSES

DR. I.R. GOODMAN

COMMAND & CONTROL DEPARTMENT

CODE 421

NAVAL OCEAN SYSTEMS CENTER

SAN DIEGO, CALIFORNIA 92152-5000

ABSTRACT

This paper continues the past effort on the part of the author in establishing an approach to a general theory of C^3 processes and an ensuing decision game. Previous work is corrected, modified, and extended here. An outline of a general C^3 decision game is first presented, utilizing as an integral part, a formal theory describing the evolution of a typical C^3 node state vector. The determination of the C^3 decision game also requires an algebraic logic description pair assigned to the formal theory, such as probability logic or fuzzy logic, depending on the type of information considered. In the probability logic case, a general result concerning uniform approximation by linear-gaussian mixtures of distributions is presented. This is shown to have potentially good applications for carrying out reduced calculations of evolving node state distributions, as functionals of pertinent distributions of various C^3 variables and relations.

1. INTRODUCTION

For the past ten years or so, C^3 as an organized discipline has evolved greatly from the fledgling concepts of J. Lawson, M. Athans, and others, as presented in the Proceedings of the First MIT/ESL-ONR Workshop on Distributed Communication and Decision Problems [1], to the much more sophisticated views presented in the succeeding Proceedings of the 1988 Command and Control Research Symposium sponsored by the Joint Directorate of Laboratories [2].

In addition to the seminal work presented in the Proceedings during the eight years of the MIT/ONR Workshop on C^3 Systems (from the First to the Ninth) and in the Proceedings of the past two years of the Command and Control Research Symposium, other basic sources for unclassified unlimited analysis of general C^3 systems include:

(1) C^3 Handbook (Joint Directorate of Laboratories) [3], where a large number of abstracts and summaries of C^3 published papers are collected along with a number of critical reviews, comparisons, and contrasts.
(2) Proceedings of the British-based Conference on Command, Control, Communications, and Management Information Systems [4] and related publications.
(3) Special Issue on Information Technology for Command and Control, IEEE Transactions on Systems, Man, and Cybernetics [5].
(4) Science of Command and Control (AFCEA)[6], exhibiting selected papers from presentations at the Command and Control Research Symposium. In addition, see the AFCEA magazine Signal for many nontechnical papers on C^3.
(5) Command and Control Evaluation Workshop (Proceedings) sponsored by the Military Operations Research Society [7].
(6) Elements of C^2 Theory [8].
(7) Proceedings for Quantitative Assessment of Utility of Command and Control Systems [9].

For a brief overview of the more important papers from the MIT/ONR Workshop on C^3, up to 1985, see Goodman [10]. More recently, H.L. Van Trees (in this issue [11]) has presented an excellent critical analysis of the state-of-the-art of C^3 work.

Based upon all of the above-mentioned studies of C^3 systems, one can conclude that past and current work in this area can be divided roughly into the following taxonomy:

A. C^3 systems relative to long range strategies, global and political ramifications, and high-level planning.
B. Tactical or mid-level C^3 systems and processes, emphasizing the actual dynamics of C^3 events during typical engagements. Note. Throughout this paper, little or no distinction is made between the use of the terms "systems" and "processes", nor between "measures-of-effectiveness (MOE's) and "measures-of-performance" (MOP's). However, for a different viewpoint, see e.g., Metersky [12].

1. Qualitative studies, emphasizing the use of graphs, flow-charts, and verbal descriptions.
2. Quantitative, emphasizing numerical measures or algebraic relations, usually centered about the stochastic state space approach.
   a. Aspect-oriented quantitative work, where one, or possibly more, particular facets of tactical C^3 processes are considered. Typical examples of this are. surveillance and tracking/correlation; tactics for a particular class of scenarios; deployment for air-to-ground conflicts, determination of MOE's for C^3 systems; resource allocation of troops or weapon studies; communications analysis, C^3 de-

cision-aids; aspects of distributive decision-making and game theory applied to restricted situations; etc.

b. General structured C^3 analysis, where an attempt is made toward developing a general view of C^3 processes as a whole entity.

C. Technological hardware/software considerations; database management; computer design; field exercise studies; teaching and training of C^3 concepts.

Suffice it to say, the vast majority of endeavor in C^3 analysis has been directed to B1, B2a, and C, with relatively little attention paid to A (understandable relative to the open literature format) and B2b. Certainly, qualitative studies are still very useful due not only to the inherently great complexity of C^3 systems, but also since heuristic analysis must always precede any quantitative. Furthermore, quantitative aspect-oriented work is also of prime importance, since the various parts of the whole C^3 picture are themselves challenging and complex problems. Understanding of the bits and pieces of C^3 cannot but contribute to the entire view. Nevertheless, it is the contention here that more efforts must be made to discern the general pattern of C^3 dynamics, before C^3 work is to gain the stature of other scientific disciplines.

For some attempts at considering C^3 tactical processes as a whole entity, see e.g., the independent work of Ingber [12] and Rubin and Mayk [13], as well as follow-up papers by these authors in the Proceedings of the MIT/ONR Workshop on C^3 Systems and these Proceedings. Ingber's approach to the problem is a mesoscopic-macroscopic one, considering C^3 systems analogous to systems of interacting molecules or neurons, suitably modified and analyzable from a statistical mechanics viewpoint. On the other hand, Rubin and Mayk's approach is more microscopic in nature, also utilizing a purely stochastic technique, originally centering about numerical supply and attrition levels- generalizing the well-known Lanchester equations for mutual growth and decay of interacting populations. Although the thrust of their work has greatly expanded, modeling of the internal behavior of C^3 node complexes of decision-makers is not explicitly taken into account, as Levis and others have done [14],[15]. There, the behavior of individual decision-makers and their constraints and interactions with others are modeled from an organizational viewpoint, utilizing in one key part Conant's decomposition of entropy, as well as other criteria. (See also the subsequent papers by Levis and his students at the Laboratory for Information and Decision Systems (LIDS), MIT in these Proceedings and those of the MIT/ONR Workshop on C^3 Systems.)

With relatively modest changes, one can attempt to model the entire tactical C^3 process, using individual node structures and states as the basic building blocks, without necessarily making stochastic assumptions. In particular, the author has proposed such an approach [16], with a follow-up paper [17] elaborating upon the use of non-probabilistic models in C^3 analysis. Much of this was based on previous work concerning the modeling and combining of uncertainties arriving from possibly disparate sources, stochastic or linguistic [18]. (See also [19] for a practical implementation of such an approach for tracking and correlation of data.)

2. SCOPE OF THE PRESENT WORK

As stated earlier, the current task is an improvement and enlargement of the previous efforts in establishing an overall model for C^3 dynamics [16],[17].

The basic goal of this paper is twofold:

(1) To show tactical C^3 processes can be modeled within a game theory setting, using a formal system of axioms, capturing a minimally required number of relations among C^3 variables and operators.

(2) To provide an outline for a feasible implementation of this program, as an aid in designing and predicting global behavior of C^3 systems.

In modeling of C^3 processes, one must always be aware that fidelity of a theory must be traded-off with complexity of resulting computations required for implementation. With this in mind, it is the thesis here that one should still attempt to first model C^3 processes as a whole- despite the complexities- and then seek reasonable reductions of computations.

In section 3, the basic C^3 model is developed as a decision game. This requires five steps:

(i) Choice of a set of axioms involving relevant C^3 variables and operators, so that a formal description of dynamically evolving node states can be obtained.

(ii) Choice of an algebraic logic description pair for evaluating numerically the formal theory developed in step (i).

(iii) Specification of an averaging procedure relative to all C^3 node state evaluations in (ii).

(iv) Use outputs of (iii) to determine the overall "health" state, i.e., tendency for winning, of each C^3 system.

(v) Use the figures-of-merit in (iv) to define the overall loss function, and hence, C^3 decision game.

As in previous work, C^3 processes here are viewed as interacting networks of node complexes of decision-makers. Relevant C^3 variables are first identified, including nodal ones such as equations of motion, attrition level, detection state, algorithm selection, and hypotheses evaluations. Other C^3 variables treated involve the reception of signals or incoming exploding weapons, and the response following data processing. In addition, logical operators, such as conjunction, disjunction, and negation, and conditioning operators among C^3 variables are also taken into account. To implement step (i), a collection of axioms is presented in section 3, formally reproducing the essential relations among C^3 variables and operators. These axioms modify and extend analogous ones presented in [16]. These axioms reflect the typical time cycle- somewhat simplified- of a node, beginning with the reception of arriving "signals", i.e., information, weapons, or any other incoming entity or entities which can provoke change in the node state, followed by information processing and related activity, and ending with the time of output response by the node to other nodes, friendly or adversary. With the formal language established through the axioms, using the standard rules of deduction, a basic theorem is derived (Theorem 4.1), giving a formal description of the dynamic evolution of a typical node state vector at the end of the input-output cycle as a functional of the same node state at the beginning of the cycle just prior to the reception of the "signal" and of all other pertinent C^3 variables. This leads immediately to a recursive form for each node state's evolution, beginning with the original initialization of the C^3 system.

In step (ii) an algebraic logic description pair (ALDP) is chosen, compatible with the above formal descriptions, in order to obtain the full quantitative evaluation. Typically, probability logic can play this role, but in order to utilize linguistic-based and other types of information, other logics can be used just as well, such as Zadeh's fuzzy logic or Dempster-Shafer belief logic. A scheme is presented for utilizing these evaluations in a suboptimal or marginal sense as inputs to an overall two "person" (actually, friendly vs adversary C^3 processes) decision game. For more details on steps (iii)-(v), see the concluding part of section 3.

By choosing probability logic for implementation, a fundamental result (Corollary 5.2) can be invoked which is of potentially good use in evaluating the overall dynamic evolution of C^3 node states. Section 5 presents the details of this, where essentially, a uniform close approximation by finite linear-gaussian mixtures of distributions can be used to represent distributions of C^3 variables and in turn the evolving node states.

Finally, section 6 presents a brief discussion for application of the model in general and to a simplified inner-outer air battle scenario, in particular.

3. THE C^3 DECISION GAME

In all that follows, for simplicity, suppose that only two C^3 processes are being considered: one friendly and one adversary.

There are three types of variables describing a C^3 process:

N, node complex variables, representing the decision-makers, human or automated, and their immediate environment,

S, input "signal" variables, where, as explained before, "signal" need not refer to just an ordinary array of incoming information, but may, as well, denote incoming activated weapons or other disturbances to the initial node complex,

R, node output responses immediately following the complete "signal" processing.

Symbolically, one can represent the temporal relation among N,S,R, as

$$\cdots \to N \to R \to S \to N \to R \to S \to N \to R \to S \to \cdots , \qquad (3.1)$$

regardless of the particular nodes interacting with other nodes and the multiplicity of "signals" and responses. In general, an arriving "signal", originates- possibly from several-nodes N as intial responses R, but due to intermediate media distortion and change becomes S. See Figure 1 for a typical example of a C^3 process, showing how one can scope out the roles of "signals", responses, and nodes.

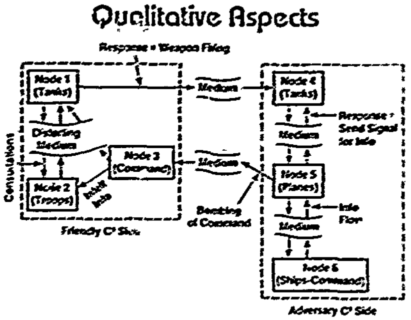


Figure 1. External Dynamics of C^3 Processes: Simplified

Variables N,S,R can be indexed - [illegible] - to indicate the particular C^3 process, friendly or adversary, the specific node involved, and the time. Node variables can be decomposed into first, two parts, with similar remarks holding for indexing of these and all subsequent decompositions of the main variables into subvariables:

$$N = (N , T) \qquad (3.2)$$

where N represents the node state, while T represents the internal node structure. See Figure 2 for an illustration of a typical node structure (relative to the processing of incoming "signals"). See Figure 3 for the basic evolution cycle due to the processing of a "signal" as in Figure 2. Note the superscripts $( )^-$, $( )^+$ to indicate relative times and $(S|P^-)$ to indicate conditioning. For simplicity, changes are only noted at times of response and times of reception, following medium distortion of responses.

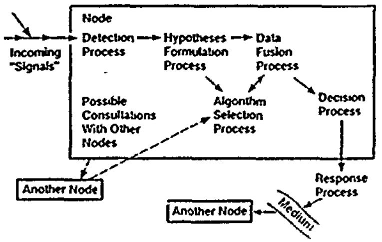


Figure 2. Internal Dynamics of C^3 Processes. Simplified

Typically, node states involve: threat levels, numbers of troops; numbers of weapons of different types; supply levels; equations of motion of the node, if it is not stationary, such as a tank or a formation of airplanes, [illegible] adversary. See Table I for a simple example. Symbolically, one can write N and T in subvariable form such as

$$N = (\#WP_1, \#WP_2, \#WP_3, \#TROOP, EQMO, \widehat{INFO}), \quad (3.3)$$

$$T = (DET, ALG, HYP, FUS, CONS, DEC), \quad (3.4)$$

where the above symbols are essentially self-explanatory, such as: $\#WP_2$ = number of weapons on-board of type 2; EQMO = equations of motion (of entire node); INFO = knowledge or estimates of other node states; DET = detection state (usually 0 or 1, for probabilistic interpretations); ALG = algorithms chosen for possible response; FUS = data fusion variables, further decomposable into appropriate subvariables, when required, such as in correlation of data in multi-target tracking; and DEC = decision variables representing actual decisions to be made based upon arriving "signals".

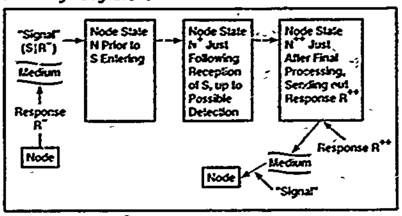


Figure 3. Basic Evolution Cycle of a Node Due to "Signal" Processing and Response.

Quantitative Aspects

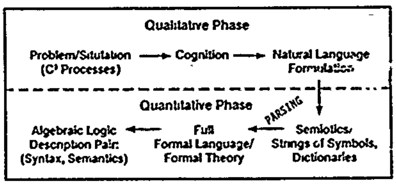
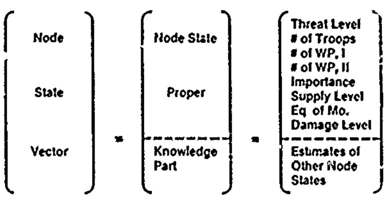


Figure 4. Knowledge Flow in Describing Situations

Table I. Components of C³ Node States



NODE = (NODE STATE , NODE STRUCTURE)

Consider now the basic C^3 structure indicated in (3.1)-(3.4), as well as time shift markers $(\ )^-$, $(\ )^+$, $(\ )_0$, conditioning operator (|). Also, note logical connectors & (conjunction/and/intersection), v (disjunction/or/union), $(\ )'$ (negation/not/complement) and set/event relations DOM (domain of possible values of the associated variable), ε (set membership relation), Ω (universal set of discourse) and $\emptyset$ (null or empty set). It is clear that any qualitative or quantitative description of a C^3 process as a whole entity must entail descriptions of the nodes constituting the process utilizing in some sense the above concepts.

With all of the above C^3 variables and operators noted and interpreted, the fundamental steps in establishing an overall model of tactical C^3 processes can now be attempted:

(i) Obtain a formal theory for the dynamic evolution of a typical node state N by use of an appropriate set AX of axioms involving C^3 variables and operators. In this case, N can be expressed as some functional of appropriate C^3 variables and operators symbolically described as

$$N = \mathcal{G}(AX). \quad (3.5)$$

Details of these relations are given in Table II, section 4.

(ii) Evaluate/quantify typical C^3 node states as given in (3.5) by choice of some <u>algebraic logic description pair</u> (ALDP). Such a pair consists of a syntax space with algebraic structure and a compatible semantic evaluation/logic function with range in the positive real line. Examples include:

CL(classical logic)=(boolean algebra,0-1 truth function),(3.6)

PL(probability logic)=(boolean algebra,probability measure),(3.7)

FL(Zadeh's fuzzy logic)=(browerian lattice,possibility function),(3.8)

DSBL(Dempster-Shafer belief logic)=(boolean alg., belief function).(3.9)

(See [17],[18] for further details.)

Indeed, one can view the ALDP evaluation as the final part of knowledge flow in describing not only C^3 processes, but situations in general, beginning with cognition. (See Figure 4 for a brief outline of this.)

Denote the numerical evaluation of each formal node state description symbolically as

$$p_{ALDP}(N,C^3) = [\![(N,C^3)]\!]_{ALDP}, \quad (3.10)$$

where p_{ALDP} or $[\![\ ]\!]_{ALDP}$ indicates the logic function associated with chosen ALDP. Thus, for PL, p_{ALDP} means a probability measure (or function, etc.), or if one chooses ALDP=FL, p_{FL} = poss, a possibility function, etc. For an ALDP in general call p_{ALDP} a <u>dispersion</u>, unless otherwise specified.

Utilizing (3.5) in (3.10) and applying the compatibility property of ALDP's yields the relation

$$p_{ALDP}(N,C^3) = \mathcal{G}_{ALDP,AX}(p_{ALDP}(AX,C^3)), \quad (3.11)$$

where N is a given node state of a C^3 process, denoted simply as C^3, $\mathcal{G}_{ALDP,AX}$ is a functional and $p_{ALDP}(AX,C^3)$, extending somewhat the notation in

(3.10), represents the collection of all dispersions involved in evaluating the RHS(right hand side) of (3.10) for given C^3 process (friendly or adversary) and choice of ALDP. Note that in a real-world situation, for given AX and ALDP, the set of dispersions $p_{ALDP}(AX,C^3)$ in general is highly dependent in form upon just what C^3 is, or equivalently, any allowable collection of dispersions represented by $p_{ALDP}(AX,C^3)$ determines completely C^3 and thus may be identified with any of its primitive set of dispersions determining all of C^3.

In the approach outlined in steps (i) and (ii), in effect, each C^3 node is considered separately or marginally, taking into account, however, all of the nodes that interact with it as well as all relevant C^3 variables, friendly or adversary. This is assumed at the outset in order to obtain a more simplified model than if all possible joint interactions at a given time were taken into account, analogous to the suboptimal determination of individual marginal distributions of a the much more (exponentially) complex joint distribution from which the marginals arise.

(iii) In turn, combine the results of step (ii) - individual updated marginal node state descriptions - back into a relatively simple global description of the C^3 process at hand. As mentioned above, this is done in the spirit of considering the very real tradeoff of model fidelity vs. complexity of calculations, in place of attempting to carry out the optimal full joint description of C^3 behavior. A reasonable combination operator AV here can be simple arithmetic, or more generally, weighted sum,averaging of corresponding node state entry dispersions,representing thus an updated overall C^3 process behavior. If arithmetic averaging is not desired, other suitable measures of central tendency can be used, such as modes or medians, among others. The basic output at this stage can be symbolized as

$$\bar{p}_{ALDP}(C^3) = AV(p_{ALDP}(N,C^3),\ \text{all } N \in C^3), \quad (3.12)$$

where as usual C^3 represents the particular C^3 process of interest.

(iv) Next, determine a functional $\mathcal{H}_{ALDP}$ which directly relates the updated averaged dispersions of a given C^3 process with a single or multiple attribute vector describing the overall "health" state of C^3:

$$HLTH_{ALDP}(C^3) = \mathcal{H}_{ALDP}(\bar{p}_{ALDP}(C^3)). \quad (3.13)$$

For example, $\mathcal{H}_{ALDP}$ could be a certain weighted mean - or perhaps involve set thresholds - of a number of MOE's (measures-of-effectiveness) and/or MOP's (measures-of-performance) of the C^3 process as a whole, each of which, in turn, depends upon $\bar{p}_{ALDP}(C^3)$.

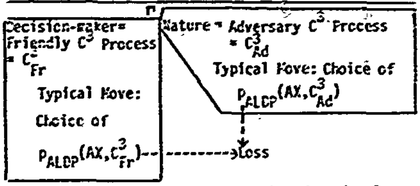
Examples of such MOE's and MOP's can include overall supply/attrition level, overall entropy level, various system performance criteria, including timliness, damage levels, kills, destruction of enemy level, etc. (For a good systematic exposition on MOE's and MOP's and related measures, see Sweet [21] or Rubin and Mayk's discussion in [13], pp. 15,16.)

Utilizing (3.11) and (3.12) in (3.13) yields

$$HLTH_{ALDP}(C^3) = \mathcal{G}_{ALDP}(p_{ALDP}(AX,C^3)), \quad (3.14)$$

for some functional $\mathcal{G}$ depending in part on $\mathcal{G}$, $\mathcal{H}$.

(v) Finally, one can establish the desired two person zero sum decision game Γ as shown schematically in Figure 5 below:



Loss is determined through a functional $\mathcal{G}_{ALDP}$.(See eq. (3.15).)

Figure 5. Schematic for C^3 Decision Game Γ.

The loss function in Figure 5 resulting from the moves of each player can be expressed as

$$\text{Loss} = \mathcal{G}_{ALDP}(HLTH_{ALDP}(C^3_{Fr}),HLTH_{ALDP}(C^3_{Ad}))$$
$$= \mathcal{K}_{ALDP}(p_{ALDP}(AX,C^3_{Fr}),p_{ALDP}(AX,C^3_{Ad})), \quad (3.15)$$

where $\mathcal{K}$ is determined through (3.11)-(3.14).

Thus, one can inquire whether Γ has a game value, what are the bayes strategies for each side, what is the least favorable move or strategy for the adversary vs. the friendly side , what is the friendly side's minimax strategy, what is the class of admissible strategies for each side, etc, noting all answers must depend upon the allowable classes of choices for the primitive relations $p_{ALDP}(AX,C^3)$.

Such a game can be of great use in the design and study of sensitivity of outcomes for C^3 processes, provided that these questions can be addressed using feasible computations. This of course depends on whether (3.5) and hence (3.11) can be effectively limited in complexity.

A summary of steps (i)-(v) is given below in Figure 6:

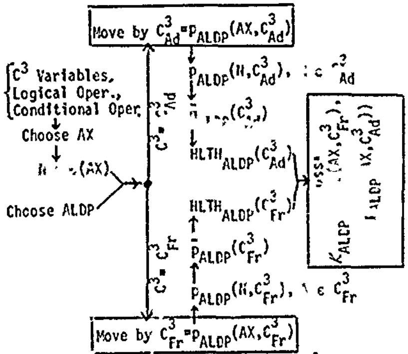


Figure 6. Functional Development of C^3 Decision Game Γ.

In the next section, a key part of the development of Γ is given: the choice of axioms AX and the resulting formal structure for node state evaluation as in step (i).

4. FORMAL THEORY AND EVALUATION FOR THE EVOLUTION OF A TYPICAL C^3 NODE STATE

In the past, a number of attempts have been made to develop formal theories for various scientific and social disciplines which traditionally had not been analyzed from such viewpoints. See, e.g., Woodger [22] for biological systems and Carnap [23], Part Two, for other applications including legal and social considerations. In this paper, a rather simplified list of axioms in non-quantifier propositional form is proposed, characterizing the essential features required to evaluate dynamically evolving C^3 node states. In turn, as outlined in section 3, these can be of use in developing a C^3 decision game. These axioms are of three kinds: logical connections, properties of conditioning, and reduction of relations among C^3 variables, or sufficiency conditions. The axiom collection AX is summarized in Table II as follows:

Table II . Formal Language Description of a C³ Node Evolution:

Equality Symbol: $\approx$

Constants: Ω, ϕ

Dummy Variables: α, β, γ

Specific Variables: N, R, S, T

N and T can be partitioned into subvectors, e.g.,

$N = (\#WP_1, \#WP_2, \#WP_3, \#TROOP, EQMO, \widehat{INFO})$

$T = (DET, ALG, HYP, FUS, CONS, DEC)$

Operators: $(\,)^+, (\,)^-, (\,)_0, (\,|\,), \&, \vee, DOM, \epsilon$

<u>General Axioms:</u> For all α, β, γ, and for $* = \&, \vee$:

<u>Ring structure</u> for &,v:

$\alpha * \beta \approx \beta * \alpha, \quad \alpha * (\beta * \gamma) \approx (\alpha * \beta) * \gamma,$

$\alpha \& \phi \approx \phi, \quad \alpha \& \Omega \approx \alpha \approx \alpha \vee \phi, \quad \alpha \vee \Omega \approx \Omega,$

$\alpha \& (\beta \vee \gamma) \approx (\alpha \& \beta) \vee (\alpha \& \gamma).$

<u>General Axioms:</u> For all α, β, γ, and for $* = \&, \vee$:

<u>Implicative/Conditional structure</u> for &,v:

$(\alpha|\Omega) \approx \alpha, \quad (\alpha|\beta) \approx (\alpha \& \beta|\beta),$

$(\alpha \& \beta|\gamma) \approx (\alpha|\beta \& \gamma) \& (\beta|\gamma),$

$(\alpha * \beta|\gamma) \approx (\alpha|\gamma) * (\beta|\gamma),$

$\bigvee_{\alpha \epsilon DOM(\alpha)} \alpha \approx \Omega.$

<u>Sufficiency Axioms:</u> For $\widetilde{N} \stackrel{d}{=} (N \& N^{--} \& N^{--} \& \ldots \& N_0)$:

$(N^{++}|R^{++} \& T^+ \& N^+ \& S \& R^- \& \widetilde{N}) \approx (N^{++}|R^{++} \& N^+),$

$(R^{++}|T^+ \& N^+ \& S \& R^- \& \widetilde{N}) \approx (R^{++}|DEC^+ \& N^+).$

$(T^+|N^+ \& S \& R^- \& \widetilde{N}) \approx (T^+|N^+)$

$(N^+|S \& R^- \& \widetilde{N}) \approx (N^+|S \& N),$

$(S|R^- \& \widetilde{N}) \approx (S|R^-),$

$(R^-|\widetilde{N}) \approx (R^-|N)$

The ring structure axioms are the minimal ones necessary to characterize conjunction and disjunction. (Note, as usual, no inverses are postulated.) The [illegible]oms for conditioning, while reflecting well-know[illegible] [illegible]operties, such as for probability interpretations, are new in that they refer to <u>measure-free</u> conditioning (see discussion later in this section). The sufficiency axioms show the basic relations among the C^3 variables of interest, based on sequence of events as outlined in Figures 2 and 3.

The axiom set in Table II leads to a particular form of eq. (3.5) describing recursively node state evolution:

<u>Theorem</u> 4.1.

Under the assumptions in Table II:

$$N^{++} = \bigvee_{\substack{R^{++} \epsilon DOM(R^{++}), \\ T^+ \epsilon DOM(T^+),\ S \epsilon DOM(S), \\ N^+ \epsilon DOM(N^+),\ R^- \epsilon DOM(R^-), \\ \widetilde{N} \epsilon DOM(\widetilde{N})}} \left((N^{++}|R^{++} \& T^+ \& N^+ \& S \& R^- \& \widetilde{N}) \& (R^{++} \& T^+ \& N^+ \& S \& R^- \& \widetilde{N})\right) \tag{4.1}$$

$$= \bigvee_{\substack{R^{++} \epsilon DOM(R^{++}),\ T^+ \epsilon DOM(T^+), \\ N^+ \epsilon DOM(N^+),\ S \epsilon DOM(S),\ R^- \epsilon DOM(R^-),\ N \epsilon DOM(N)}} \left((N^{++}|R^{++} \& N^+) \& (R^{++}|DEC^+ \& N^+) \& (T^+|N^+) \& (N^+|S \& N) \& (S|R^-) \& (R^-|N) \& N\right).$$

Regrouping terms above, compatible with Table II:

$$N^{++} = \bigvee_{N \epsilon DOM(N)} ((N^{++}|N) \& N), \tag{4.2}$$

$$(N^{++}|N) = \bigvee_{\substack{R^{++} \epsilon DOM(R^{++}) \\ N^+ \epsilon DOM(N^+)}} \left((N^{++}|R^{++} \& N^+) \& (R^{++}|N^+) \& (N^+|N)\right), \tag{4.3}$$

where

$$(N^+|N) = \bigvee_{R^- \epsilon DOM(R^-)} ((N^+|R^- \& N) \& (R^-|N)), \tag{4.4}$$

$$(R^{++}|N^+) = \bigvee_{T^+ \epsilon DOM(T^+)} ((R^{++}|DEC^+ \& N^+) \& (T^+|N^+)), \tag{4.5}$$

$$(N^+|R^- \& N) = \bigvee_{S \epsilon DOM(S)} ((N^+|S \& N) \& (S|R^-)), \tag{4.6}$$

$$\begin{aligned}(T^+|N^+) = {} & (DEC^+|CONS^+ \& FUS^+ \& HYP^+ \& ALG^+ \& DET^+ \& N^+) \\ & \& (CONS^+|FUS^+ \& HYP^+ \& ALG^+ \& DET^+ \& N^+) \\ & \& (FUS^+|HYP^+ \& ALG^+ \& DET^+ \& N^+) \\ & \& (HYP^+|ALG^+ \& DET^+ \& N^+) \\ & \& (ALG^+|DET^+ \& N^+) \& (DET^+|N^+) .\end{aligned} \tag{4.7}$$

<u>Proof</u>: Details omitted, but straightforward application of the usual rules of substitution and the axioms in Table II.

∎

Table III presents interpretations of the formal language used in Table II for presenting the axioms and consequent structure of node state evolution in Theorem 4.1:

Table III. Interpretations of the Formal Language for C^3 Node Evolution

N	= Node state vector, T = Node structure
R	= Response vector, S = "Signal" vector
$(\ )^+$	= Positive time shift to new phase
$(\ )^-$	= Negative time shift to old phase
$(\ )_0$	= Initialization of state (time-wise)
$(\ \vert\ )$	= Implication or conditioning
$\&$	= AND, $\vee$ = OR, $(\ )'$ = NOT (explained earlier)
DOM	= Domain of possible values
ε	= Set membership relation as used before
ϕ	= Null set
Ω	= Universal set

Thus, using the interpretation in Table II, compatible with Figure 3,

$$(R^{++}|T^+\&N^+) = \text{response following processing,} \tag{4.8}$$

$$(N^{++}|R^{++}\&N^+) = \text{new node state due to its sending out response ,} \tag{4.9}$$

$$(T^+|N^+) = \text{processing data,} \tag{4.10}$$

$$(N^{++}|N) = \text{full cycle of node change due to "signals" received, over all possible processing, and responses.} \tag{4.11}$$

etc.

Following the five-step procedure presented in section 3 for developing the overall decision game, the next stage requires the numerical evaluation of N^{++} given in Theorem 4.1 through choice of a suitable ALDP.

Remark 4.1

It should be noted that one can verify readily that all boolean algebras- and more generally, all browerian lattices- satisfy the axioms in Table II. Hence, all four examples of ALDP's given in (3.6)-(3.9) can be used to evaluate Theorem 4.1. For background and discussions in considering the most appropriate ALDP for a given situation, see [18] and [30]. Thus, if PL were chosen as the ALCP by assuming only stochastic relations should be used, then (4.3) becomes

$$p(N^{++}|N) = \int\limits_{\begin{pmatrix}\text{over all}\\ R^{++}\varepsilon DOM(R^{++}),\\ N^+\varepsilon DOM(N^+)\end{pmatrix}} p(N^{++}|R^{++},N^+)\cdot p(R^{++}|N^+)\cdot p(N^+|N)dR^{++}dN^+ \tag{4.12}$$

yielding, in turn, the counterpart of (4.2)

$$p(N^{++}) = \int\limits_{N\varepsilon DOM(N)} p(N^{++}|N)\cdot p(N)dN\ . \tag{4.13}$$

Or, if FL were chosen as the ALCP, by assuming only fuzzy relations should be used, then (4.3) becomes

$$poss(N^{++}|N) = \max_{\begin{pmatrix}\text{over all}\\ R^{++}\varepsilon DOM(R^{++}),\\ N^+\varepsilon DOM(N^+)\end{pmatrix}}(\min(poss(N^{++}|R^{++},N^+),poss(R^{++}|N+)\ poss(N^+|N))), \tag{4.14}$$

yielding, in turn, the counterpart of (4.2)

$$poss(N^{++}) = \max_{N\varepsilon DOM(N)}(\min(poss(N^{++}|N),poss(N))), \tag{4.15}$$

One could also choose combinations of PI and FL or other ALDP's. (Again, see [17].)

Similarly, by applying e.g., PL, one can evaluate in turn

$$p(N^+|N) = \int\limits_{R^-\varepsilon DOM(R^-)} p(N^+|R^-,N)\cdot p(R^-|N)dR^- \tag{4.16}$$

$$p(R^{++}|N^+) = \int\limits_{T^+\varepsilon DOM(T^+)} p(R^{++}|DEC^+,N^+)\cdot P(T^+|N^+)dT^+ \tag{4.17}$$

$$p(N^+|R^-,N) = \int\limits_{S\varepsilon DOM(S)} p(N^+|S,N)\cdot p(S|R^-)dS \tag{4.18}$$

$$p(S|R^-) = p(v=S-f(R^-)), \tag{4.19}$$

when the nonlinear additive regression relation holds

$$S = f(R^-) + v, \tag{4.20}$$

where v is a random vector representing additive medium error betweeb responses and "signals" and f is a known function representing medium distortion.

Thus, one can identify the probabilities in (4.12),(4.13),(4.16)-(4.19) with $P_{PL}(AX,C^3)$ as given in step (ii), following (3.11). Hence, by specifying $P_{PL}(AX,C^3)$ for each process, friendly or adversary, one can then proceed, at least in theory, with the construction of the C^3 decision game, as outlined in steps (iii)-(v), section 3. Similar remarks hold for the use of FL or any other of the ALDP's.

In particular, for FL, it follows from Theorem 4.1 that for each C^3 process, $P_{PL}(AX,C^3)$ is determined by first specifying 12 relatively primitive relations. Thus, one can write:

$$\left.\begin{array}{r} p(N^{++}|R^{++},N^+),\\ p(R^{++}|DEC^+,N^+),\\ p(DEC^+|CONS^+,FUS^+,HYP^+,ALG^+,DET^+,N^+),\\ p(CONS^+|FUS^+,HYP^+,ALG^+,DET^+,N^+),\\ p(FUS^+|HYP^+,ALG^+,DET^+,N^+),\\ p(HYP^+|ALG^+,DET^+,N^+),\\ p(ALG^+|DET^+,N^+),\\ p(DET^+|N^+),\\ p(N^+|S,N),\\ p(S|R^-),\\ p(R^-|N),\\ p(N_0) \end{array}\right\} = P_{PL}(AX,C^3) \tag{4.21}$$

Of course, it is understood in (4.21) that the relations - except for the last - must be specified for each time cycle of node processing input [illegible] to output response.

In another direction, noting that many of the relations in Table II and in subsequent equations are in conditional form, it is of some interest to inquire whether this theory can treat [illegible] or conditional forms when the antecedents need not be identical, generalizing the axiom in Table II

$$(\alpha*\beta|\gamma) = (\alpha|\gamma)*(\beta|\gamma) \tag{4.22}$$

to the case

$$(\alpha * \beta | \rho) = (\alpha | \gamma) * (\beta | \eta), \tag{4.23}$$

where now η and γ need not be the same and where ρ is some computable function (another event or set) of $\alpha, \beta, \gamma, \eta$, for $* = \&, v$. For example, one may wish to determine Q, and by choosing PL, p(Q), for some appropriate joint probability measure p, where

$$Q = ((a|b) \& (c|d)) \vee (e|f), \tag{4.24}$$

where

a = a(x) = enemy will move up about x troops tomorrow; x=0,50,100,150.
b = b(y) = it will y tomorrow; y=be clear, snow,rain.
c = c(z) = enemy will use Pass z to approach us; z=I,II,III,IV.
d = d(r,s)= morale of enemy node 17 is at level r and number of their troops left is s ; r=very low, low, medium,high, very high, s=0,100,200,300.
e = e(w) = enemy will w tomorrow; w=surrender,not surrender.
f = f(q) = enemy overall damage level is q ; q=0,1,2,...,10.

Of course, if the antecedents in eq. (4.24) were all the same, then no real problem would arise, since for example it is readily justified that for any choice of ALOP - certainly for PL ! - that

$$\begin{aligned} Q &= ((a|d) \& (c|d)) \vee (e|d) \\ &= (((a\&c) \vee e)|d) , \end{aligned} \tag{4.25}$$

even though normally one does not talk about such *measure-free* entities (up to now). Indeed, since the goal is the evaluation of Q, for PL, choosing a probability measure p over all the relevant events, one would usually evaluate Q as simply

$$\begin{aligned} p(Q) &= p(((a|d) \& (c|d)) \vee (e|d)) \\ &= p(((a\&c) \vee e)|d) \\ &= p(((a\&c) \vee e)) \& d)/p(d), \end{aligned} \tag{4.26}$$

etc., assuming p(d)>0.

But the point of the above example given in (4.24) is that the antecedents in the conditional forms are <u>not</u> in general identical! What to do?

Contrary to popular belief [author's note: this author and his colleague Prof. H.T. Nguyen, Math. Dept. New Mexico State University, Las Cruces, have undertaken an extensive informal survey of the probability community- both applied and theoretical-resulting in the following conclusions - see [24],[25]]: there is <u>no</u> systematic and mathematically sound procedure for computing p(Q) (or Q, for that matter) in (4.24) or, in fact, for any similar problem!

Indeed, there do exist "folk" remedies to this situation which roughly speaking reduce to either identifying conditioning with material implication, forcing conditioning to be a closed operation over the boolean algebra of events, or identifying conditional events as marginal to a universal joint event having a fixed antecedent common to these marginal ones. In either case, serious difficulties arise - either mathematically or computationally. For a satisfactory solution to this problem, see [24],[25], where a sound and complete ALOP (among other properties) is developed, compatible with, and extending,classical PL, called CPL (conditional probability logic).

One consequence of the calculus of operations in CPL is that the evaluation of Q in (4.24) becomes

$$Q = (\alpha | \beta) , \tag{4.27}$$

resulting in the value

$$p(Q) = p(\alpha)/p(\beta), \tag{4.28}$$

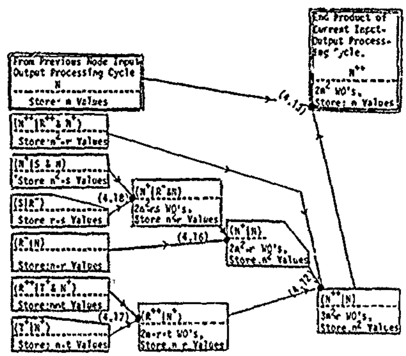
where

$$\alpha \overset{d}{=} (a\&b\&c\&d) \vee (e\&f), \tag{4.29}$$

$$\beta \overset{d}{=} \alpha \vee (((a'\&b) \vee (c'\&d) \vee (b\&d))\&f), \tag{4.30}$$

differing considerably from the "folk" approaches.

Returning to the construction of the C^3 decision game, again note that by utilizing the evaluations as in (4.12),(4.13),(4.16)-(4.19) for PL, or (4.14),(4.15), and analogous calculations for FL, etc., one can then fully evaluate the dispersion of N^+ in (4.1) recursively. However, as gleaned from Figure 7 below, regardless of the ALOP chosen, even the basic scheme for evaluating a typical single node processing cycle from input to output- without further decomposition of the C^3 variables and subvariables, especially that of T^+ and N^+, still requires on the order of q^4 matrix/vector addition or multiplication operations, when,for simplicity, the domains of possible values for N,R,S,T have the same cardinality q. Under the same assumptions, one must also store about $2 \cdot q^3$ domain values in order to accomplish all the computations.



Totals: #(Initial Stored Values) = $n(1+r)(1+t) + n^2(r+s) + rs$
#(WO's) = $n^2 \cdot (2 + 5r + 2rs) + 2nrt$
( ) Counts as 1 Weighted Operation (WO)
( )*( ) Counts as 2 Weighted Operations (WO's), etc
$n \overset{d}{=} \mathrm{Card}(\mathrm{DOM}(N))$, $r \overset{d}{=} \mathrm{Card}(\mathrm{DOM}(R))$, $s \overset{d}{=} \mathrm{Card}(\mathrm{DOM}(S))$, $t \overset{d}{=} \mathrm{Card}(\mathrm{DOM}(T))$

Figure 7 Computation Requirements for High-Level Processing of a Typical Node Input-Output Processing Cycle.

Needless to say, the evaluation procedure can become extremely tedious due to the multiplicative forms of the terms involved compounded with the lengthy iterative disjunction operations over the domains of the variables. In the next section, a possible solution to this problem is offered for the case of PL.

5. UTILIZATION OF LINEAR-GAUSSIAN MIXTURE APPROXIMATIONS

As stated before, all modeling of C^3 processes in general must take into account the very real problem of accuracy of model vs. complexity of calculations. One possible resonable solution to this difficulty - at least for the choice of ALDP = PL - involves the use of some fundamental approximation and representation theorems, which are useful in their own right.

First, before considering a theorem based in part on material in [26], the following definitions and notational conventions will be introduced:

Letting r be any positive integer and $\mathbb{R}^r$ the real r-dimensional euclidean space, with $[0,1]$ being the real unit interval, call any cumulative probability distribution function (cdf) $F:\mathbb{R}^r \to [0,1]$ well-behaved iff in its Jordan-Lebesgue decomposition (see, e.g. Feller [27], pp. 135-140) no singular component appears, the discrete component distribution, if present is at most finite, and the absolutely continuous component admits a probability density function (pdf) which is bounded and uniformly continuous over $\mathbb{R}^r$. Thus,

$$F = \lambda \cdot F^{(1)} + (1-\lambda)\cdot F^{(2)}, \tag{5.1}$$

where $F^{(j)}:\mathbb{R}^r \to [0,1]$ corresponds to a finitely discrete probability measure for $j=1$ and an absolutely continuous probability measure for $j=2$, with pdf $f^{(2)}$ being bounded and uniformly continuous over $\mathbb{R}^r$.

Clearly, large classes of common probability measures have cdf's satisfying these criteria.

For notational purposes, also let

$$F^{(1)} = \sum_{j\in J} p_j \cdot \delta(\cdot - \mu_j) , \tag{5.2}$$

where $\mu_j \in \mathbb{R}^r$ is a mass point for $F^{(1)}$, δ is the dirac delta function, p_j represents the probability of occurence relative to $F^{(1)}$ at μ_j, $0 \le p_j \le 1$, $j \in J$, $\sum_{j\in J} p_j = 1$, J being finite or even vacuous. By convention, if $J=\phi$, $F^{(1)}$ is vacuous and F reduces to the purely absolutely continuous cdf $F^{(2)}$.

Depending on the context used, denote g_Σ to mean either the pdf, cdf, or probability distribution, corresponding to an r-dimensional gaussian distribution with mean 0_r and positive definite covariance matrix Σ. Next, for any sequences

$$\underset{\sim}{\tau} \overset{d}{=} (\tau_1,\dots,\tau_m),\ \underset{\sim}{\mu} \overset{d}{=} (\mu_1,\dots,\mu_m),\ \underset{\sim}{\Sigma} \overset{d}{=} (\Sigma_1,\dots,\Sigma_m), \tag{5.3}$$

$0 \le \tau_i \le 1, \tau_1 + \cdots + \tau_m = 1$, noting the pdf for g_Σ is

$$g_\Sigma(x) = ((2\pi)^r \cdot \mathrm{determ}(\Sigma))^{-\frac{1}{2}} \cdot e^{-(\frac{1}{2}) x^T \cdot \Sigma^{-1} \cdot x}, \tag{5.4}$$

for all $x \in \mathbb{R}^r$, so that $g_\Sigma(\cdot - \mu)$ corresponds to an r-dimensional gaussian distribution with mean μ and covariance matrix Σ, let the finite gaussian mixture

$$g_{\underset{\sim}{\tau},\underset{\sim}{\mu},\underset{\sim}{\Sigma}} \overset{d}{=} \sum_{i=1}^{m} \tau_i \cdot g_{\Sigma_i}(\cdot - \mu_i), \tag{5.5}$$

which can represent a pdf, cdf, or probability distribution, depending on the context.

With all of the above established, a uniform approximation theorem can now be stated.

Theorem 5.1.

Let $F:\mathbb{R}^r \to [0,1]$ be any given well-behaved cdf with possibly vacuous J containing mass points μ_j, $j \in J$, in accordance with the above notation.

Then, F can be arbitarily uniformly closely approximated over $\mathbb{R}^r$, except at all of F's mass points μ_j, $j \in J$, by a sequence of cdf's which are finite gaussian mixtures. Denote this relation as

$$g_{\underset{\sim}{\tau},\underset{\sim}{\mu},\underset{\sim}{\Sigma}} \approxeq F. \tag{5.6}$$

Proof: First consider separately the truth of the theorem for the purely finitely discrete case $F=F^{(1)}$. Clearly in this case, the sequence

$$\underset{\sim}{g}_1 \overset{d}{=} (g_{\underset{\sim}{\tau}_1,\underset{\sim}{\mu}_1,\underset{\sim}{\Sigma}_{1,n}})_{n=1,2,\dots} \tag{5.7}$$

where

$$\underset{\sim}{\tau}_1 \overset{d}{=} (p_j)_{j\in J},\ \underset{\sim}{\mu}_1 \overset{d}{=} (\mu_j)_{j\in J},\ \underset{\sim}{\Sigma}_{1,n} \overset{d}{=} ((1/n)\cdot I_r)_{j\in J}, \tag{5.8}$$

obviously uniformly approaches $F^{(1)}$ as a cdf, except over $\underset{\sim}{\mu}_1$.

Next, consider the validity of the theorem for the purely absolutely continuous case $F=F^{(2)}$. Now, from [26], Theorem 1, there exists a sequence of finite gaussian mixtures

$$\underset{\sim}{g}_2 \overset{d}{=} (g_{\underset{\sim}{\tau}_{2n},\underset{\sim}{\mu}_{2n},\underset{\sim}{\Sigma}_{2n}})_{n=1,2,\dots} , \tag{5.9}$$

which approaches $f^{(2)}$ in L^1-norm. But using the basic absolute inequality relation for all $A \subseteq \mathbb{R}^r$ (measurable)

$$\left| \int_{x\in A} g_{\underset{\sim}{\tau}_{2n},\underset{\sim}{\mu}_{2n},\underset{\sim}{\Sigma}_{2n}}(x)dx - \int_{x\in A} f^{(2)}(x)dx \right| \le \int_{x\in\mathbb{R}^r} |g_{\underset{\sim}{\tau}_{2n},\underset{\sim}{\mu}_{2n},\underset{\sim}{\Sigma}_{2n}}(x) - f^{(2)}(x)|\, dx, \tag{5.10}$$

and letting $A = A_s \overset{d}{=} \{x : x \in \mathbb{R}^r \text{ and } x \le s\}$, for any $s \in \mathbb{R}^r$, it follows that as cdf's, $\underset{\sim}{g}_2$ approaches $F^{(2)}$ uniformly over $\mathbb{R}^r$.

Finally, since F is a linear combination of $F^{(1)}$ and $F^{(2)}$, it follows that one can let in (5.6)

$$g_{\underset{\sim}{\tau},\underset{\sim}{\mu},\underset{\sim}{\Sigma}} \overset{d}{=} \lambda \cdot \underset{\sim}{g}_1 + (1-\lambda)\cdot \underset{\sim}{g}_2 , \tag{5.11}$$

which is again a legitimate finite gaussian mixture. ∎

Remark 5.1.

It should be noted that in the construction of $\underset{\sim}{g}_2$ above, the $\underset{\sim}{\tau}_{2n}$ may approach zero, a not necessarily desirable property due to the resulting fluctuations of form of the pdf's. However, in [26] and in [29] alternative forms can be utilized. Of course, in the case of $\underset{\sim}{g}_1$, this cannot be avoided. ∎

The next theorem establishes a unique linear regression relation among any given pair of random vectors, provided sufficient joint second moments exist. Although this is a basic result, appearing in many places in one form or another (see, e.g. [28], sections 3.3, 3.4), it is surprisingly not often in the full form to be given here with direct application to reducing nonlinear relations to "exact" linear ones, without employing approximating expansions.

Theorem 5.2.

Let $(\Omega, \mathcal{A}, p)$ be a probability space and $X:\Omega \to \mathbb{R}^k$ and $Y:\Omega \to \mathbb{R}^m$ be random vectors such that $\mathrm{Cov}\binom{Y}{X}$ exists and Cov(X) is positive definite.

Then, there exists constant m by k (real) matrix B, called the regression transition matrix from Y to X, and random vector $W:\Omega\to\mathbb{R}^m$ such that the linear regression relation holds

$$Y = B\cdot X + W\ , \qquad (5.12)$$

such that X and W are uncorrelated. W and B are uniquely determined with

$$W = Y - B\cdot X\ ,\quad B = \mathrm{Cov}(Y,X)\cdot\mathrm{Cov}^{-1}(X), \qquad (5.13)$$

$$\mathrm{Cov}(W) = \mathrm{Cov}(Y)-\mathrm{Cov}(Y,X)\cdot\mathrm{Cov}^{-1}(X)\cdot\mathrm{Cov}^{T}(Y,X), \qquad (5.14)$$

and a constant $A=E(W)=E(Y)-B\cdot E(X)$, etc. can be added to the RHS of (5.12), provided W is replaced by the zero mean random vector $W-E(W)$.

Proof: The result is self evident, once equation (5.13) is noted, using the calculus of multivariate moments and matrices. ∎

Remark 5.2.

Note that Theorem 5.2 can be applied to produce an "exact" linearization of nonlinear relations such as given below where X and V are uncorrelated random vectors with $E(V)=0_r$ so that

$$Y = f(X) + V\ , \qquad (5.15)$$

where $f:\mathbb{R}^k\to\mathbb{R}^m$ is some fixed (measurable) function. The result immediately follows that (5.12) becomes here

$$Y = B\cdot X + W\ ,$$

but where

$$B = \mathrm{Cov}(f(X),X)\cdot\mathrm{Cov}^{-1}(X). \qquad (5.16)$$

Now, if one were to apply a standard linearization approach to the model in (5.15), by linearizing $f(X)$,

$$f(X) \simeq A_o + B_o\cdot(X-X_o)\ , \qquad (5.17)$$

where

$$A_o \overset{d}{=} f(x_o)\ ,\quad B_o \overset{d}{=} (df(X)/dX)_{X=x_o}\ , \qquad (5.18)$$

for some constant $x_o \in \mathbb{R}^k$ such as $E(X)$. Substituting (5.17) into (5.15) yields the approximation

$$Y \simeq A_o + B_o\cdot(X-x_o) + V. \qquad (5.19)$$

Next, if one defines the true error in (5.19) as

$$Z \overset{d}{=} Y - A_o - B_o\cdot(X-x_o)$$
$$= W + (B-B_o)\cdot X + C\ , \qquad (5.20)$$

where

$$C \overset{d}{=} B_o\cdot x_o - A_o\ , \qquad (5.21)$$

then by the above equations and standard matrix manipulations of covariances and means

$$\begin{aligned} E(Z\cdot Z^T) &= \mathrm{Cov}(W) + (B-B_o)\cdot\mathrm{Cov}(X)\cdot(B-B_o)^T + D\cdot D^T \\ &\gtrsim \mathrm{Cov}(W) \\ &= \mathrm{Cov}(V) + H \\ &\gtrsim \mathrm{Cov}(V), \end{aligned} \qquad (5.22)$$

where

$$D \overset{d}{=} E(W) + (B-B_o)\cdot E(X) + C\ , \qquad (5.23)$$

$$H \overset{d}{=} \mathrm{Cov}(f(X))-\mathrm{Cov}(f(X),X)\cdot\mathrm{Cov}^{-1}(X)\cdot\mathrm{Cov}^{T}(f(X),X). \qquad (5.24)$$

$\gtrsim$ indicates positive semidefinite partial ordering among all m by m matrices, where one has $M_1 \gtrsim M_2$ iff M_1-M_2 is a positive semidefinite matrix.

Thus, (5.22) shows in a quantitative manner that, though one pays a penalty (H) for using the exactly linearized form in (5.12), (5.16), in place of the original nonlinear additive form in (5.15), the former is still a more accurate approach than use of the standard linearization (without reiteration, of course) in (5.19).

Finally, it should be noted that the above remarks are valid for the exact linearization of the conditional "signal" relation in (4.19), (4.20). ∎

Corollary 5.1.

Suppose $(\Omega,\mathcal{A},p)$ is a probability space and $X:\Omega\to\mathbb{R}^k$, $Y:\Omega\to\mathbb{R}^m$ are random vectors such that $\binom{Y}{X}$ has a well-behaved cdf over $\mathbb{R}^{m+k}$ and $\mathrm{Cov}\binom{Y}{X}$ exists with Cov(X) positive definite.

Then:

(i) An exact linear regression relation holds as in eq. (5.12) between X and Y.

(ii) For conditional random vector $(Y|X)$, for all outcomes y of Y and x of X, for some τ_1,μ_1,Σ_1,

$$g_{\tau_1,\mu_1,\Sigma_1}(y-B\cdot x) \approxeq F(Y=y|X=x). \qquad (5.25)$$

(iii) For unconditional random vector X at any possible outcome x, for some τ_2,μ_2,Σ_2,

$$g_{\tau_2,\mu_2,\Sigma_2}(x) \approxeq F(X=x). \qquad (5.26)$$

(iv) For the marginal integrated-out cdf of Y, at any outcome y

$$g_{\tau,\mu,\Sigma}(y) \approxeq F(y) = \int_{x\in\mathbb{R}^k} F(Y=y|X=x)dF(X=x), \qquad (5.27)$$

where letting

$$\tau_i \overset{d}{=} (\tau_{ij})_{j\in J_i},\ \mu_i \overset{d}{=} (\mu_{ij})_{j\in J_i},\ \Sigma_i \overset{d}{=} (\Sigma_{ij})_{j\in J_i}, \qquad (5.28)$$

$i=1,2$,

$$\tau \overset{d}{=} (\underbrace{\tau_{1j_1}\cdot\tau_{2j_2}}_{\overset{d}{=}\tau(j_1,j_2)})_{j_i\in J_i,\ i=1,2},\quad \mu \overset{d}{=} (\underbrace{B\cdot\mu_{2j_2} + \mu_{1j_1}}_{\overset{d}{=}\mu(j_1,j_2)})_{j_i\in J_i,\ i=1,2},$$

$$\Sigma \overset{d}{=} (\underbrace{B\cdot\Sigma_{2j_2}\cdot B^T + \Sigma_{1j_1}}_{\overset{d}{=}\Sigma(j_1,j_2)})_{j_i\in J_i,\ i=1,2}, \qquad (5.29)$$

Proof: Note that all linear transforms on $\binom{Y}{X}$ preserve the well-behaved property. Then, applying Theorems 5.1 and 5.2 yields (i)-(iii), with (iv) obtained by multiplying out all mixture terms and using the well-known convolution of gaussian pdf's which here takes the form, for any typical term

$$g_{\Sigma(j_1,j_2)}(y-\mu(j_1,j_2)) = \int_{x\in\mathbb{R}^k} g_{\Sigma_{1j_1}}(y-Bx-\mu_{1j_1})\,g_{\Sigma_{2j_2}}(x-\mu_{2j_2})dx \qquad (5.30)$$

∎

Before establishing the main result, actually a corollary of the above two theorems, the following additional notation will be useful:

As before, suppose Y and X are random vectors with $Y:\Omega\to\mathbb{R}^m$ and $X:\Omega\to\mathbb{R}^k$, but now where X is partitioned into

$$X = \begin{pmatrix} X_1 \\ \vdots \\ X_s \end{pmatrix},\ X_i:\Omega\to\mathbb{R}^{k_i},\ i=1,\dots,s, \qquad (5.31)$$

where necessarily $k_1 + \cdots + k_s = k$. Partition, similarly, any $x \in \mathbb{R}^k$ into

$$x = \begin{pmatrix} x_1 \\ \vdots \\ x_s \end{pmatrix}, \tag{5.32}$$

and let for any i, $i=1,\ldots,s$,

$$X^{(i)} \stackrel{d}{=} \begin{pmatrix} X_i \\ \vdots \\ X_s \end{pmatrix}, \tag{5.33}$$

so that one notes

$$X^{(1)} = X, \quad X^{(s)} = X_s. \tag{5.34}$$

Also, extend the notation, where

$$X_0 \stackrel{d}{=} Y, \quad X_{s+1} \stackrel{d}{=} 0, \quad X^{(s+1)} \stackrel{d}{=} \emptyset, \tag{5.35}$$

etc., with similar remarks holding for x.

Corollary 5.2.

Suppose the hypotheses of Corollary 5.1 hold, with the notation introduced in (5.31)-(5.35) valid. Then:

(i) For each i, $i=0,1,\ldots,s$, the exact linear regression relation holds

$$X_i = B_i \cdot X^{(i+1)} + W_i, \tag{5.36}$$

where $X^{(i+1)}$ and W_i are uncorrelated random vectors and each B_i is a constant k_i by $k_{i+1}+\cdots+k_s$ regression transition matrix, obtained analogous to B in (5.13). Denote the partitioning of each B_i

$$B_i = (B_{i,i+1}, B_{i,i+2}, \ldots, B_{i,s}) = B^{(i,i+1)}, \tag{5.37}$$

$$B^{(i,i+t)} \stackrel{d}{=} (B_{i,i+t}, \ldots, B_{i,s}), \quad t=1,\ldots,s-i. \tag{5.38}$$

(ii) In turn, there exists finite gaussian mixtures, where for all possible outcomes x_i of X_i,

$$g_{\underline{\tau}_i,\underline{\mu}_i,\underline{\Sigma}_i}(x_i - B_i \cdot x^{(i+1)}) \cong F(X_i = x_i \mid X^{(i+1)} = x^{(i+1)}), \tag{5.39}$$

where $\underline{\tau}_i$, $\underline{\mu}_i$, $\underline{\Sigma}_i$ are all formally the same as the corresponding values in (5.28), except here, $i = 0, 1, 2, \ldots, s$, noting for $i=s$, RHS of (5.36) is identical to $X_s = X_s$ and, in effect, B_s is 0.

(iii) Finally, the marginal, integrated-out cdf of Y, compatible with (i) and (ii), is uniformly approximated as, for any outcome y of Y,

$$g_{\underline{\tau},\underline{\mu},\underline{\Sigma}}(y) \cong F(y) = \int_{x\in\mathbb{R}^k} F(Y=y \mid X=x) \cdot \prod_{i=1}^{s} F(X_i = x_i \mid X^{(i+1)} = x^{(i+1)})\,dx, \tag{5.40}$$

where

$$\underline{\tau} \stackrel{d}{=} \underbrace{(\tau_{0,j_0} \cdots \tau_{s,j_s})}_{\stackrel{d}{=}\tau(\underline{j})}{}_{j_i \in J_i,\ i=0,\ldots,s}, \quad \underline{\mu} \stackrel{d}{=} \underbrace{\left(\sum_{i=0}^{s} B^{(i)} \cdot \mu_{i,j_i}\right)}_{\stackrel{d}{=}\mu(\underline{j})}{}_{j_i \in J_i,\ i=0,\ldots,s},$$

$$\underline{\Sigma} \stackrel{d}{=} \underbrace{\left(\sum_{i=0}^{s} B^{(i)} \cdot \Sigma_{i,j_i} \cdot B^{(i)T}\right)}_{\stackrel{d}{=}\Sigma(\underline{j})}{}_{j_i \in J_i,\ i=0,\ldots,s}, \quad \underline{j} \stackrel{d}{=} (j_0,\ldots,j_s) \in J_0 \times \cdots \times J_s, \tag{5.41}$$

and where

$$B^{(0)} \stackrel{d}{=} I_m, \quad B^{(1)} \stackrel{d}{=} B_{0,1}, \quad B^{(2)} \stackrel{d}{=} B_{0,2} + B_{0,1} \cdot B_{1,2},$$

$$B^{(3)} \stackrel{d}{=} B_{0,3} + B_{0,2} \cdot B_{2,3} + B_{0,1} \cdot B_{1,3} + B_{0,1} \cdot B_{1,2} \cdot B_{2,3},$$

and more generally,

$$B^{(i)} \stackrel{d}{=} \sum_{\substack{\text{over all possible} \\ \text{paths } 0<i_1<\cdots<i_t<i, \\ t=1,2,\ldots}} B_{0,i_1} \cdot B_{i_1,i_2} \cdots B_{i_t,i} = \sum_{\ell=1}^{i} B^{(i-\ell)} \cdot B_{i-\ell,i}. \tag{5.42}$$

Proof: (i) and (ii) follow from Corollary 5.1. For (iii), consider

$$F(y) = \int_{x\in\mathbb{R}^k} \left(\prod_{i=0}^{s} F(X_i = x_i \mid X^{(i+1)} = x^{(i+1)})\right) dx$$

$$\cong \int_{x\in\mathbb{R}^k} \left(\prod_{i=0}^{s} g_{\underline{\tau}_i,\underline{\mu}_i,\underline{\Sigma}_i}(x_i - B_i \cdot x^{(i+1)})\right) dx$$

$$= \sum_{\underline{j}\in J_0\times\cdots\times J_s} (\quad h(x_0;\underline{j})), \tag{5.43}$$

where

$$h(x_0;\underline{j}) \stackrel{d}{=} \int_{x\in\mathbb{R}^k} \left(\prod_{i=0}^{s} g_{\Sigma_{i,j_i}}(x_i - B_i \cdot x^{(i+1)} - \mu_{i,j_i})\right) dx$$

$$= h_s(x_0;\underline{j};x^{(s+1)}), \tag{5.44}$$

where recursively h_i is defined as follows:

$$h_0(x_0;\underline{j};x^{(1)}) \stackrel{d}{=} g_{\Sigma_{0,j_0}}(x_0 - B_0 \cdot x^{(1)} - \mu_{0,j_0}), \tag{5.45}$$

and for $i=1,\ldots,s$,

$$h_i(x_0;\underline{j};x^{(i+1)}) \stackrel{d}{=} \int_{x_i\in\mathbb{R}^{k_i}} \left(h_{i-1}(x_0;\underline{j};x^{(i)}) \cdot g_{\Sigma_{i,j_i}}(x_i - B_i \cdot x^{(i+1)} - \mu_{i,j_i})\right) dx_i. \tag{5.46}$$

Then, beginning with $i=1$, one applies the identity in (5.30) to obtain h_1, followed in turn by a similar procedure for $i=2$, using (5.46), ... , until step $i=s$ is reached. ∎

Remark 5.3.

Note that the results in Corollary 5.2(iii) are equivalent to specifying a finite gaussian mixture, where for each mixing index $\underline{j}\in J_0\times\cdots\times J_s$, the corresponding component distribution is gaussian determining a random vector, say $U_{\underline{j}}$, which has $E(U_{\underline{j}}) = \mu(\underline{j})$, $Cov(U_{\underline{j}}) = \Gamma(\underline{j})$, values given in (5.41).

In turn, from the forms in (5.41), each $U_{\underline{j}}$ is seen to be a linear combination of statistically independent gaussian random vectors, say $U_{\underline{j},\ell}$, where

$$U_{\underline{j}} = B^{(0)} \cdot U_{\underline{j},0} + \cdots + B^{(s)} \cdot U_{\underline{j},s}, \tag{5.47}$$

and each $U_{\underline{j},\ell}$ has distribution $g_{\Sigma_{i,j_i}}(\cdot - \mu_{i,j_i})$.

Each coefficient matrix $B^{(i)}$ in the linear combination via (5.42) represents the overall regression transition matrix from Y multiplicatively up to X_i via all possible distinct path combinations of individual regression transition matrices among pairs of random vectors from $(Y, X_1, \ldots, X_s)$ connecting Y to X_i. Possible connections may exist also with Ingber's mesoscopic-macroscopic approach to C^3 modeling, where the path-integral representation relative to nonlinear nonequilibrium gaussian-markovian statistical mechanics is used. This avenue remains to be explored. (Again, see [12].) ∎

Remark 5.4.

In applying Corollary 5.2, the philosophy of approach is as follows: One does not know a priori the distribution of Y, the desired goal, but one does know - or has control in assigning - all of the intermediate or auxiliary conditional distributions, conditioning Y on X, X_1 on $X^{(2)}$, X_2 on $X^{(3)}$, ..., and finally X_s ($X^{(s+1)}$ being trivialized to make X_s an unconditional random vector). Further help in reducing calculations in (5.41),(5.42) will occur if sufficiency or markovian-like assumptions can be made, thereby causing in effect a number of individual transition matrices B_{i_1,i_2} to be zero.

In particular, consider now applying Corollary 5.2 to the evolution of node states as in steps (i) and (ii), sections 3 and 4, where ALDP=PL:

$$Y = X_0 \overset{d}{=} N^{++} = (\#WP_1^{+}, \#WP_2^{+}, \#WP_3^{+}, \#TROOP^{+}, EQMO^{+}, \widehat{INFO}^{+}), \tag{5.48}$$

$$X = \begin{pmatrix} X_1 \\ \vdots \\ X_{21} \end{pmatrix}, \quad s = 21, \tag{5.49}$$

where, replacing directly N, and hence N^+, as well as T^+, by their component subvariables, as given in (3.3),(3.4),

$$X_1 \overset{d}{=} R^{++}, \tag{5.50}$$

$$X_2 \overset{d}{=} DEC^{+}, X_3 \overset{d}{=} CONS^{+}, X_4 \overset{d}{=} FUS^{+}, X_5 \overset{d}{=} HYP^{+}, X_6 \overset{d}{=} ALG^{+}, X_7 \overset{d}{=} DET^{+}, \tag{5.51}$$

$$X_8 \overset{d}{=} \#WP_1^{+},\ X_9 \overset{d}{=} \#WP_2^{+},\ X_{10} \overset{d}{=} \#WP_3^{+}, \\ X_{11} \overset{d}{=} \#TROOP^{+}, X_{12} \overset{d}{=} EQMO^{+},\ X_{13} \overset{d}{=} \widehat{INFO}^{+}, \tag{5.52}$$

$$X_{14} \overset{d}{=} S, \tag{5.53}$$

$$X_{15} \overset{d}{=} R^{-} \tag{5.54}$$

$$X_{16} \overset{d}{=} \#WP_1,\ X_{17} \overset{d}{=} \#WP_2,\ X_{18} \overset{d}{=} \#WP_3, \\ X_{19} \overset{d}{=} \#TROOP,\ X_{20} \overset{d}{=} EQMO,\ X_{21} \overset{d}{=} \widehat{INFO}. \tag{5.55}$$

In turn, due to the assumptions in Table II, a number of transition matrices can be set equal to zero, as mentioned earlier:

$$\left.\begin{array}{l} B_{0,i},\ i=2,\ldots,7,14,\ldots,21, \\ B_{1,i},\ i=3,\ldots,7,14,\ldots,21, \\ B_{i_1,i_2},\ i_1=2,\ldots,7,\ i_2=14,\ldots,21, \\ B_{i,15},\ i=8,\ldots,13, \\ B_{14,i},\ i=16,\ldots,21 \end{array}\right\} = 0. \tag{5.56}$$

Thus, the key calculations in (4.12),(4.13), (4.16)-(4.19) can all be replaced, using (5.48)-(5.56) in Corollary 5.2 (iii).

If one considers the number of operations required in (4.12),(4.13),(4.16)-(4.19) directly to compute $p(N^{++})$, assuming equally sized domains of values, say q (as noted at the end of section 4), one obtains here on the order of q^{21}, 21 being the number of variables being integrated out in (4.1). On the other hand, the linear-gaussian mixture approximation of Corollary 5.2(iii) requires about $card(J_0)^{21}$ mixing coefficients, when $card(J_i) = card(J_0)$, for all i. In addition, for each mixture distribution, corresponding to a mixing term, there are 2x22 + 3x22 entire matrix multiplication and addition operations for obtaining the characterizing gaussian parameters: the mean and covariance matrix. There is also an upper bound of 2^{21} - reduced by use of (5.56) - number of matrix additions of matrix multiplication required to obtain $B^{(0)},\ldots,B^{(21)}$, recursively, as in (5.42). Hence for the linear-gaussian approach, a multiplicative value of $(2card(J_0))^{21}$ could be required for implementation.

Thus, if the average number of mixing coefficients can be reduced so that

$$card(J_0) \ll q/2, \tag{5.57}$$

then the linear gaussian mixture approach can be of real use. This will occur especially when the number of C^3 variables describable by single gaussian distributions or by some absolutely continuous distributions which are bimodal or at least relatively minimum in number of modes. This is opposite in kind to the situation where most C^3 variables are discrete with the only good gaussian mixture approximations being essentially the same as the original discrete distributions, but with each corresponding dirac delta function replaced by a gaussian distribution of sufficiently small covariance matrix, as in the proof of Theorem 5.1. In the latter case, (5.57) will be violated and the better approach is to stay with the original integrals, possibly discretizing them. ∎

One desirable property of gaussian mixtures is the ease in computing means and covariance matrices, once the mixing parameters are all determined. Thus, it follows easily that for Y in Corollary 5.2

$$E(Y) \approx \sum_{\underline{j} \in J_0 \times \cdots \times J_s} \tau^{(\underline{j})} \cdot \mu^{(\underline{j})}, \tag{5.58}$$

$$Cov(Y) \approx \sum_{\underline{j} \in J_0 \times \cdots \times J_s} \tau^{(\underline{j})} \cdot \left(\Sigma^{(\underline{j})} + (\mu^{(\underline{j})} - E(Y))(\mu^{(\underline{j})} - E(Y))^T\right), \tag{5.59}$$

Finally, mention must be made of another possible source of difficulty in implementing the linear gaussian mixture method. This involves the actual construction of the gaussian mixtures, mentioned previously. (Again, see [26] and [29] for techniques.)

6. CONCLUDING REMARKS

In order to implement the proposed general C^3 decision game, as outlined in the previous sections, for a given simulated scenario, one must be careful in defining the boundaries for what constitutes the relevant C^3 variables. This is a relative concept. In a given situation, a node may represent simply a single person or machine, such as a tank crew, or it may represent an entire group of tanks, depending on the desired aggregation or hierarchical level considered. In addition, before implementing the model proposed here, one must scope out what constitutes a "signal" input-output node cycle and

Table IV, Categorization of Events Occurring During Inner-Outer Air Battle.

<table>
<tr><td>NODE TYPE</td><td colspan="13">INPUT-OUTPUT NODE CYCLES OCCUR AT PHASE CHANGES WHICH ARE BY DOCTRINE AND GAME DESIGN</td></tr>
<tr><td>I:ENEMY BOMBERS</td><td>Initial</td><td>BFA$^-$ by III
BFA$^-$ by IV
BFA$^-$ by V</td><td>BFA by III
BFA by IV
BFA by V</td><td colspan="2">BFA$^+$ by III
BFA$^+$ by IV
BFA$^+$ by V</td><td colspan="2">BOMB$^-$ IV
BOMB$^-$ V</td><td colspan="2">BOMB IV
BOMB V</td><td colspan="2">BOMB$^+$IV
BOMB$^+$V</td><td colspan="2">End</td></tr>
<tr><td>II:FRIEND SURV.</td><td>Initial</td><td>SPOT$^-$ I</td><td>SPOT I</td><td colspan="2">SPOT$^+$I</td><td colspan="2">SINF III
SINF IV
SINF V</td><td colspan="2">End</td><td colspan="4"></td></tr>
<tr><td>III:FRIEND FIGHT</td><td>Initial</td><td>RINF$^-$ from II</td><td>RINF from II</td><td colspan="2">RINF$^+$ from II</td><td colspan="2">FIR$^-$at I</td><td colspan="2">FIR atI</td><td colspan="3">FIR$^+$at I</td><td>End</td></tr>
<tr><td>IV:FRIEND SHIP</td><td>Initial</td><td>RINF$^-$ from II</td><td>RINF from II</td><td>FIR$^-$ at I</td><td colspan="2">FIR at I</td><td>FIR$^+$ at I</td><td>BBOM$^-$ by I</td><td colspan="2">BBOM by I</td><td colspan="2">BBOM$^+$ by I</td><td>End</td></tr>
<tr><td>V:FRIEND SHIP</td><td>Initial</td><td>RINF$^-$ from II</td><td>RINF from II</td><td>FIR$^-$ at I</td><td colspan="2">FIR at I</td><td>FIR$^+$ at I</td><td>BBOM$^-$ by I</td><td colspan="2">BBOM by I</td><td colspan="2">BBOM$^+$ BY I</td><td>End</td></tr>
</table>

SYMBOLS: BFA = BEING FIRED AT, SPOT= SPOTTING, SINF = SENDING INFORMATION
RINF = RECEIVING INFORMATION, FIR = FIRING , BBOM = BEING BOMBED ,
BOM = BOMBING, ( )$^-$ = JUST BEFORE, ( )$^+$ = JUST AFTER

EACH ROW LISTS EVENTS IN SEQUENCE OF OCCURRENCE. COLUMNS ARE NOT RELATED TO SUCH OCCURRENCES.

over what time periods does it occur.

Consider, for example, the following possible simplified Inner-Outer Air Battle:

(1) Enemy bombers(I) arrive in formation towards grouping of friendly ships of two types (IV,V).
(2) Friendly scout airplanes (II) detect/surveil I and pass information to friendly fighter airplanes (III) as well as to IV,V.
(3) III meet I and attack, I being passive.
(4) Remaining I continue toward IV,V, with III now ceasing attacks.
(5) IV,V send missiles against remaining I, before themselves are bombed by I.
(6) Remaining I, following now above missile attack, bomb IV,V.
(7) End of scenario as I turns away.

For an outline of approach, the aggregation level here is to make each individual combatant a distinct node. Thus, there are five types of nodes here: enemy bombers (I), friendly surveillance airplanes (II), friendly fighter planes (III), friendly ships of type (IV) and (V). Within each type, one can designate particular individual nodes by suitable indexing. In addition, one must determine for each side, friendly or adversary, all of the relative primitive relations such as given in (4.21) in probability form. All of this will also depend on what constitutes node cycles. Based on this scenario, Table IV presents a tentative collection of epochs of node interaction which can be identified as node input-output processing cycles.

Future efforts will be directed toward further implementation of the C^3 decision game relative to particular scenarios such as the Inner-Outer Air Battle.

7. ACKNOWLEDGMENTS

This work was supported jointly by the Naval Ocean Systems Center (NOSC) Program for Independent Research (IR), Code 013, under Project ZT52, and by the Joint Directorate of Laboratories, Technical Panel for Command, Control, and Communications, Basic Research Group (JDL,TPC^3,BRG). Also acknowledgments should be extended to: Prof. A.H. Levis, LIDS, MIT, for valuable discussions concerning the scope of C^3 efforts; M.C. Mudurian, NOSC, Code 421, for his specific support and interactions; and to V.J. Monteleon, NOSC Code 421 and J.A. Salzmann, Jr., NOSC Code 42, for their overall support and encouragement.

8. REFERENCES

1. Athans, M., ed. *Proceedings of the First MIT/ESL-ONR Workshop on Distributed Communications and Decision Problems Motivated by Naval C^3 Systems, Vols. 1,2,3*, Cambridge, Mass.:LIDS, MIT, Sept., 1978.

2. Joint Directorate of Laboratories, Technical Panel for C^3, Basic Research Group (JDL,TPC^3, BRG), *1988 Proceedings of the Command and Control Research Symposium*, McLean, VA.: SAIC, 1988.

3. JDL,TPC^3, BRG, *C^3 Handbook (Draft)*, San Diego, CA.: NOSC and NPRDC, 1988.

4. Institute of Electrical Engineers (IEE), Computing, Control, and Electronics Division, *Proceedings of the First and Second International Conferences on Command, Control, Communications and Management Information Systems*, Great Britain, 1987, 1988.

5. Institute of Electrical and Electronics Engineers (IEEE), *Special Issue on Information Technology for Command and Control*, *IEEE Trans. Sys. Man, Cybern., Vol SMC-16(6)*, Nov.-Dec., 1986.

6. Armed Forces Communications and Electronics Association (AFCEA), *Science of Command and Control: Coping with Uncertainty*, Fairfax,VA.: AFCEA International Press, 1988.

7. Sweet, R., Metersky,M., and Sovereign,M, eds., *Command and Control Evaluation Workshop (Proceedings)*, Based on Military Operations Research Society (MORS) MOE Workshop, Jan, 1985, Naval Postgraduate School, MORS and MITRE Corp. Nov., 1985.

8. Strack, C., *Elements of C^2 Theory*, McLean, VA.: Defense Systems Inc. (DSI), draft version; Oct. 15, 1985, abridged version, Jan. 30, 1985.

9. Office of the Secretary of Defense and MITRE Corp., C^3 Div., *Proceedings for Quantitative Assessment of Utility of Command and Control Systems*, Proceedings of conference held at National Defense University (NDU), Ft.McNair, Wash.,D.C., Oct., 1979, McLean, VA.: MITRE Corp., Jan., 1980.

10. Goodman, I.R., "Combination of evidence in C^3 systems"., *Proc. 8th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge, Mass.:LIDS, MIT, Dec., 1985, pp. 161-166.

11. Van Trees, H.L., "C^3 systems research- a decade of progress", *Proc. 1988 Comm. Cntrl. Res. Symp.*, McLean, VA.: SAIC, 1988.

12. Ingber, L., "Nonlinear, nonequilibrium statistical mechanics approach to C^3 systems", *Proc. 9th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge, Mass: LIDS,MIT, Dec., 1986, pp. 237-244.

13. Rubin,I. and Mayk,I., "Markovian modeling of canonical C^3 systems components", *Proc. 8th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge, Mass: SAIC, Dec., 1985, pp. 15-23.

14. Levis, A.H., "Information processing and decisionmaking organizations: a mathematical description", *Proc. 6th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge. Mass.: Dec., 1983, pp. 30-38.

15. Tomovic, M.M. and Levis, A.H., "On the design of organizational structures for command and control", *Proc. 7th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge, Mass.: Dec., 1984, pp. 131-138.

16. Goodman, I.R., "A probabilistic-possibilistic approach to modeling C^3 systems", *Proc. 9th MIT/ONR Wrkshp. C^3 Sys.*, Cambridge, Mass: LIDS, MIT, Dec., 1986, pp. 53-58.

17. Goodman, I.R., "A general theory for the fusion of data". *Proc. 1987 Tri-Service Data Fusion Symp., Vol.1*, Warminster, PA.: NADC, Nov.,1987, pp. 254-270.

18. Goodman, I.R. and Nguyen, H.T., *Uncertainty Models for Knowledge-Based Systems*, Amsterdam: North-Holland, 1985.

19. Goodman, I.R., *PACT:An Approach to Combining Linguistic-Based and Probabilistic Information for Correlation and Tracking*, NOSC TD 878, San Diego, CA.: NOSC, March, 1986.

20. Metersky, M.L., "A C^2 process and an approach to design and evaluation", in [5], pp. 880-889.

21. Sweet, R., "The MCES and the search for generic measures", *Proc. 1987 C^2 Res. Symp.*, McLean,VA: SAIC, Nov., 1987, pp. 341-348.

22. Woodger, J.H., *The Technique of Theory Construction*, monograph, part of *International Encyclopedia of Unified Science, Vol.2(5)* Chicago: University of Chicago Press, 1956.

23. Carnap, R., *Introduction to Symbolic Logic and Its Applications*, New York: Dover Publications, 1958.

24. Goodman, I.R., "Applications of a conditional event algebra to data fusion", *Proc.1988 Tri-Service Data Fusion Symp.*, Warminster, PA;NOSC, to appear, 1988.

25. Goodman, I.R. and Nguyen, H.T., *A Theory of Measure-Free Conditioning*, monograph, to appear 1988-89.

26. Goodman, I.R. and Boyer, R.P., "Normal mixture approximations to prior densities in the linear regression model", *Proc. 5th Symp. Nonlinear Estimation Theory & Its Applic.*, North Hollywood, CA.: Western Periodicals, 1975, pp.74-77.

27. Feller, W., *An Introduction to Probability Theory and Its Applications, Vol.II*, New York: John Wiley & Sons, 1966.

28. Deutsch, R., *Estimation Theory*, Englewood Cliffs, New Jersey: Prentice-Hall, 1965.

29. Medgyessy, P., *Decomposition of Superpositions of Distribution Functions*, Budapest, Hungary: Publishing House of the Hungarian Academy of Sciences, 1961.

30. Goodman, I.R. and Nguyen, H.T., "Comparisons of uncertainty measures within a game theory context", to be submitted, 1988.

AN APPLICATION OF CATASTROPHE THEORY TO C^3I PROBLEMS

Patricia Langendorf [1,2,3]
Computer Scientist, Rome Air Development Center
Griffiss Air Force Base, New York 13441

Dr. Alexander E. R. Woodcock [1,2,3]
Chief Scientist, Synectics Corporation
10400 Eaton Place, Fairfax, Virginia 22030

ABSTRACT

Mathematical and statistical tools based on catastrophe theory have been developed to permit individuals with a minimal mathematical background to undertake a rigorous analysis of non-linear indications and warning (I&W)-related phenomena. These tools have been used in experiments with experienced intelligence analysts that involved the simulated detection of Soviet Operational Maneuver Groups (OMGs), one of the most difficult problems of tactical analysis. The analysts who served as subjects had no mathematical background, yet were enthusiastic about the technology. They were particularly interested in a perceived capability to analyze themselves and to identify and correct inconsistent and ambiguous responses.

1. NEW I&W AND C^2 MODELING TECHNOLOGIES ARE NEEDED

There is evidence that sudden perceptual changes often occur in the indications and warning (I&W) and command and control (C^2) worlds and that different analysts can have different, ambiguous, perceptions of the same military situation. Such behavior is obviously non-linear and must be investigated with the aid of non-linear techniques and models. Linear models tend to mask rather than highlight sudden changes and therefore may be highly misleading to their users when applied to the analysis of ambiguous situations.

The fitting of a series of data points to a straight line by linear regression, for example, will indeed provide a mathematical model of these data. However, such a process is not aimed at detecting, and therefore would not reveal, the existence of discontinuities where a small change in the value of an independent variable can produce large (step-function-like) changes in the value of an associated dependent variable, for example. Normally, such changes would be identified with an increase in the error terms associated with the regression procedures, and therefore would tend to be overlooked during the process of data analysis.

Linear combat models and normal or uni-modal statistics may have been adequate and/or appropriate in the past. However, the introduction of highly complex weapons systems and other advanced military technologies to the modern battlefield has led to multiple interactions and highly non-linear combat behavior. Such advances in complexity have stressed the existing fabric of understanding of combat to a point where it is clear that non-linear, multi-modal, modeling approaches are urgently needed to support those individuals involved in the modeling and analysis of such combat-related areas as indications and warning and command and control.

During a recent investigation of the impact of ambiguity on I&W analyst perceptions, a set of unclassified notional indicators predicting the development of a Soviet OMG was produced. Ten specific settings of these indicators were presented to intelligence analysts who were asked to assess the probability of OMG development. The resulting assessments were captured and analyzed. This analysis revealed the existence of perceptual ambiguity as well as the potential for sudden and gradual perceptual changes, perceptual hysteresis, and perceptual trapping associated with the analyst responses. This research has also provided evidence that I&W indicators are neither linear nor uncorrelated in the mind of experienced intelligence analysts, a finding which is itself of value. The technology is also directly applicable to communicating analyst understanding to battlefield commanders, and to capturing ambiguities in a battlefield commander's perception of the combat environment.

2. OPERATIONAL MANEUVER GROUP IDENTIFICATION: AN IMPORTANT AND UNSOLVED PROBLEM

The identification of an Operational Maneuver Group (OMG) is an important and unsolved problem which has been the subject of a recently completed investigation titled "I&W Applications of Catastrophe Theory (IWCAT)" for the

Rome Air Development Center (RADC). This paper reports some of the results of this investigation. An OMG is a highly mobile military unit derived from the Soviet Mobile Group (SMG), a tank formation used extensively during World War II (see Donnelly, 1982; Dick, 1983, for example). The OMG is designed to operate behind North Atlantic Treaty Organization (NATO) lines to facilitate a quick Soviet-Warsaw Pact victory by opening a second front and destroying the NATO defensive capacity. This goal is to be achieved by inserting an OMG into combat at weak points in the NATO defense lines to attack valuable targets, destroy or limit the nuclear capability of NATO, disrupt reinforcement supply lines, and maintain close proximity with NATO troops in order to prevent them from introducing tactical nuclear weapons.

An OMG achieves a high level of mobility and self-sufficiency by attaching self-propelled artillery, combat engineers, lift capacity, signal troops (to provide long range communication), and increased amounts of fuel and ammunition. OMGs also depend upon other military units for support. Such support would generally include air support, heavy artillery preparation, covering fire barrages, the use of jammers (to disrupt NATO air and fire support nets and command and control in the sector at which the break is expected to occur), and traffic control and lane clearing (to permit the rapid transit of the OMG to its intended target). The two major artillery units used to support an OMG are the Division Artillery Group (DAG) and the Regimental Artillery Group (RAG). Both groups are usually reinforced with non-divisional artillery battalions. Air defense for the OMG is provided by integrated systems of anti-aircraft artillery, surface-to-air missiles (SAMs), and fixed and rotary wing aircraft.

2.1 KNOWLEDGE DEVELOPMENT ACTIVITIES

Knowledge development activities aimed at obtaining an understanding of the activities involved in I&W threat perception were undertaken during the IWCAT project. A set of ten notional and unclassified indicators predicting the development of a Soviet OMG was developed for the investigation (Figure 1). Settings of these indicators and a group of context variables describing time of day, weather conditions, an assumed politico-military background (including conditions characterized as: "Treaty Obligation," "Friendly Ally," and "Third Party Hostilities"), as well as an assessment of the degree of indicator reliability (or "level of confidence"), were presented to intelligence analysts. These analysts were asked to describe their perception of the level of OMG threat as reflected in these data and record this information in an OMG threat assessment data base, as shown in Figure 2.

Analyst OMG threat assessments were captured and analyzed with the aid of statistical procedures based both on linear techniques and on catastrophe theory (Cobb, 1978, 1983). This analysis has demonstrated the existence of ambiguous perceptions and has provided an understanding of perceptual divergence and hysteresis and of a phenomena that Woodcock has called "perceptual trapping," (where an analyst would not withdraw an issued OMG warning despite a subsequent large reduction in the number of active indicators, for example) as described in more detail below.

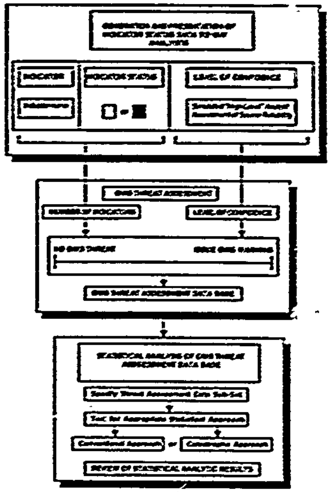


Figure 1. Catastrophe Theory-Based I&W Assessment Activities

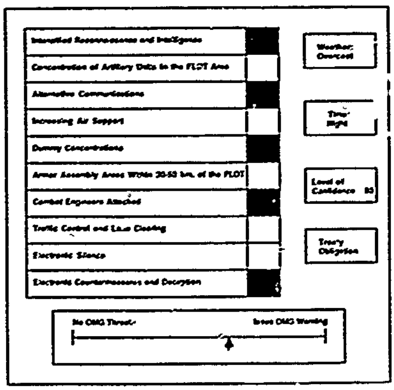


Figure 2. The IWCAT Intelligence Analyst Computer Display

3. CATASTROPHE THEORY CAN PROVIDE NEW TOOLS FOR THE I&W ANALYST

Catastrophe theory has been used to develop mathematically transparent tools for the study and analysis of I&W-related problems. The theory provides a framework for analyzing and modeling systems which possess some or all of the following characteristics:

1. Gradual and sudden changes, divergence, ambiguity, bimodality, and hysteresis that are characteristic of the behavior exhibited by the elementary catastrophes.

2. Situations where small changes in inputs can give rise to either small or large changes in outcome under the same conditions, and where small biases in input can give rise to dramatically different outcomes.

The elementary catastrophes provide the simplest models that are topologically equivalent of the behavior of systems with up to four key, independent, variables (called control or conflicting factors) and up to two key, dependendent, variables (called behavior variables) (Thom, 1975; Poston and Stewart, 1978; Woodcock and Davis, 1978; Woodcock and Poston, 1974; Gilmore, 1981; Isnard and Zeeman, 1976; and Zeeman, 1977). Catastrophe theory has been used in a number of different applications, including military applications (Dockery and Chiatti, 1986; Dockery and Woodcock, 1988; Holt, Job, and Marcus, 1978; Woodcock, 1988a, b, 1987a, b, 1986; Woodcock and Dockery, 1988, 1987, 1986, 1984; and Woodcock, Langendorf, and Cobb, 1988, for example).

3.1 CATASTROPHE LANDSCAPES CAN ILLUSTRATE NON-LINEAR SYSTEM BEHAVIOR

The cusp catastrophe has two control factors and one behavior variable and a catastrophe function ($V_c(x)$) of the form:

$$V_c(x) = x^4/4 + ax^2/2 + bx \qquad (1)$$

where a and b are the control factors and x is the behavior variable (Figure 3). Stationary states of this function occur when:

$$dV_c(x)/dx = x^3 + ax + b = 0 \qquad (2)$$

This equation describes a three-dimensional (x, a, b) curved surface (technically called the catastrophe manifold) that Woodcock and Dockery have called the catastrophe landscape. The catastrophe manifold surface has either one or three layers corresponding to the number of real solutions of equation (2) The single solution is a single minimum value of the function in equation (1) while the triple solution consists of two minima separated by a maximum value. The bifurcation set of the catastrophe represents the set of control factor values at which one or other of the minima of equation (1) is destroyed in response to control factor changes. It is described by equation (3):

$$4a^3 + 27b^2 = 0 \qquad (3)$$

Sudden or catastrophic changes of system behavior can occur at the bifurcation set in response to changes in the values of the control factors as one or other of the two minima of the catastrophe function (equation (1)) becomes a point of inflexion and then disappears. Non-catastrophic changes in behavior can also occur when one minimum of equation (1) is smoothly transformed into the other minimum by an appropriate set of control factor changes. Both types of changes in system behavior can be illustrated by the pattern of movement of a point called the state point on the surface of the catastrophe landscape in response to changes in control factor values (Figure 3). Thus, the path (p, q, r) represents a sudden or catastrophic change in system behavior while the path (s, t, u) represents control factor changes that cause gradual changes in system behavior.

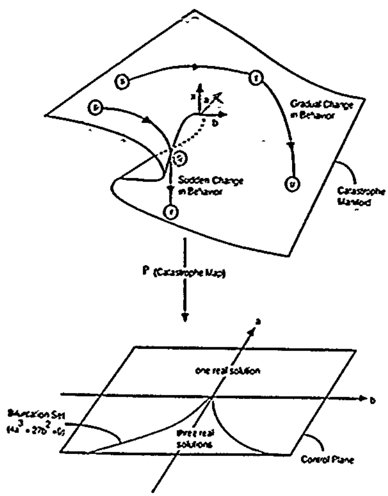


Figure 3: The Cusp Catastrophe Manifold and Control Plane

3.2 CUSP SURFACE ANALYSIS USES BIMODAL STATISTICS

The data produced by the analyst assessments of the OMG threat performed during the project was analyzed with the aid of cusp surface analysis procedures based on catastrophe theory that have been developed by Cobb (1978, 1980, 1983). These procedures permit the construction of statistical models involving one dependent variable and an arbitrary number of independent variables

3.2.1 THE STATISTICAL CUSP MODEL

In the cusp surface analysis program, the original control variables and the original behavior variable are transformed by a mathematical process that adjusts the coordinate system so that the shape of the original response surface matches the statistically-derived cusp catastrophe surface near the cusp point or origin of the pleat (Figure 2). This results in the behavior variable for the statistical model becoming a function of both the original behavior variable and the original control variables. This is the principal difference between the statistical and canonical (or elementary catastrophe) cusp models.

The statistical cusp catastrophe model is a transformation of the original system response surface which is itself generated from analyst-derived data. A series of statistical cusp control factors A(X), B(X), and C(X) that are functions of a vector of independent variables, X, are defined as a follows:

$$A(X) = A_0 + A_1X_1 + \dots + A_rX_r \quad (4)$$

$$B(X) = B_0 + B_1X_1 + \dots + B_rX_r \quad (5)$$

$$C(X) = C_0 + C_1X_1 + \dots + C_rX_r \quad (6)$$

These values are used to determine the value of the dependent or output variable, Y, where the predicted values of this variable are solutions of the equation:

$$A(X) + B(X)[Y - C(X)] - D[Y - C(X)]^3 = 0 \quad (7)$$

This equation can be written in the following form:

$$a + by - dy^3 = 0 \quad (8)$$

Equation (8) is similar to the cusp manifold equation (2) if the coefficient (d) in equation (8) is set equal to the value (-1) and the dependent variable (y) of the statistical, cusp surface, model is identified with the dependent variable (x) of the canonical model.

3.2.2 ESTIMATING PARAMETERS

The cusp surface analysis program uses the method of maximum likelihood to estimate the parameters of the cusp model. The process begins with the estimation of the coefficients of a linear regression model. These coefficients are then used as a starting point for an iterative process which employs a modified Newton-Raphson method to construct a statistical model. This process uses a set of input data elements and computes the values of a set of parameter values that maximize the likelihood of the cusp model based on these data. If the very first iteration performed by this process yields a decrease in the likelihood function, or no cubic term is detected in the data, the program indicates that a linear model is preferable to a non-linear (or cusp) model and the analysis is halted (Figure 1).

When cubic terms are detected in the data, the cusp analysis program continues to compute logarithmic likelihood values of this model until convergence is achieved. The analysis is halted if convergence is not possible, or if the value of the estimated likelihood value fails to increase. When successful convergence to parameter values that maximize the likelihood function is achieved, the cusp surface analysis program presents the estimated coefficients of each factor. The program also computes the most likely dependent variable values and presents these data in tabular form. This variable may have either one value (corresponding to a single mode of an associated conditional probability density function (PDF)) or three values (corresponding to two modes and a single anti-mode of the PDF). Graphical displays of "slices" of the estimated catastrophe surface model for sets of fixed and variable parameter values are also presented.

3.2.3 MAKING PREDICTIONS

There is no single definitive statistical test for the acceptability of a catastrophe model because these models generally offer more than one predicted value for the dependent (or behavior) variable given a set of values of independent (or control) variables. Under such circumstances, it is difficult to find a suitable definition for prediction error. Another difficulty arises because the statistical model is not linear in its parameters. In spite of these difficulties, several methods (involving the chi-square and other statistical tests) exist to validate a catastrophe model and some of these methods are used by the cusp surface analysis program (Figure 4). The cusp catastrophe model provides a suitable description of the relationship between a series of independent variables and a dependent variable when the following conditions are satisfied:

1. The chi-square test shows that the likelihood of the cusp model is significantly higher than that of the linear model.
2. The coefficient for the cubic term and at least one of the coefficients of the factors A and B are significantly different from zero.
3. At least 10% of the data points in the estimated model fall in the bimodal or ambiguous zone.

In applications of catastrophe theory, there are two distinct ways, called Maxwell and delay transition conventions, for calculating predicted values from a catastrophe model. The cusp surface analysis program provides estimates of which of these conventions is most appropriate for describing transitions of system behavior. The Maxwell convention considers the position of the highest mode of the probability density function to be the predicted value. The delay convention defines the predicted value as the equilibrium point towards which the equivalent dynamical system would have "moved."

4. OMG THREAT ASSESSMENT. MAPPING DATA TO THE CATASTROPHE SURFACE

Test data sets each with a different number of active indicators and level of confidence, time of day, weather conditions, and politico-military properties chosen to reflect indications of different adversarial status conditions were presented to experienced intelligence analysts. These

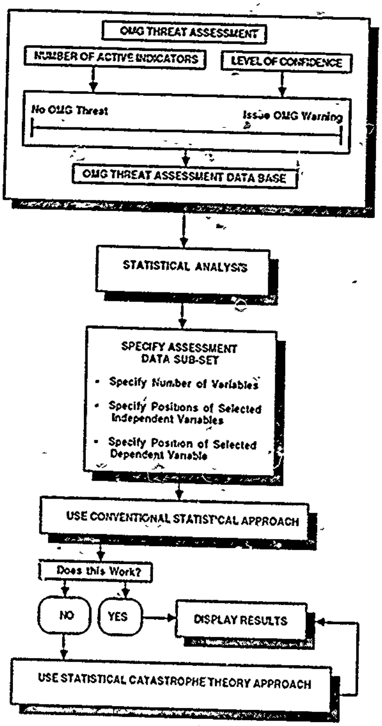


Figure 4: Cusp Surface Analysis

analysts were then asked to assess the level of OMG threat that they perceived to be reflected in these data (Figure 2).

During these threat assessment activities, the analyst formed a judgment as to his or her own relative degree of certainty that the display of data elements indicated that OMG formation would appear to be in progress. This information was then entered into the OMG threat assessment data base with the aid of the scale displayed on the computer screen. Following this task, the analyst was asked to designate which of the indicators were of primary importance and which were of secondary importance in determining the level of perceived OMG threat. Information generated during these activities was then analyzed with the cusp surface analysis program which attempted to construct a mathematical model of the analyst-derived assessment data, as described above.

These analysis and model-building activities were then followed by the production of a series of diagrams which display the major features of the statistical catastrophe model In one display (see Figure 5, for example), the transformed data are presented as locations on the control plane formed from transformed versions of the control factors (which are called the bifurcation (or splitting) and asymmetry (or normal) factors). In the sample case reported in the figure, some 41.5 percent of the data points are located within the v-shaped bimodal region. Under such circumstances, a linear model of these data would provide an incorrect data fit in over four out of ten cases.

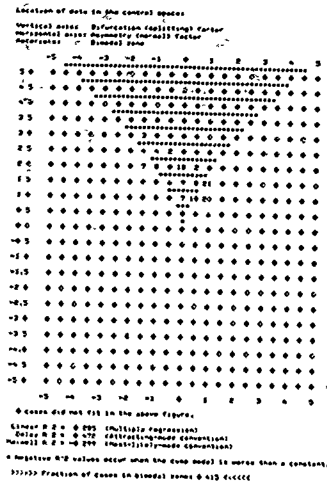


Figure 5 OMG Assessments Displayed on the Cusp Control Plane

The cusp surface analysis program constructs a cusp catastrophe model of the I&W analyst-derived data by a process that is illustrated in Figure 6 The transformed data is located within the circle drawn on the control plane formed from transformations of the number of active indicators and level of confidence control factors. Some of these data lie inside, and the remainder lie outside, the region of bimodality or ambiguity on the control plane.

4.1 CATASTROPHE MODELS

When a catastrophe model can be constructed from the analyst's threat assessment data, it is possible to describe a range of different I&W analyst response behaviors, such as sudden and gradual perceptual changes, divergence, ambiguity, hysteresis, perceptual "trapping," and counter-intuitive or paradoxical behavior. One particularly interesting discovery provides statistical evidence that suggests that I&W analysts with different types of training and previous mission responsibilities appear to respond to different features of the overall OMG indicator data set.

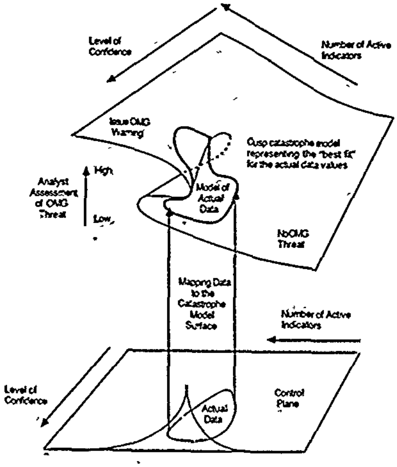


Figure 6: Mapping OMG Data to the Catastrophe Manifold

1. Sudden and Gradual Perceptual Changes The catastrophe model describes conditions under which sudden and gradual changes in analyst perceptions can occur in response to changes in the number of active indicators and the level of confidence attributed to these indicators. An increase in the number of active indicators with high confidence values can cause a sudden increase in the perceived level of OMG threat and may lead to the issuing of an OMG warning (Figure 7, path (a-b-c), for example). By contrast, a similar increase in the number of active indicators with a low level of confidence may not lead to the issuing of an OMG warning (Figure 7, path (d-e), for example).

2. Divergence Perceptual divergence occurs when relatively small differences in the initial number of active indicators presented to an I&W analyst can have a profound impact on their future perception of OMG threat. Path (a-b) (Figure 8) represents changing conditions, caused by an increase in the level of confidence, which would lead to the issuing of an OMG warning, while path (c-d) would lead to the perception that a very low level of OMG threat existed.

3. Ambiguity Perceptual ambiguities can be caused by "pre-conditioning" (Figure 9). The sequence of data sets represented by positions (a, b, c, and d), for example, could lead an analyst to the perception of a low level of OMG threat. By contrast, the sequence (e, f, g, and h), could result in the analyst issuing an OMG warning. Thus, while positions (d) ("no perceived OMG threat") and (h) ("issue an OMG warning") can represent exactly similar level of confidence and number of active indicator inputs, these inputs can produce ambiguous analyst perceptions of the level of OMG threat.

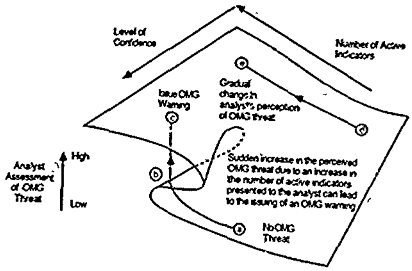


Figure 7: Sudden and Gradual Changes in Perception

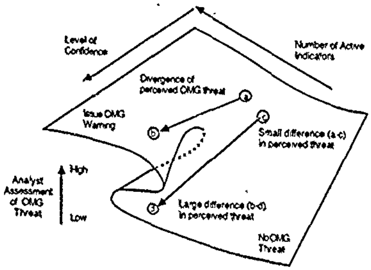


Figure 8 Divergent Perceptions

4. "Slicing" The Catastrophe Surface Slices of the catastrophe surface formed by maintaining a fixed level of confidence and varying the number of active indicators, for example, produces a continuous curve resembling an overfolded "S" (Figure 10). The shapes of such slices can illustrate such phenomena as perceptual hysteresis and perceptual "trapping," as described below.

5 Perceptual Hysteresis Perceptual hysteresis can be illustrated with the aid of the

catastrophe model (Figure 11). Beginning with a small number of active indicators and a low level of perceived OMG threat, an increase in the number of these indicators presented to the analyst can lead to a sudden increase in perceived threat. By contrast, beginning with a large number of active indicators and a high level of OMG threat, a decrease in this number of indicators can lead to conditions under which a sudden decrease in perceived OMG threat can take place. These sudden changes in perceived OMG threat generally take place at different numbers of active indicators, for example

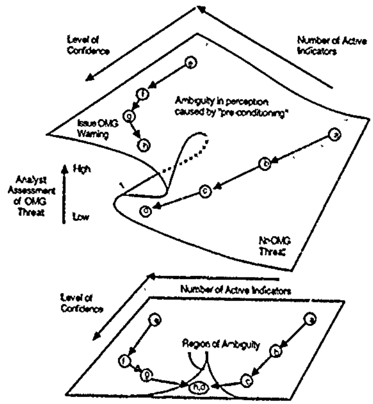


Figure 9: Perceptual Ambiguity

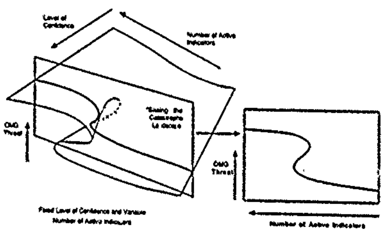


Figure 10: "Slicing" the Cusp Surface

6. Perceptual "Trapping" Perceptual trapping occurs because an individual's perceptions appear to be restricted to a particular portion of the catastrophe landscape so that no "return path" exists to permit a transition to another region of this surface (Figure 12). Under partial trapping conditions, for example, an increase in the number of active indicators presented to the analyst can set the scene for a sudden and irreversible increase in perceived OMG threat. This observation reflects the reluctance of some analysts to withdraw an OMG warning once it has been issued as revealed in comments made during the debriefing performed after the experiments had been completed. Complete perceptual trapping can occur when no path exists for transitions between the low OMG and high OMG threat perception states. Under such circumstances, the analyst's perception of OMG threat would remain at either a low or a high level despite very large changes in number of active indicators or level of confidence, for example.

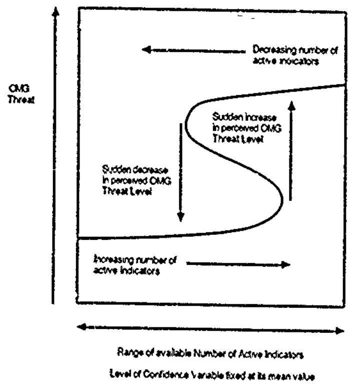


Figure 11: Perceptual Hysteresis

7. Counter-intuitive or Paradoxical Behavior Catastrophe models based on analysts' perceptions of OMG threat suggest that these perceptions may exhibit patterns of counter-intuitive or paradoxical behavior (Figure 13). Beginning with a high level of perceived OMG threat, with a small number of active indicators, and with the level of confidence maintained at its mean value, a subsequent increase in the number of active indicators presented to the analyst can lead to a sudden decrease in the level of perceived OMG threat, for example. Subsequent increases in the number of active indicators can now cause a gradual increase in the level of perceived threat, but the analyst's perceptions now appear to be "trapped" to a particular S-shaped slice of the catastrophe

surface. Under these circumstances, a return to the initial conditions would appear to be impossible without changes in the level of confidence or the presentation of additional sets of data elements to the analyst, for example.

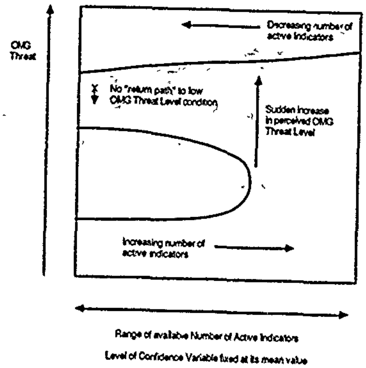


Figure 12: Perceptual Trapping

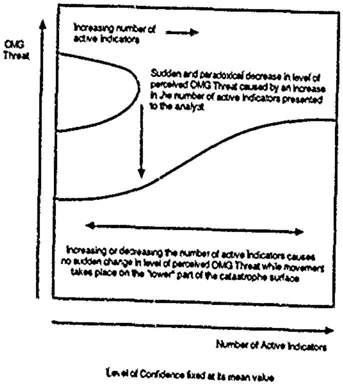


Figure 13: Counter-Intuitive or Paradoxical Behavior

4.2 SPECIFIC ANALYST ASSESSMENTS

Experienced intelligence analysts have tested the IWCAT system. It is a suggestive finding of the statistical analyses performed during this investigation that the nature of the response of the different analysts to the OMG threat test data appeared to depend upon the background and experience of these analysts. Thus analysts with extensive *active duty military* experience appeared to pay almost exclusive attention to the number of active indicators while another analyst with much more strategic-level experience appeared to pay almost exclusive attention to the patterns (or sequence type) of the displayed indicators, for example.

The following is a brief review of the analysis of the OMG threat data collected from an experienced intelligence analyst which describes the impact of the number of primary indicators, sequence type (or pattern of indicators), and level of confidence on the analyst's perception of OMG threat level (Figures 14a to d).

The control plane plot (Figure 14a) shows that 54.8% of the data are located within the bimodal zone and represent analyst assessment conditions which are subject to ambiguity. A slice of the cusp model surface for a range of values of the primary indicators and sequence type and level of confidence fixed at their mean values reveals situations in which partial perceptual trapping can occur (Figure 14b). A slice of the cusp surface at a fixed level of confidence and number of primary indicators and with variable sequence-type values reveals situations in which perceptual trapping can occur with the analyst's OMG threat perceptions restricted either to a high value range or a low value range for all ranges of sequence-type values (Figure 14c). A slice of the cusp model surface for a range of values of the level of confidence and with the numbers of primary indicators and sequence type fixed at their mean value reveal situations in which perceptual hysteresis can occur (Figure 14d) as the level of confidence is increased or decreased.

(a) Control Plane Plot

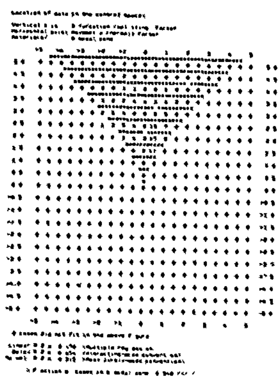
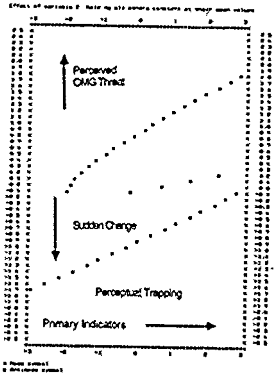


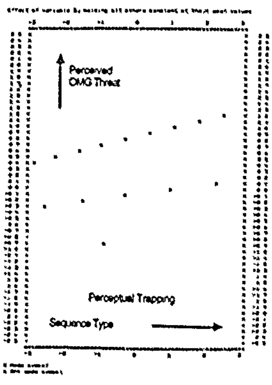
Figure 14a: Analysis of OMG Threat Assessment Data

These knowledge development activities involving the generation of OMG indicator test data sets, their presentation to I&W analysts, the construction of an OMG threat assessment data base, and its statistical analysis have



(b) Variable Primary Indicators, Fixed Sequence Type and Level of Confidence

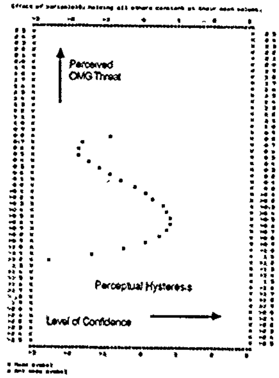
Figure 14b: Analysis of OMG Threat Assessment Data



(c) Variable Sequence Type, Fixed Primary Indicators and Level of Confidence

Figure 14c Analysis of OMG Threat Assessment Data

provided new insights to the perception of threat, particularly in the area of the OMG. It is now possible to identify conditions under which sudden and gradual changes, divergence, ambiguities, paradoxical reversals, hysteresis, and perceptual "trapping" can take place and provide an assessment of their impact on the interpretation of intelligence data by intelligence analysts and others These findings have significant implications in many combat-related areas, particularly in the communication of intelligence analyst perceptions to battlefield commanders, a subject that is under investigation.



(d) Variable Level of Confidence, Fixed Primary Indicators and Sequence Type

Figure 14d: Analysis of OMG Threat Assessment Data

5. CONCLUSIONS

The research described in this paper has produced a successful demonstration of the use of catastrophe theory mathematics to analyze indications and warnings-related data associated with a Soviet OMG. The effort has also produced a prototype computer-based system that can be used to support analysts and decision-makers. The system is able to capture and use small, and apparently insignificant, changes in information that are the precursors of dramatic changes in overall system behavior. The work has clearly demonstrated that I&W-related data can be generated and fitted to non-linear models in a rigorous extension of the ways that data are generated and fitted to linear models in the process of linear regression. The activity has gone far beyond simple qualitative model building by constructing quantitative, statistically-verified, models which can be used for analysis and prediction.

Several experienced intelligence analysts have tested the IWCAT system and the information generated by this process was subjected to non-linear statistical analysis using the cusp surface analysis program. It is a suggestive finding of this analysis that the nature of the response of the different analysts to the OMG threat test data appeared to depend upon their background and experience. While the observation that analyst background and experience lead to a specifity of analyst-response is a tentative finding due to the small sample size of analysts that were used in the experiment, such a suggestion can have profound implications on the way that I&W and other forms of intelligence analyses are interpreted and used. These possibilities should be the subject of further analytic activities and investigations with the aid of the IWCAT system.

This research has provided evidence that I&W indicators are neither linear nor uncorrelated in the mind of experienced intelligence analysts, a finding which is itself of value. The technology developed to investigate I&W analyst responses is directly applicable to communicating analyst understanding to battlefield commanders, and to capturing ambiguities in battlefield commanders' perception of the combat environment. The technology also appears to be of value as an aid to analysts and decision-makers by:

1. Alerting individuals to conditions where small changes in indicator input can give rise to either gradual or sudden changes of perception in the same situation under different conditions.

2. Providing an analytic capability that can give rational interpretations of non-linear and apparently counter-intuitive behavior and clarifying the causes and effects of ambiguous peceptions of the same situation.

3. Identifying and characterizing the different types of responses of I&W analysts and others to features of I&W-related data sets.

4. Providing methods that can be used to support the training of such analysts and the interpretation of their assessments of particular sets of indicators.

The catastrophe theory-based technology provides a method for developing a new understanding of the non-linear aspects of the military intelligence-analytic process. Such an understanding can then be extended from intelligence analysis to command and control by providing new methods for identifying and selecting particular responses from a list of response options available to the commander. The development of this type of facility is of obvious importance in the C^2 arena, and is being investigated

6. BIBLIOGRAPHY

Cobb, L., 1978. "Stochastic Catastrophe Models and Multimodal Distributions." *Behavioral Science*, 23: 360-374.

Cobb, L., 1980. "Estimation Theory for the Cusp Catastrophe Theory Model." *Proceedings of the Section on Survey Research*. Washington, D.C.: American Statistical Association,

Cobb, L., 1983. *A Maximum Likelihood Computer Program to Fit a Statistical Cusp Hypothesis* The Hague, The Netherlands: SHAPE Technical Centre.

Dick, C. J., 1983. "Soviet Operational Maneuver Groups. A Closer Look." *International Defense Review*, 769-772.

Dockery, J. T and S. Chiatti, 1986 "Application of Catastrophe Theory to the Problems of Military Analysis." *European Journal of Operations Research*, 24: 46-53.

Dockery, J T., and A. E. R. Woodcock, 1988. "Models of Combat with Embedded C2. II. Catastrophe Theory and Chaotic Behavior." *International C. I. S. Journal*. In Press.

Donnely, C. N., 1982. "The Soviet Operational Maneuver Group A New Challenge for NATO." *International Defense Review*, 1177-1186.

Gilmore, R., 1981. *Catastrophe Theory for Scientists and Engineers*. New York: Wiley Interscience.

Holt, R. T., B. Job, and L. Marcus, 1978. "Catastrophe Theory and the Study of War." *Journal of Conflict Resolution*. 22: 171-208.

Isnard, C. A. and E. C. Zeeman, 1976. "Some Models from Catastrophe Theory in the Social Sciences" In. *The Use of Models in the Social Sciences*. London. Tavistock Publications, Collins, L. (ed.) 44-100.

Poston, T. and I. Stewart, 1978. *Catastrophe Theory and its Applications*. London: Pitman.

Thom, R., 1975. *Structural Stability and Morphogenesis* Reading, Massachusetts: W. A. Benjamin.

Woodcock, A. E. R., 1988a. "Catastrophes, Chaos, and C^3I: Toward a New Understanding of Military Combat." In: *The Application of AI to C^3I*. Harris, C. J. (ed.). London: Institution of Electrical Engineers. In Press.

Woodcock, A. E. R., 1988b. "Indications and Warnings as an Input to the C^3 Process." In. *The Calculus of Uncertainty*. Johnson, S. E and A. H. Levis, (eds.). Washington D.C.: AFCEA International Press.

Woodcock, A. E. R., 1987a. "Catastrophes, Chaos, and C^3I." In: *Proceedings of the Second International Conference on C^3 Management Information Systems*. London: Institution of Electrical Engineers.

Woodcock, A. E. R., 1987b. "I&W as an Input to the C3 Process." In. *Proceedings of The Command and Control Research Symposium*. Washington, D. C. Joint Directors of Laboratories Basic Research Group, National Defense University, 83-92.

Woodcock, A. E. R., 1986. *An Investigation of Catastrophe Theory as a Command and Control Device* Rome, New York: Synectics Corporation.

Woodcock, A. E. R. and M. Davis, 1978. *Catastrophe Theory*. New York: E. P. Dutton.

Woodcock, A. E. R. and J. T. Dockery, 1988. "Models of Combat with Embedded C2. I. Catastrophe Theory and the Lanchester Equations." *International C I. S Journal*. In Press.

Woodcock, A E R. and J. T. Dockery, 1987 "Mathematical Catastrophes and Military Victories." *Signal*, April.

Woodcock, A. E. R. and J. T. Dockery, 1986. "Models of Combat with Embedded C2. I: Catastrophe Theory and the Lanchester Equations." Technical Report. Rome, New York: Synectics Corporation.

Woodcock, A. E. R. and J. T. Dockery, 1984. *Application of Catastrophe Theory to the Analysis of Military Behavior*. The Hague, The Netherlands: SHAPE Technical Centre Consultants Report, STC CR-56.

Woodcock, A E. R., P. Langendorf, and L. Cobb, 1988. "Perceptual Hysteresis. You Can't Get There From Here." *Signal*. July.

Woodcock, A. E. R. and T. Poston, 1974. "A Geometrical Study of the Elementary Catastrophes." *Lecture Notes in Mathematics*, 373. Berlin and New York. Springer-Verlag.

Zeeman, E C, 1977. *Catastrophe Theory, Selected Papers, 1972-1977*. Reading, Mass.: Addison-Wesley.

---

1. This research was supported by Rome Air Development Center Laboratory Director's funds. The authors are very grateful for the interest in and support of this work by Dr. F. Diamond, Chief Scientist, Rome Air Development Center, Griffiss Air Force Base, New York.

2. The authors would like to acknowledge with gratitude the participation of Dr. Loren Cobb (Medical University of South Carolina) and Dr. M. E. Familant (Synectics) in the IWCAT project.

3. The views, opinions, and findings contained in this paper are those of the authors and should not be construed as an official United States Air Force or Rome Air Development Center position, policy, or decision, unless so designated by other official documentation.

Stochastic Modeling and Analysis for Multi-Battle and Multi-Stage C^3 Systems*

Izhak Rubin*, Joseph Baker* and Israel Mayk#
*IRI Corporation, Tarzana, CA and UCLA, Los Angeles, CA
#USARMY CECOM, Center for C^3 Systems, Fort Monmouth, NJ

Abstract

The modeling of C^3 systems suffers from many of the same problems encountered in modeling other large, complicated systems. Because C^3 systems have many attributes, and complicated interactions, the state spaces of detailed models are so large that standard solution methods are computationally intractable. To combat this problem, we have developed high-level, aggregated C^3 models with reduced state spaces and an hierarchical structure. These models are easily programmed and run, and can be easily modified. Parameters of the high-level models are estimated using detailed simulation runs and other data.

For definiteness, we consider a specific battle scenario. However, we use a modular approach, and the model can easily be extended to other scenarios.

Introduction

In [1], we have presented a stochastic C^3 model for a multi-stage and multi-phase C^3 battle system. In our model, the battle is described as a serial succession of battle stages. During each stage, the forces engage in a certain type of battle. Following the termination of a stage, a new battle stage is initiated, so that the currently available forces now engage in a different battle type. The state of the system at the start of a battle stage depends upon the state at the termination of the previous battle stage.

Each stage is further divided into a serial succession of phases. During a phase, a battle takes place in accordance with the underlying system state conditions. The state of the system at the termination of a battle phase provides a key input for the determination of the related state of the system at the start of the next phase, within the same stage. The stage phases involve similar battle types engaged under varying system and force state conditions.

Performance equations are derived to describe the state distribution at the termination of each battle phase and stage, as well as to compute mission win probabilities, asset losses and battle durations.

Our multi-stage C^3 models have been applied to various Army, Air Force, and Navy scenarios. For example, in [1] an Army-based model is presented; in addition, extensive applications, tests and simulations have been carried out at NOSC in providing key modeling and performance evaluation and design tools for Outer-Air-Battle (OAB) and Inner-Air-Battle (IAB) air-defense scenarios, as employed by the U.S. naval forces in protecting a naval battle group.

To model the hierarchical integration of multiple levels of command structures, we need to hierarchically combine basic single-battle single and multi-phase models (see [2, 3, 4, 5, 6, 7, 1, 8]). It is essential to model the special hierarchical command, control and decision structure of the C^3 system, so that qualitative, as well as quantitative system understanding, modeling, evaluation, analysis and optimization functions can be accomplished.

Our Multi-Stage, Multi-Battle models describe hierarchically structured command organizations which are involved in managing and controlling the conduct of multiple battles. Each battle consists of multiple stages and phases, as described above. Once allocations of resources and assignments of tasks are made by the decision maker (command) at a certain level of the hierarchy, the affected lower level battle systems proceed to operate independently until the next decision point. At this latter time, information is aggregated and dis-aggregated as it moves up and down through the C^3 hierarchy.

Overview of the Multi-Stage, Multi-Battle MSMB-1 Model

For definiteness, we consider a specific battle scenario. However, we use a modular approach, and the model can easily be extended to other scenarios.

We consider a battle between two forces: the F-force and the A-force. Each force has, for example, tanks and aircraft. The tanks fight each other on a battlefield. Aircraft can be allocated to attack either tanks or aircraft from the opposing side. Thus, two battles are observed the tank battle, that may involve aircraft attacks as well, and the air-to-air aircraft combat battle.

Each force has a two-level hierarchical command structure. At the top level of the hierarchy, the commander for each side allocates aircraft either to the air-to-air battle, or to the tank attack mission. This decision is based upon summary statistics describing the state of the battles.

---

*This research was supported at IRI Corp. by U.S. Army CECOM contract no DAAB07-84-C-K577 Mailing Address Professor Izhak Rubin, Electrical Engineering Department, 6731 Boelter Hall, University of California, Los Angeles, CA 90024 Telephone. (213)825-2327

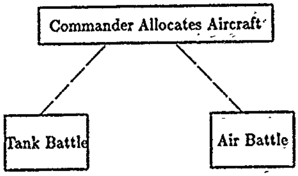


Figure 1: Command Hierarchy

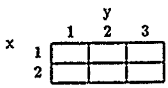


Figure 2: Tank Battle Grid

For instance, we assume that the commander knows the number of surviving tanks under his command, and their average fuel level, but does not know their exact position. The second level of the hierarchy consists of the individual commonads of the air-to-air battles on one hand, and of the tank (and air attack) battles, on the other hand.

The battle proceeds in stages, which correspond to decision points for the commanders. At the start of a stage, each commander decides how to allocate the aircraft under his control: some are allocated to the air battle, and the rest are allocated to the tank battle. During a stage, the battles proceed independently. Both of the battles are modeled as phased battles. A stage consists of a fixed number of phases (these numbers can be different for the tank battle and air battle.) Each phase of the tank battle consists of three subphases: tank movement, tank vs. tank attrition, and aircraft vs. tank attrition. The tank groups move on a 2×3 grid (see figure 2.) All members of a tank group occupy the same grid square. Grid locations affect tank vs. tank detection and kill probabilities, and aircraft vs. tank detection and kill probabilities. A tank group has either high or low fuel status. When a tank group with high fuel status moves, with some probability it changes to low fuel status. A tank group with low fuel status has limited movement options. (Rather than modeling a fuel limited battle, we could easily model an ammunition limited battle in the same manner. As the model undergoes further development, we will explore this option.)

Each phase of the air battle consists of a pairing of adversary aircraft, followed by the firing of a number of air-to-air missiles. The number of missiles fired per subphase is an input parameter. Unpaired aircraft wait for subsequent phases. Alternative air battle models could easily be used in place of this model.

At the end of each stage of the battle, the commanders reallocate aircraft based upon the outcomes of the air and tank battles. The battle can proceed for an arbitrary number of stages, and various win events can be defined. Statistics showing the transient and overall progress of the battle are collected.

We have derived analytic performance equations that can be recursively solved through the use of a computer, to yield the distributions of the underlying battle states (see [1] for details.) These results are then used to compute win probabilities and loss distributions, for prescribed battle durations, under various predetermined or adaptively and dynamically adjusted rules of engagement, resource allocation policies, and other command and control options. We present performance curves for C^3 battle examples under a variety of command and control policies for various parameters and scenario conditions.

Input Variables

The inputs to the model are:

- N_S = Number of stages in battle.
- N_{TS} = Number of tank battle phases per stage.
- N_{AS} = Number of air battle phases per stage
- T_1^F = Initial number of F-Tanks.
- X_1^F = Initial X coordinate of F-Tank group.
- Y_1^F = Initial Y coordinate of F-Tank group.
- T_1^A = Initial number of A-Tanks.
- X_1^A = Initial X coordinate of A-Tank group.
- Y_1^A = Initial Y coordinate of A-Tank group.
- A_1^F = Initial number of F-Aircraft.
- A_1^A = Initial number of A-Aircraft.
- N_{MT}^F = Number of missiles fired by an F-Tank which has detected a target, during a tank vs. tank subphase of the tank battle.
- N_{MT}^A = Number of missiles fired by an A-Tank which has detected a target, during a tank vs. tank subphase of the tank battle.
- N_{MAT}^F = Number of missiles fired by an F-Aircraft against a detected A-Tank, during an aircraft vs. tank subphase of the tank battle.
- N_{MAT}^A = Number of missiles fired by an A-Aircraft against a detected F-Tank, during an aircraft vs. tank subphase of the tank battle.
- N_{MAA}^F = Number of missiles fired by an F-Aircraft against a targeted A-Aircraft, during a phase of the aircraft vs. aircraft battle.
- N_{MAA}^A = Number of missiles fired by an A-Aircraft against a targeted F-Aircraft, during a phase of the aircraft vs. aircraft battle.

- $P^F_{DTT}(X_1,Y_1,X_2,Y_2)$ = Probability that an F-Tank at (X_1,Y_1) will detect an A-Tank at (X_2,Y_2) during a tank vs. tank subphase.
- $P^A_{DTT}(X_1,Y_1,X_2,Y_2)$ = Probability that an A-Tank at (X_1,Y_1) will detect an F-Tank at (X_2,Y_2) during a tank vs. tank subphase.
- $P^F_{DAT}(X,Y)$ = Probability that an F-Aircraft will detect an A-Tank at (X,Y) during an aircraft vs. tank subphase of the tank battle.
- $P^A_{DAT}(X,Y)$ = Probability that an A-Aircraft will detect an F-Tank at (X,Y) during an aircraft vs. tank subphase of the tank battle.
- $P^F_{KTT}(X_1,Y_1,X_2,Y_2)$ = Probability that a single missile from an F-Tank will kill a detected A-Tank.
- $P^A_{KTT}(X_1,Y_1,X_2,Y_2)$ = Probability that a single missile from an A-Tank will kill a detected F-Tank.
- $P^F_{KAT}(X,Y)$ = Probability that a single missile from an F-Aircraft will kill a detected A-Tank at (X,Y).
- $P^A_{KAT}(X,Y)$ = Probability that a single missile from an A-Aircraft will kill a detected F-Tank at (X,Y).
- P^F_{KAA} = Probability that a single missile from an F-Aircraft will kill a targeted A-Aircraft.
- P^A_{KAA} = Probability that a single missile from an A-Aircraft will kill a targeted F-Aircraft.
- $P^F_{HL}(X_1,Y_1,X_2,Y_2)$ = Probability that an F-Tank group in the high fuel state will switch to the low fuel state if it moves from (X_1,Y_1) to (X_2,Y_2).
- $P^A_{HL}(X_1,Y_1,X_2,Y_2)$ = Probability that an A-Tank group in the high fuel state will switch to the low fuel state if it moves from (X_1,Y_1) to (X_2,Y_2).

Definition of the Model

We will define the model in sections corresponding to the top level of the hierarchy, the tank battle, and the air-to-air battle. We will further separate the definitions of the movement, tank vs. tank attrition, and aircraft vs. tank attrition subphases of the tank battle. This approach makes it very easy to modify assumptions about firing doctrine, allocation rules, and so forth. By using this approach, we can consider multiple tank battles and/or air battles by changing only the top level of the model.

The Top Level Model

The top level of the battle proceeds as follows:

- The battle begins with an initial allocation of tanks and aircraft.
- At the start of each stage, the commanders each consider summary results of the previous stage of the battle, and allocate their aircraft between the tank and air battles.
- The process repeats itself.

The State

We define the state of the top level battle at the start of stage n to be:

$$(T^F_n, I^F_n, A^F_n, T^A_n, I^A_n, A^A_n)$$

where

- T^F_n = The Number of F-Tanks at the Start of Stage n.
- I^F_n = The Fuel State of the F-Tanks at the Start of Stage n.
- A^F_n = The Number of F-Aircraft at the Start of Stage n.
- T^A_n = The Number of A-Tanks at the Start of Stage n.
- I^A_n = The Fuel State of the A-Tanks at the Start of Stage n.
- A^A_n = The Number of A-Aircraft at the Start of Stage n.

Aircraft Allocation Rules

We assume that both commanders allocate aircraft based upon a deterministic function of the state. That is, we let

- $A^F_{Tn} = F^F_n(T^F_n, I^F_n, A^F_n, T^A_n, I^A_n, A^A_n)$ = The Number of F-Aircraft allocated to the Stage n Tank Battle, $0 \le A^F_{Tn} \le A^F_n$.
- $A^F_{An} = A^F_n - A^F_{Tn}$ = The Number of F-Aircraft allocated to the Stage n Air Battle.
- $A^A_{Tn} = F^A_n(T^F_n, I^F_n, A^F_n, T^A_n, I^A_n, A^A_n)$ = The Number of A-Aircraft allocated to the Stage n Tank Battle, $0 \le A^A_{Tn} \le A^A_n$.
- $A^A_{An} = A^A_n - A^A_{Tn}$ = The Number of A-Aircraft allocated to the Stage n Air Battle.

The allocation rules $F^F_n(\cdot)$ and $F^A_n(\,)$ do not have to use all of the state information; we can assume that each side has only partial information about the other side. The allocation rules are easily changed. We will use the following allocation rules in the numerical examples presented later:

1. If your side has more ($\geq$) aircraft than the other side, allocate half of your aircraft (rounded down to the nearest integer number) to the tank battle and the rest to the air battle; otherwise allocate all of your aircraft to the air battle;
2. Always allocate one half (rounded down to the nearest integer) of your aircraft to the tank battle, and the rest to the air battle;
3. Always allocate all of your aircraft to the tank battle;
4. Always allocate all of your aircraft to the air battle.

Update of the State Probability Distribution at the Start of Each Stage

The state probability distribution at the start of each stage is calculated in the following manner:

- Assume that the state probability distribution for the start of stage n has been calculated.
- For each state with positive probability, the allocation rules are used to determine the allocation of aircraft to the tank and air battles.
- The conditional distributions of the outcomes of the tank battle and the air battle are each calculated (independently, using the tank and air battle models), given the initial state.
- The conditional distributions are combined to give the overall joint conditional distribution for the start of stage $n+1$, given the state at the start of stage n.
- The conditional distributions are combined to yield the overall joint state distribution for the start of stage $n+1$.

Note that this is not the most computationally efficient approach. Since many of the joint states will give the same inputs to either the tank battle or the air battle, either sub-battle could be recalculated for the same set of inputs many times. It would be more computationally efficient to store the results of the sub-battles for each set of input conditions, and then combine them appropriately. However, this would require a great deal of storage. For our initial model, we calculate and store the results of various sub-battles and phases only once, and then combine them as needed. In future work we will examine of the computation/storage tradeoffs.

The Tank Battle

The tank battle is modeled as follows: Two opposing groups of tanks battle each other on a 2-dimensional surface. The surface is divided into 6 sections (for our initial model, a 2×3 grid.) All the tanks in a tank group are in the same grid square. The tank groups move about the grid, fighting each other. Opposing aircraft also strafe the battlefield. The tank vs. tank and aircraft vs tank probabilities of detection and kill depend upon the tank locations. When a tank group moves, with some probability the tank group will enter the "low fuel state." Once in the low fuel state, tank movement is restricted.

Each stage of the tank battle is divided into phases, and the phases are further divided into sub-phases of movement, tank vs. tank fighting, and aircraft vs. tank fighting.

A stage of the tank battle proceeds as follows:

- A set of initial conditions is passed in from the upper level of the hierarchy. This consists of the initial number of tanks and aircraft on each side, and the initial tank fuel status for each side.
- The state is expanded to include position information. For the first stage, this is done using the initial tank positions which are input by the user. For later stages, a conditional distribution of position, given the number of surviving tanks and their fuel status is calculated at the end of each stage, and is then used at the beginning of the next stage to expand the state (see below for more details.) Note that this is not exact. To be exact, we would have to use a conditional distribution that also included the number of aircraft allocated to the tank battle by each side, and track the number of aircraft allocated in the previous stage, so that we could properly expand the state.
- The effect of each phase of the tank battle upon the expanded state distribution is calculated.
- At the end of the stage, the distribution of the summary state is calculated and passed to the upper level of the hierarchy. The conditional distribution of the position information is also calculated.

Expanding The State

At the start of stage n, the tank model receives initial values for the number of tanks and aircraft on each side, and the fuel status of the tank groups. We use the subscript ni to indicate values at the end of phase i of stage n. We use the subscript $n0$ to indicate the initial values. Thus, in keeping with our other notation, we have

- T^F_{ni} = The Number of F-Tanks at the end of phase i of Stage n.
- I^F_{ni} = The Fuel State of the F-Tanks at the end of phase i of Stage n.
- A^F_{ni} = The Number of F-Aircraft at the end of phase i of Stage n.
- T^A_{ni} = The Number of A-Tanks at the end of phase i of Stage n.
- I^A_{ni} = The Fuel State of the A-Tanks at the end of phase i of Stage n.
- A^A_{ni} = The Number of A-Aircraft at the end of phase i of Stage n.

The tank battle model receives initial values for T^F_{n0}, I^F_{n0}, A^F_{n0}, T^A_{n0}, I^A_{n0}, and A^A_{n0}. Note that since we assume that there is no aircraft attrition in the tank battle, $A^F_{ni} = A^F_{n0}$ and $A^A_{ni} = A^A_{n0}$ for all i. Since the number of aircraft allocated to the tank battle do not change during a stage, we do not explicitly carry the numbers in the state description.

We define the state of the tank battle at the start of phase i of stage n to be

$$(T^F_{ni}, I^F_{ni}, X^F_{ni}, Y^F_{ni}, T^A_{ni}, I^A_{ni}, X^A_{ni}, Y^A_{ni})$$

where

- X^F_{ni} = the x-coordinate of the F-tank group at the start of phase i of stage n.
- Y^F_{ni} = the y-coordinate of the F-tank group at the start of phase i of stage n.

- X^A_{ni} = the x-coordinate of the A-tank group at the start of phase i of stage n.
- Y^A_{ni} = the y-coordinate of the A-tank group at the start of phase i of stage n.

At the start of the first stage, tank group position information is derived from the input values.

At the end of each stage of the tank battle, the conditional distribution of the tank group positions, given the number of surviving tanks on each side, and the fuel status of each tank group is calculated.

At the start of each stage other than the first, the expanded state distribution is calculated using the initial conditions and the conditional distribution of the tank group positions.

Once the expanded state distribution has been calculated, the subphase calculations are performed for each of the N_{TS} phases of the tank battle. We define the state of the tank battle at the start of subphase j of phase i of stage n to be

$$(T^F_{nij}, I^F_{nij}, X^F_{nij}, Y^F_{nij}, T^A_{nij}, I^A_{nij}, X^A_{nij}, Y^A_{nij}).$$

We define the state probability distribution at the start of subphase j of phase i of stage n to be

$$P^T_{nij}(j^F, k^F, x^F, y^F, j^A, k^A, x^A, y^A).$$

Finally, the ending summary state distribution is calculated and passed to the higher level of the command hierarchy, and the conditional position distribution is calculated and stored for the next stage.

The Movement Subphase

The tank groups move according to deterministic functions which are specified by the user. These functions can depend upon the number of surviving tanks on each side, and their fuel status' and positions. Of course, the functions can ignore some of this information to model partial knowledge. We assume that a tank group can move at most one square to the right or left, up or down during a single movement subphase. When a tank group in the high fuel state moves, with some probability it changes to the low fuel state. This probability is a function of the tank group location, and the movement it makes. The probability is input by the user.

The Tank vs. Tank Subphase

In the tank vs. tank subphase, the two tank groups attempt to detect and fire upon each other. Various models of the detection, target allocation, and firing processes are possible.

Detection. Various assumptions about tank detection and communication abilities can be made. We assume here that each of the tanks in a tank group independently scans the battlefield. The probability that a tank detects a particular opposing tank depends only upon the two tanks' grid locations, and is independent of any other factors. We assume that members of a tank group can communicate and share targeting information. Thus, the crucial factor is whether any member of a tank group has detected a particular opposing tank: detection by multiple tanks adds nothing.

We recall that

- $P^F_{DTT}(X^F, Y^F, X^A, Y^A)$ = Probability that an F-Tank at (X^F, Y^F) will detect an A-Tank at (X^A, Y^A) during a Tank vs. Tank Subphase.
- $P^A_{DTT}(X^A, Y^A, X^F, Y^F)$ = Probability that an A-Tank at (X^A, Y^A) will detect an F-Tank at (X^F, Y^F) during a Tank vs. Tank Subphase.

Thus, the probability that a particular A-Tank at (X^A, Y^A) is detected by an F-Tank group at (X^F, Y^F) with j tanks is given by

$$P^F_{DC}(X^F, Y^F, X^A, Y^A|j) = 1-(1-P^F_{DTT}(X^F, Y^F, X^A, Y^A))^j$$

The other probabilities can also be easily calculated.

Firing Models. We assume that during a single tank vs. tank subphase, each F-tank can fire N^F_{MT} missiles, and each A-tank can fire N^A_{MT} missiles. A number of firing models are possible. For our sample firing model, we ignore the effects of attrition within a subphase upon the total number of missiles fired during the subphase. We also assume that any detected ta get can be fired upon by any tank in the opposing group, and ignore any retargeting delays. We also assume that all missiles are fired at live targets. This provides a simple first order model. Various assumptions can be relaxed at the cost of added complexity, or by adding simple degrade factors.

The Aircraft vs. Tank Subphase

In the aircraft vs. tank subphase of the tank battle, we assume that the aircraft allocated by each side to the tank battle pass over the battlefield, search for and firing upon the opposing tanks. For the MSMB-1 model, we assume that there is no attrition of aircraft. As in the tank vs. tank subphase, numerous detection and firing models are possible. We present a representative model here.

Detection and Firing Model. We assume that the detection and firing phases are separate, and very similar to firing model 1 in the tank versus tank subphase. We assume here that each of the aircraft in a aircraft group independently scan the battlefield. The probability that an aircraft detects a particular opposing tank depends only upon the tank's grid location, and is independent of any other factors. We assume that members of an aircraft group can communicate and share targeting information. Thus, the crucial factor is whether any member of an aircraft group has detected a particular opposing tank. detection by multiple aircraft adds nothing. We recall that

- $P^F_{DAT}(X^A, Y^A)$ = Probability that an F-Aircraft will detect an A-Tank at (X^A, Y^A) during an Aircraft vs. Tank Subphase.

- $P^A_{DAT}(X^F,Y^F)$ = Probability that an A-Aircraft will detect an F-Tank at (X^F,Y^F) during an Aircraft vs. Tank Subphase.

Thus, the probability that a particular A-Tank at (X^A,Y^A) is detected by an F-Aircraft group with h^F aircraft is given by

$$P^F_{DA}(X^A,Y^A|j) = 1-(1-P^F_{DAT}(X^A,Y^A))^{h^F}$$

The probability that an F-Aircraft group with h^F aircraft will detect l out of the m members of an A-Tank group at (X^A,Y^A) is, for $0 \le l \le m$, $P^F_{DAC}(l|h^F,X^A,Y^A,m) =$

$$\binom{m}{l} P^F_{DA}(X^A,Y^A|h^F)^l(1-P^F_{DA}(X^A,Y^A|h^F))^{m-l}$$

Similarly, the probability that a particular F-Tank at (X^F,Y^F) is detected by an A-Aircraft group with h^A aircraft is given by

$$P^A_{DA}(X^F,Y^F|h^A) = 1-(1-P^A_{DAT}(X^F,Y^F))^{h^A}$$

The probability that an A-Aircraft group with h^A aircraft will detect l out of the m members of an F-Tank group at (X^F,Y^F) is, for $0 \le l \le m$,

$$P^A_{DAC}(l|h^A,X^F,Y^F,m) =$$

$$\binom{m}{l} P^A_{DA}(X^F,Y^F|h^A)^l(1-P^A_{DA}(X^F,Y^F|h^A))^{m-l}$$

We assume that all missiles are fired at live targets. This provides a simple first order model. Various assumptions can be relaxed at the cost of added complexity, or by adding simple degrade factors.

Recording The Position Information

During each stage of the battle, the tank battle is run for each set of initial conditions which has positive probability. After the calculation of the ending state distribution (at the end of a stage) for a particular set of initial conditions, the conditional distribution of the positions of the tank groups, given the number of surviving tanks on each side, and the fuel status of each tank group, is calculated. The overall conditional distribution of tank position is calculated by weighting the distributions calculated for each set of initial conditions by the probability of that set of initial conditions.

Returning Summary Information to the Top Level

During each stage of the battle, the tank battle is run for each set of initial conditions which has positive probability. After the calculation of the ending state distribution (at the end of a stage) for a particular set of initial conditions, the marginal distribution of the summary information which will be returned to the top level of the hierarchy is calculated. The overall marginal distribution of the summary information is calculated by weighting the distributions calculated for each set of initial conditions by the probability of that set of initial conditions.

The Air Battle

Each phase of the air battle consists of a pairing of adversary aircraft, followed by the firing of a number of missiles by each aircraft in a pair. The number of missiles fired per phase is an input parameter. In our initial model, we assume that the number of missiles fired during a phase is the same for F-aircraft and A-aircraft. Unpaired aircraft wait for subsequent phases. Alternative air battle models could easily be used in place of this initial model.

Consider the phase n air battle. The battle starts with A^F_{An} F-aircraft, and A^A_{An} A-aircraft. We let

$P_{nl}(i,j)$ = the probability that phase l of stage n of the air battle ends with i F-aircraft and j A-aircraft surviving.

During each phase of the air battle, opposing aircraft are "paired up". Unpaired aircraft (if one side has more aircraft than the other) wait until the next phase. We assume that each pair of aircraft participate in a stand-off dogfight: they fire only at each other, but the time between firing a missile and having it reach its target is long enough that the target also has time to fire a missile. During a phase, paired aircraft fire up to $N^F_{MAA} = N^A_{MAA}$ missiles at each other. We assume that both aircraft take their first shots at essentially the same time. They wait to see the outcome of the first exchange and, if both survive, fire again. This continues for $N^F_{MAA} = N^A_{MAA}$ exchanges. There are four possible outcomes to a single phase dogfight: either both aircraft survive, the F-aircraft is killed but the A-aircraft survives, the A-aircraft is killed but the F-aircraft survives, or both aircraft are killed. We let

P_{AA} = probability both aircraft survive.

P_{DA} = the probability that the F-aircraft is killed, but the A-aircraft survives

P_{AD} = the probability that the A-aircraft is killed, but the F-aircraft survives

P_{DD} = the probability that both aircraft are killed

If a phase starts with i F-aircraft and j A-aircraft, there will be $m = \min(i,j)$ standoff dogfights. The joint distribution of the number of dogfights which end with both aircraft alive (AA), the number which end with the F-aircraft dead and the A-aircraft alive (DA), the number which end with the F-aircraft alive and the A-aircraft dead (AD), and the number which end with both aircraft dead (DD) will have a multinomial distribution with probabilities P_{AA}, P_{DA}, P_{AD}, and P_{DD}.

Numerical Examples

In this section we will consider two simple numerical examples which illustrate the use of the model

Example 1

In example one, we focus upon aircraft allocation A group of 5 f-tanks starts in grid location (1,1) and approaches a group of 5 a-tanks which remain in grid location (1 3).

Each stage of the tank battle consists of 1 phase. During the first stage, the f-tanks do not move. During the second stage, they move to (1,2) and during the third stage they move to (1,3). We assume that the tanks remain in the high fuel state for the entire battle.

Each side starts with five aircraft. Each stage of the air battle consists of two phases. We consider four aircraft allocation strategies:

1. If your side has more (≥) aircraft than the other side, allocate half of your aircraft (rounded down to the nearest integer number) to the tank battle and the rest to the air battle; otherwise allocate all of your aircraft to the air battle;
2. Always allocate one half (rounded down to the nearest integer) of your aircraft to the tank battle, and the rest to the air battle;
3. Always allocate all of your aircraft to the tank battle;
4. Always allocate all of your aircraft to the air battle.

We set:

no. of missiles fired by an f-tank in a subphase	= 2
no. of missiles fired by an a-tank in a subphase	= 2
no. of missiles fired by an f-aircraft against a tank during a subphase	= 2
no. of missiles fired by an a-aircraft against a tank during a subphase	= 2
no. of missiles fired by an f-acft against an a-acft during a subphase of the air battle	= 2
no. of missiles fired by an a-acft against an f-acft during a subphase of the air battle	= 2

The probability that an f-tank will detect an a-tank increases from .4 to .6 to .8 as the f-tanks move across the grid. The corresponding probability that an a-tank will detect an f-tank increases from .4 to .6 to .8.

The probability that an f-aircraft will detect an a-tank is .3 for each stage and equals the probability that an a-aircraft will detect an f-tank.

The probability that a single missile from an f-tank will kill an a-tank increases from .2 to .3 to .4 as the f-tanks move across the grid. The corresponding probability that a single missile from an a-tank will kill an f-tank increases from .3 to .4 to .5.

The probability that a single missile from an f-aircraft will kill a detected a-tank equals .2 for each stage, and equals the probability that a single missile from an a-aircraft will kill a detected f-tank.

The probability that a single missile from an f-aircraft will kill an a-aircraft equals .3, and vice versa.

We now present examples of the types of results which can be determined by the model.

In tables 1 through 3 we show the mean number of surviving F-tanks, A-tanks, and the difference, respectively.

Similar tables could easily be calculated for the aircraft battle. In figures 3 – 6 we see the marginal distributions of surviving tanks and aircraft at the end of each stage of the battle, when both sides use strategy 1.

		A-Strategy			
		1	2	3	4
F-Strategy	1	0.489	0.399	0.027	0.719
	2	0.494	0.404	0.028	0.725
	3	0.943	0.564	0.057	1.313
	4	0.301	0.257	0.015	0.466

Table 1. Mean Number of F-tanks Surviving the Battle

		A-Strategy			
		1	2	3	4
F-Strategy	1	1.506	1.511	1.954	1.250
	2	1.232	1.236	1.207	1.061
	3	0.110	0.111	0.153	0.086
	4	2.030	2.033	2.527	1.739

Table 2: Mean Number of A-tanks Surviving the Battle

		A-Strategy			
		1	2	3	4
F-Strategy	1	-1.017	-1.111	-1.927	-0.530
	2	-0.738	-0.832	-1.179	-0.337
	3	0.834	0.454	-0.097	1.227
	4	-1.728	-1.776	-2.512	-1.273

Table 3: (Mean Number of F-tanks - Mean Number of A-tanks) Surviving the Battle

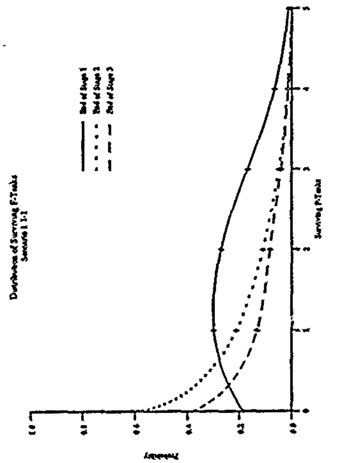
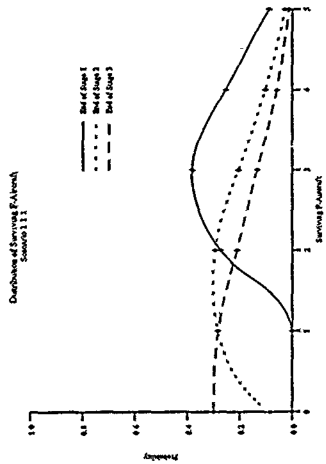


Figure 3: Scenario 1-1-1



Figur^ 5: Scenario 1-1-1

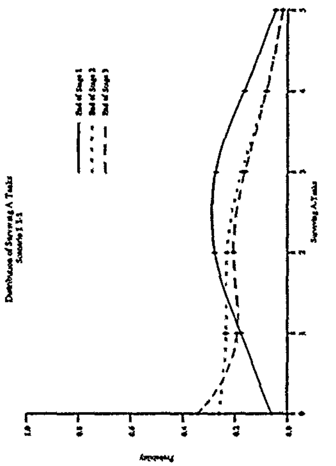


Figure 4: Scenario 1-1-1

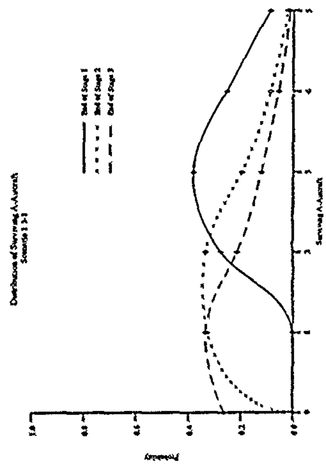


Figure 6: Scenario 1-1-1

Example 2

In example 2, we consider various movement rules. We consider a group of 5 f-tanks which start at grid location (1,1) and wish to attack a group of 5 a-tanks located at grid location (2,3). Along the route $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3)$, the tank vs. tank probabilities of detection and kill are lower than along the route $(1,1) \rightarrow (2,1) \rightarrow (2,2) \rightarrow (2,3)$, but the probability of detection of the f-tanks by the a-aircraft is higher along the first route. See tables 4 – 7 for details. We consider three possible movement rules:

1. The F-tank group remains at location (1,1) during the first stage of the battle. If, after the first stage, the F-tank group has more (≥) survivors than the A-tank group, the F-tank group takes the route $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3)$, moving one square per stage. Otherwise the F-tank group takes the route $(1,1) \rightarrow (2,1) \rightarrow (2,2) \rightarrow (2,3)$, moving one square per stage.
2. The F-tank group remains at location (1,1) during the first stage of the battle. Regardless of the outcome of the first stage, the F-tank group the route $(1,1) \rightarrow (2,1) \rightarrow (2,2) \rightarrow (2,3)$, moving one square per stage.
3. The F-tank group remains at location (1,1) during the first stage of the battle. Regardless of the outcome of the first stage, the F-tank group takes the route $(1,1) \rightarrow (1,2) \rightarrow (1,3) \rightarrow (2,3)$, moving one square per stage.

Both sides allocate their aircraft evenly between the tank and air battles.

In figures 7 to 10 we show the results for movement rule 1.

X \ Y	1	2	3
1	.3/.3	.05/.05	.05/.05
2	.3/.3	.3/.3	.4/.4

Table 4: Probability of Detection by/of F-tank at (X,Y) of/by A-tank at (2,3)

X \ Y	1	2	3
1	.2/.3	.05/.05	.05/.05
2	.2/.2	.2/.2	.3/.3

Table 5. Single-Shot Probability of Kill by/of F-tank at (X,Y) of/by A-tank at (2,3)

X \ Y	1	2	3
1	.3	.3	.2
2	.2	.2	.2

Table 6: Probability of Detection of F-tank at (X,Y) by A-aircraft

X \ Y	1	2	3
1	.2	.2	.2
2	.2	.2	.2

Table 7: Single-Shot Probability of Kill of F-tank at (X,Y) by A-aircraft

References

[1] I. Rubin and J. Baker, "Stochastic Modeling and Analysis for Maneuverable Multi-Battle and Multi-Phase C^3 Systems," Tech. Rep. IRI-FR-8702, IRI Annual Technical Report, October 1987.

[2] I. Rubin and I. Mayk, "Markovian Modeling of Canonical C^3 Systems Components," in *Proceedings of the 8th MIT/ONR Workshop on C^3 Systems*, June 1985.

[3] I. Rubin and I. Mayk, "Dynamic Stochastic C^3 Models and Their Performance Evaluation," in *Proceedings of the 9th Workshop on C^3 Systems*, June 1986.

[4] I. Rubin and I. Mayk, "Modeling and Analysis of Markovian Multi-Force C^3 Processes," in *Proceedings of the 9th Workshop on C^3 Systems*, June 1986.

[5] I. Rubin, J. Baker, and I. Mayk, "Stochastic C^3 Modeling and Analysis for Multi-Phase Battle Systems," in *Proceedings of the JDL C^2 Research Symposium*, (National Defense University, Fort McNair, Washington, D.C.), June 1987.

[6] I. Mayk and I. Rubin, "Paradigms for Understanding C^3, Anyone?," in *Proceedings of the JDL C^2 Research Symposium*, (National Defense University, Fort McNair, Washington, D.C.), June 1987.

[7] I. Mayk and I. Rubin, "Stochastic C^3 Modeling and Analysis for Maneuverable Battle Systems," in *Proceedings of the IEEE Military Communications Conference (MILCOM '87)*, (Washington, D.C.), October 1987.

[8] I. Rubin and J. Baker, "Stochastic Modeling and Analysis for Maneuverable Multi-Phase C^3 Systems," Tech. Rep. IRI-CC-AR-8701, IRI Annual Technical Report, January 1987.

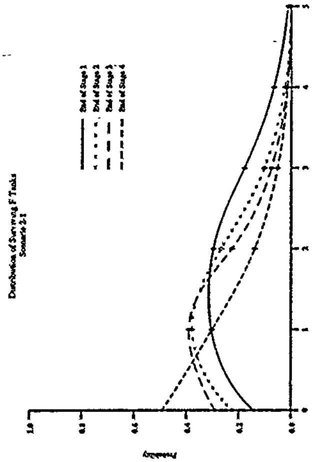


Figure 7: Scenario 2-1

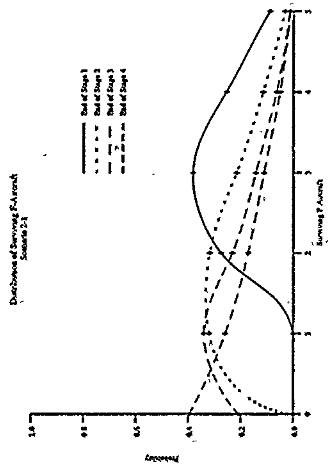


Figure 9: Scenario 2-1

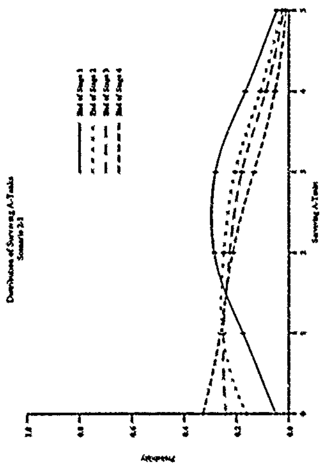


Figure 8: Scenario 2-1

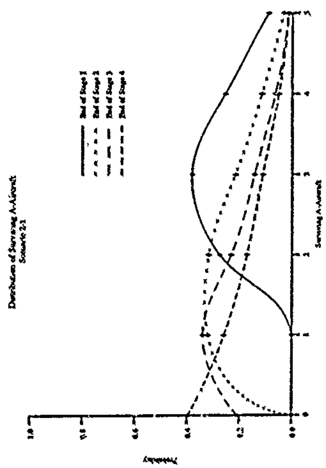


Figure 10: Scenario 2-1

A Self-Correlation Procedure for Data Fusion

Donald E. Brown, Paula S. Gabbert and Andrew R. Spillane

The Institute for Parallel Computation and Department of Systems Engineering
University of Virginia, Charlottesville, Va. 22901
804-924-5393

Abstract

Tactical data fusion is composed of a number of separate processes with the goal of supporting the complete assessment of the tactical situation. One of the component processes of data fusion is the correlation of sensor reports with existing data records to determine the presence of new targets and to update the status of previously noted targets. This paper describes a correlation procedure for data fusion. Current approaches, which use linear discriminant functions, serve as the basis for the procedure described here. However, whereas the current approaches use only subjective assessments of attribute importance, the procedure described here adaptively acquires attribute importance measures using information contraints generated by examples. This technique provides for an effective method of acquiring the correlation parameters. Additionally, the correlation process itself is described in terms of the bipartite matching problem and a new method for correlation matching is described.

1. INTRODUCTION

Data fusion is an activity which is becoming more important to military planners in all services. It has arisen from the need to process large amounts of data in dynamic scenarios with fluid battle formations, where commanders are dependent on sensor resources to provide an accurate picture of the environment.

While data fusion is important to all three services, the emphasis on the various functions that comprise data fusion changes with different areas of operation. The data fusion problem for the air and sea battles is associated with target tracking. In these operational areas, fast moving targets must be monitored for changes in position and intent. The ground forces problem changes less rapidly when viewed from the vantage point of the Corps commander. However, the target density and variety for the land battle is much higher than that found either on the sea or in the air. Additionally, the number and distribution of sensor resources in ground operations is very large. Finally, the ground forces' problem uses a different mix of sensors, relying less on active sensors, such as radar, and more on passive sensors to acquire the necessary information about the opposing force.

These characteristics of the ground forces data fusion problem have slowed the development of automated systems to support intelligence and target acquisition activities. The first generation of computer-based data fusion systems for supporting the land battle have only recently been deployed. Data on the effectiveness of these systems is just now starting to become available. Hence, the presence of a solid empirical foundation for work in this area is, understandably, lacking.

The fusion algorithms, which constitute this first generation, emphasize the rapid processing of massive amounts of incoming reports. The algorithms are not necessarily elegant or entirely effective, but efficient, easily maintained, and understandable by the users. The framework for the current algorithms was created by the developers of the BETA (Battlefield Exploitation and Target Acquisition) system. From this beginning the approaches to the ground forces data fusion problem have proceeded in a wide variety of directions ranging from expert systems to statistical pattern matching.

This paper considers extensions to BETA, known as BETA derived systems (BDS). BDS are proposed for fielded use and are designed to process incoming reports and produce an updated data base of targets or hostile entities in the environment. In addition to data fusion BDS perform other functions [1], some of which are, distribution of reports to system subscribers; display of the current battlefield situation; and targeting recommendations. However, the concern in this paper is with the data fusion activity in BDS. In particular, the focus here is centered around a specific data fusion function called self correlation. This paper presents several important new techniques for strengthening and improving the self correlation algorithm in BDS.

The next section provides a brief overview of the self correlation algorithm in BDS and section 3 extends this discussion by describing the process used in BDS to measure correlation between reports. Section 4 provides a new and needed approach to determining the contribution provided by each attribute in the BDS's measure of correlation. In section 5, an important problem with the matching algorithm in BDS is described and a solution provided. Finally, section 6 contains the conclusions.

2. SELF CORRELATION ALGORITHM

The purpose of the self correlation algorithm is to test incoming sensor reports to determine if they correspond to existing entries in the target data base or if they are new targets. The comparisons are performed using parametric data in the reports. The algorithm used in BDS is the basis for the discussion in this section. The basic steps in the algorithm are shown in figure 1, which is condensed from [1].

The first step in the process is to screen out data base entries which are not within a prespecified distance from the sensor target or are not of the same class as the sensor target. The target classes are decomposed in an in lusion hierarchy, an example of which is shown in figure 2. Each report includes the most specific classification the reporting sensor can make using the available parameters. A data base entry with a class on a path below the node in the hierarchy representing the sensor report is considered a candidate for correlation. Once all the data base entries below the sensor report's class have been exhausted, then data base entries on a direct path above the node of the sensor report are considered. A direct path in this case means that the path only goes through parent nodes

Hence, in figure 2 if a sensor report has a class "tank", then it can correlate with data entries with class "armored vehicles" but not with those with class "infantry fighting vehicles".

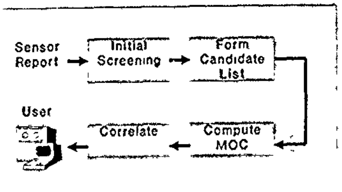


Figure 1 Correlation Algorithm

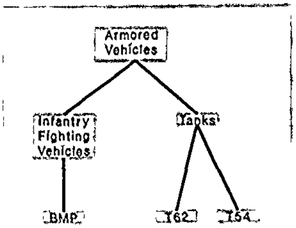


Figure 2 Inclusion Hierarchy

After initial screening of data base entries, the resulting candidate list is ordered on the basis of distance from the sensor's reported target location A measure of correlation (MOC) is then derived for the first entry in the candidate list. If the MOC is above a prespecified threshold, then the data base entry is updated to reflect the information in the sensor report. A MOC between this updated entry and the next member of the candidate list is computed and compared to the threshold. If the MOC in these comparisons is below threshold, then the next entry in the candidate list is examined. This process is repeated until the candidate list is empty.

There are two features of this algorithm that require closer scrutiny. The first is the method of computing the MOC's and threshold for use in discriminating candidate entries. The second is the procedure for matching sensor reports to candidates using the computed MOC. These are the topics in the remaining sections of this paper.

3. FIGURE OF MERIT APPROACH TO CORRELATION

The algorithm described in the previous section depends on the calculation of the MOC between two reports. There are a number of approaches to this calculation that have been proposed and implemented. Most of these are listed in [2]. However, the method of interest in this paper is the figure of merit (FOM) approach developed by Wright [3], because it serves as the basis for important implementations of BDS. The advantages and disadvantages of this approach vis-a'-vis its competitors will not be an issue here, although several important research directions are given in the conclusions. Instead, the FOM approach is taken as the appropriate method for computing MOC's, and techniques are provided to strengthen and extend it.

Sensor reports arrive for processing in batches Let $S_t = (S_{1t}, S_{2t}, \ldots, S_{nt})$ be a batch of sensor reports with $t = 1, 2, \ldots$ As the batches arrive, the elements are correlated with the elements of an existing data base of targets. Let $D_t = (D_{1t}, D_{2t}, \ldots, D_{mt})$ be the set of data base entries at the time of arrival of batch S_t. In order to avoid the accumulation of subscripts, throughout the remainder of this paper a single arbitrary batch of sensor reports is considered and labeled S.

The correlation between the elements of S and D is based on the parametric data in the reports. These data are used to determine the FOM's. Hence, for correlation purposes the data matter only insofar as their use in the FOM calculations. Thus, this paper considers both the sensor and data base reports to consist of an ordered set of attributes, which label the FOM's. In some cases the sensor data and the FOM labels or attributes are the same. Time of detection is an example. In other cases they are different. Location is an FOM label which corresponds with x, y and covariance data in the report.

Let $S_i = (s_{1i}, s_{2i}, \ldots, s_{ri})$ and $D_j = (d_{1j}, d_{2j}, \ldots, d_{rj})$ be the sets of attributes for the ith sensor report, $i = 1, 2, \ldots, n$ and jth data base entry, $j = 1, 2, \ldots, m$, respectively. There is no loss in generality in letting the number of attributes in sensor and data base reports be equal. If there is a missing attribute in a report, the correlation procedures are conducted only over the remaining attributes. Hence, in the FOM approach the number of attributes in both types of reports can be considered identical. Let $FOM_k(i,j)$ be the figure of merit between s_{ki} and d_{kj}, $k = 1, 2, \ldots, r$. The methods for calculating these FOM's are given in [3]. Two important properties of $FOM_k(i,j)$ for $i = 1, \ldots, n$, $j = 1, \ldots, m$, and $k = 1, \ldots, r$ that are required in the next section are: (1) $FOM_k(i,j) \geq 0$, and (2) $FOM_k(i,j) \in [0,1]$.

The correlation between S_i and D_j is based on combining the values obtained for $FOM_k(i,j)$, $k = 1, 2, \ldots, r$. However, the combination of these FOM's is weighted in accordance to the importance ascribed to the attribute in measuring correlation Hence, the measure of correlation between S_i and D_j is

$$MOC(i,j) = \sum_{k=1}^{r} w_k FOM_k(i,j) \qquad (1)$$

where $w_k \in [0,1]$, $k = 1, \ldots, r$ and $\sum w_k = 1$. If

$$MOC(i,j) \geq \tau, \qquad (2)$$

where τ is specified a priori, then S_i and D_j are correlated and the values for d_{kj} are updated.

The weights, w_k, in (1) are normally based on information about the sensors in reporting the attributes. Both the weights and the threshold, τ, are critical parameters to the correlation process. Allowing them to be set subjectively has its advantages if the user has sufficient experience. More than likely, users will not be able to effectively adjust these parameters. Additionally, if the parameters are set by the developer using archival data, then the system will be susceptible to errors in the presence of change in the environment. This is a critical problem, because most current data is based on peacetime activity. During war many, if not most, of the report parameters can be expected to change. The procedures for setting the weights and thresholds must be adaptive. Hence, if the FOM approach is to work effectively, then a method is required for generating weights and thresholds given exemplary data.

4. MEASURING ATTRIBUTE CONTRIBUTION

The derivation of weights for the attributes in (1) is essentially an information acquisition process. As examples of reports which should and should not correlate are provided, information is obtained, which is relevant to determining the weights and threshold. There are three problems which must be overcome in order to use this information for generating the desired values. The first is to formally define the inputs and outputs to a weight generating procedure. The second is to represent the information in the correlation examples in a usable and concise format. The final problem is to create a procedure that will automatically transform this information into the values required in (2).

Let $\Theta = (\theta_1, \theta_2, .., \theta_s)$ be a set of examples, called a training set. The elements of this set are ordered pairs, $\theta_i = (\xi_i, S_i)$. The second member of θ_i is a sensor report as defined above. The first member is a set of indices. In particular, $j \in \xi_i$ if S_i is from the same target as D_j.

Given Θ, the problem is to generate $w = (w_1, w_2, ..., w_r)$ and τ for use in (2). The w which is produced must be within Σ, where

$$\Sigma = \{w : \sum_{i=1}^{r} w_i = 1, w_i \in [0,1] \text{ for } i = 1, 2, ..., r\}.$$

The properties listed in section 3 for the FOM's constrain τ to be within [0,1]. Hence both the inputs, Θ, and the outputs, w and τ, are well defined. It remains, however, to create a suitable representation for the information in Θ, and to describe a procedure capable of converting this information into w and τ, which is both justifiable and tractable.

Given a training example $\theta_i \in \Theta$, a set of inequality constraints of the form, $MOC(i,j) > MOC(i,k)$ for $j \in \xi_i$ and $k \notin \xi_i$, are generated. For each member of Θ a set of these constraints are formed. For a training set Θ, let

$$\Omega = \{w : MOC(i,j) > MOC(i,k) \text{ for } j \in \bigcup_{i=1}^{s} \xi_i, k \notin \bigcup_{i=1}^{s} \xi_i, w \in \Sigma\} \quad (3)$$

The constraints generated by the training set are called consistent if Ω is not empty. Approaches to the problem of inconsistent constraints are discussed in the conclusions. For the remainder of this development it is assumed that the constraints are consistent.

With the representation in (3) for the information provided by the examples, the problem now is to convert this information into weights and threshold for use in (2). Essentially, a measure is needed to describe the information content in (3). This measure, H(w), should satisfy the following reasonable properties:

1. H(w) is a maximum when $\Omega = \Sigma$, and nothing else is known about w.
2. H(w) is a minimum when $w_i = 1$ for some i, or w equals its prior estimate.
3. H(w) increases monotonically as r increases.

These properties lead naturally to the traditional measure of information, entropy [4]. For the weights in (1), H(w) is defined as,

$$H(w) = -\sum_{i=1}^{r} w_i \ln w_i, \quad (4)$$

where ln is the logarithm to the base e and $0 \ln 0 \equiv 0$. When information is known a priori about the weights, then a related measure of information, relative entropy, is used. Relative entropy is defined as,

$$I(p) = \sum_{i=1}^{r} w_i \ln (w_i/w'_i), \quad (5)$$

where w'_i for $i = 1, 2, ..., r$, is a prior estimate of w.

With these measures it is now possible to obtain the w from the information in Θ as represented by Ω. This is done by maximizing (4) subject to $w \in \Omega$. If a prior estimate of w is available then a minimization is performed using (5) subject to the same constraints. In either case, the result of this process is a unique value for w, which represents all the information in Ω, and no more.

This process of generating the weights for (2) produces a solution which is not computationally expensive and is based on the principles of information theory. With regard to the first point, the special structure of the optimization problem using either (4) or (5) is easy to exploit. In practice the solution time for these nonlinear programs is not much different from a comparably sized linear program using the simplex method. As regards the second point a description of inferential axioms that are satisfied by the weight generating procedure proposed here are in [4].

The only remaining requirement before applying the discriminator in (2) is to produce an estimate for τ. This is done by letting

$$\tau = \max_{\substack{k \notin \xi_i \\ i=1,...,s}} \{MOC(i,k)\} \quad (6)$$

The MOC values in (6) are calculated when the constraints in (3) are formed. Hence, the only additional computational requirement to finding τ is to compare the current maximum value with any new values found from the θ_i.

5. MATCHING PROCEDURES

A final problem with the algorithm presented in section 2 concerns the procedures for matching an incoming sensor report with a candidate report from the data base. Using the procedures described in that section as they are implemented in existing BDS, there is potential for overcorrelation. Overcorrelation is defined as correlating a sensor report with a data base entry, when the two reports represent different target entities. Undercorrelation is failing to correlate a sensor report with a data base entry, when both are derived from the same target. Of the two types of errors, overcorrelation is considered the most serious by military planners.

The reason overcorrelation is the more serious problem is not hard to understand. If a data base record is contaminated with information from another report, this could easily produce underestimates of the number and type of target entities in the environment. Situation assessments based on these contaminated data base records will mislead the commander into expecting fewer opposing systems than are actually present. Further, overcorrelations are difficult for a human analyst to detect and correct in an automated system.

Undercorrelation errors also present difficulties for the analyst. These errors contribute to the proliferation of spurious targets in a situation assessment and overestimates of the opposing force. It has been our experience in simulation tests of BDS that undercorrelations are more numerous than overcorrelations by an order of magnitude in runs with less than one hour of data. However, undercorrelations are easier for an analyst to detect, because when reports are posted the analyst can note that the number of entities in a particular location is much higher than the known order of battle would allow. Armed with this information the analyst can make manual corrections to the data base and present a more realistic picture of the situation to the commander. Because overcorrelations create contaminated records in the data base, which are extremely difficult to correct, this type of error is the more important to guard against.

To understand how overcorrelation can occur, consider the targets shown in figure 3. The circles in the figure depict error ellipses for the locations of a single sensor report (S_1) and two candidate reports (C_1 and C_2). If the two candidates are at roughly the same distance from the sensor report, and C_2 is from the same target as S_1, while C_1 is not, then there is the potential for overcorrelation. C_1 might be examined first, and correlated with S_1 because of its proximity. When C_1 is updated the new values for the attributes might not correlate with C_2. This is the problem identified by [5], and he showed that it is not difficult to create reasonable scenarios where significant overcorrelations can occur.

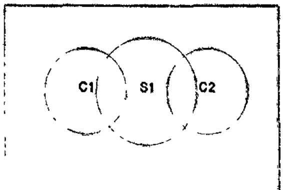


Figure 3. Overcorrelation Example 1

A related problem is shown in figure 4. In that case there are two sensor reports (S_1 and S_2) and one candidate report (C_1). If report S_2 and C_1 are from the same target and S_1 is not, then there is again the potential for overcorrelation. In this case if S_1 arrives first, and correlates with C_1 then the updated C_1 might not correlate with S_2 when it arrives. Again, it is relatively easy to create scenarios that produce significant overcorrelations of this sort.

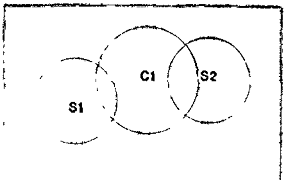


Figure 4. Overcorrelation Example 2

The problem depicted in figure 3 can be corrected by modest changes to the algorithm in section 2. The problem of figure 4 requires more fundamental changes. Essentially, the solution to both problems can be approached, but not solved, by the application of bipartite matching procedures. A bipartite graph is a graph in which the nodes are partitioned into two sets and no two nodes in the same set are adjacent. In the matching problem no two arcs are allowed to be incident to the same node.

There are at least three classic objective functions used in bipartite matching problems [5]. The first finds the maximum number of arcs in a matching. The other two are for problems with values or weights on the arcs. One of these seeks a maximum cardinality matching in which the arc with the minimum weight is maximized. The other objective for an arc-weighted matching is to maximum the sum of the weights in the match. Efficient algorithms are available for all of these problems.

Unfortunately, none of these objective functions completely describe the objective for this problem. In fact the bipartite matching problem does not completely fit the problem that the algorithm in section 2 is attempting to solve. However, the insight provided by considering bipartite matching procedures is sufficient to suggest a simple modification to the algorithm in section 2 that will improve its performance.

The proposed modification to BDS is called the Match algorithm and is shown in figure 5. Some additional notation is used in figure 5. Let (S,D,A) be the bipartite graph which includes S and D. Then $(i,j) \in A$ ($i=1,\dots,n$ and $j=1,\dots,m$) if and only if $MOC(i,j) > r$. The values in this last inequality are found using the procedures in section 3. Also, $CL_i \equiv [(D_{k_1}, MOC(i,k_1)),\dots,(D_{k_z}, MOC(i,k_z))]$, where $(i,k_j) \in A$ for $j = 1,\dots,z$, and $MOC(i,k_1) \geq MOC(i,k_2) \geq \dots \geq MOC(i,k_z)$. Thus, CL_i is the ordered candidate list for $S_i \in S$. CL_{ij} is the j^{th} ordered pair in CL_i and CL_{ijl} is the l^{th} member of CL_{ij}.

```
0. Let n = |S|; m = |D|; CL_i = Ø, ∀ i = 1, 2, ..., n
1. For i = 1, 2, ..., n and j = 1, 2, ..., m
   Compute MOC(S_i,D_j)
   If MOC(S_i,D_j) ≥ r then put (D_j,MOC(S_i,D_j)) in CL_i
   Sort CL_i
2. For i = 1,...,n
    If CL_i = Ø then remove S_i from S and put it in D
     If S = Ø then halt
     Else
       Compute MOC(S_i,D_m+1) ∀ S_i ∈ S
       Update CL_i for MOC(S_i,D_m+1) ≥ r
       Go to 2
   Else Correlate S_i* with CL_i*11
   where i* = argmax CL_k11 for k = 1,...,n
     Remove CL_i*1 from all CL_i where i ≠ i*
     Remove CL_i*11 from D
     Update CL_k using correlated values
     Go to 2
```

Figure 5 Match Algorithm

The algorithm in figure 5 can be summarized as follows. First, for all members of S compute the MOC with all members of D. In practice, this is only done for members of both sets within a distance threshold. The threshold is determined by sensor characteristics. Next, match the $S_{i^*} \in S$ and $D_{j^*} \in D$, where i^* and j^* are the values which maximize $(i,j) \in A$. Third, update D_{j^*}, eliminate S_{i^*} from the list of sensor reports in S, and return to the second step if S is not empty.

The modification in figure 5 is based on the use of batch processing of reports. Batch processing is feasible because of the high rate of incoming reports. However, a simple modification to BDS is to use batch sizes of one. This change reduces the potential for overcorrelation with minimal change to the existing implementations of BDS. In general, the modification reduces the overcorrelation problems previously described without causing a concomitant increase in undercorrelations. The modification also has potential for implementation on a parallel architecture.

6. CONCLUSIONS

The BDS self correlation algorithm described in section 2 served as the basis for the discussion in this paper of the ground force's data fusion problem. Two improvements to this algorithm were presented: (1). A method was provided for obtaining the attribute weights and threshold used in calculating the measure of correlation between reports; and (2). An improved matching procedure was described for reducing the potential for overcorrelation.

The procedure for acquiring weights assumed a consistent constraint set in (3). It is reasonable to expect this assumption will be violated by errors in the reports, sensor inaccuracies, and other noise in the environment. Hence, methods are required to cope with inconsistencies that can make $\Omega = \emptyset$. One approach which is conceptually simple, but of questionable implementation potential is to identify and remove reports creating inconsistencies. The problem with implementing this approach is that each report must be compared with every other in terms of its impact on the contraints. The comparison must be made to find the small number of reports causing inconsistencies with the remaining reports. However, inconsistencies can occur in a variety of ways, and checking different combinations of reports for different types of inconsistencies rapidly produces combinatorial explosion.

A more efficient approach to this problem is simply to allow the inconsistencies. The optimization of (4) or (5) is conducted over disjoint sets. These sets can be defined sequentially by the examples in the training set. The solution is then taken to be the optimal value from among these separate optimizations. This approach has promise because it is computationally reasonable. However, it does present the possibility that a spurious report will significantly affect the results. Checking for this effect is much easier than checking for reports which produced the inconsistencies. It seems likely that efficient algorithms can be developed to perform this type of data analysis.

As noted above this work has concentrated on improving the existing correlation procedures embodied in BDS. The improvements suggested are significant and should enhance the performance of this automated data fusion system. However, it seems unlikely that researchers and developers will be able to get many additional capabilities out of the BDS self correlation framework. Some of these additional capabilities include the need to aggregate and cross correlate reports, and produce situation assessments. The first two capabilities are features examined by the original BETA developers. There is also a clear need for increased processing speed, which is currently available in parallel machines. To take advantage of these architectures, new algorithms for data fusion are required. However, this should not be viewed as a call to continue blindly attempting to apply every new technology to the data fusion problem. Rather, there is an opportunity now to critically examine the performance of current approaches to data fusion and use this empirical evidence to suggest promising improvements.

Acknowledgement

The work reported in this paper has been partially supported by the Jet Propulsion Laboratory under grant number 95772.

REFERENCES

1. Mutchler, C. N., "Analysis of the Limited Operational Capability Europe (LOCE) Self Correlation Algorithm," Mitre briefing, The Mitre Corp, McClean, Va.

2. Hall, D. L. and Linn, R. J., "Algorithm Selection for Data Fusion Systems," *Proc. of the 1987 Tri-Service Data Fusion Symposium*, vol. 1, Naval Air Development Center, Warminster, PA.

3. Wright, F. L., "The Fusion of Multisensor Data," *Signal*, Oct 1980, 39-43.

4. Shore, J.E. and Johnson, R.W., "Axiomatic Derivation of the Principle of Maximum Entropy and the Principle of Minimum Cross-Entropy," *IEEE Trans. Information Theory*, IT-27, 472-482.

5. Lawler, E. L., *Combinatorial Optimization, Networks, and Matroids*, Holt, Rinehart, and Winston, New York, 1976.

AN APPROACH TO MULTI-SENSOR DATA FUSION BASED UPON COMPLEX OBJECT REPRESENTATIONS

Sheldon Gardner

Naval Research Laboratory
Washington, DC 20375

Abstract

Automation of data fusion in C^2 applications requires an understanding of the mechanisms by which humans fuse multi-sensor information. Motivated by the neurobiology of the human visual system [HVS], we introduce a hierarchical relational graph [HRG] paradigm for the representation, fusion, and classification of static images. Mathematically, at each level in the hierarchy, image information is represented by means of a relational graph (complex object) composed of nodes (simple objects) with edges (relations) defined between nodes. In the HRG paradigm, a simple object decomposes into a complex object at the next highest resolution level. We discuss application of the HRG representation to image representation, fusion, and classification.

1. Introduction

Although the subject of multi-sensor data fusion has been discussed within the C^2 community for many years, progress towards a unified theory has been slow. The problem is a difficult one which requires, for solution, a detailed knowledge of how humans fuse multi-sensor data.

We define data fusion as the combination (averaging) of data in order to produce an internal model. When multi-sensor data of different dimensionality is fused, the nature of an internal model is unclear. However, when data of the same dimensionality (e.g. 2-D image data) is fused, the nature of the model can be unambiguous. Once we have an unambiguous model, and a 'distance' metric, we can measure the distance between new data and the model (here we use 'distance' as an abstract measure related to correlation). This enables us to classify new data, subject to the limitations of the distance metric. In this paper we will discuss an approach to the fusion of static images. The dynamic image fusion problem can be treated in an analogous manner.

In the HVS, image recognition involves a blend of low level (early vision), and high level (cognitive) processing. Evidence exists that the early vision process can be modeled by a multi-dimensional Gabor representation [1-6]. It is reasonable to assume that the Gabor coefficients are coded as data inputs to a form of distributed associative memory [7,8].

In the next section, we describe a representation of a static image which we call a hierarchical relational graph [HRG]. In Section 3, using the HRG methodology, and the properties of the HVS, we describe an approach towards automated fusion and classification of static images.

2. Hierarchical Relational Graph Representation

Motivated by the properties of the HVS, we represent a static image by means of a hierarchical relational graph [HRG]. At each level of the hierarchy, we construct a set of nodes (simple objects), and a relational graph (complex object) based upon the relations between nodes. However, at the next lowest level in the hierarchy (finer resolution), each node is treated as a complex object, composed of it's own set of connected simple objects. Although, we describe the HRG structure in a top-down manner, in the HVS, data flow actually takes place in a bottom-up manner, since image information is first processed in the visual cortex then sent to other areas of the brain, such as the cerebral cortex. Recognition of a face can be used as a simple example of this process. Starting with the placement of features (e.g. eyes, nose, etc.) we recognize a face as a complex object composed of simple objects (features). On the next hierarchical level we examine individual facial features.

Fig. 1 illustrates the HRG model for a single level. Fig.1a shows the pixel representation of four simple objects. These simple objects are combined to form the complex object illustrated in Fig.1b, represented by the relational graph (RG) shown in Fig.1c. The nodes of the RG shown in Fig.1c, labeled 1-4, consist of internal representations of the four simple objects. In this simple example the edges represent a mapping of the spatial relationships between the simple objects.

The HRG for two hierarchical levels is illustrated in Fig.2. Figure 2a shows the complex object of Fig. 1, which, on the second hierarchical level is treated as a simple object. Fig. 2b shows the two-level HRG for this pixel image. Although we do not intend to restrict the general HRG paradigm to this simple case, for purposes of clarity, we have chosen to illustrate the HRG for a second level process identical to the first. Since the object relations in Fig.2 are identical to those of Fig.1, it is easy to see how continuation of this process will generate the HRG for a multi-scale self-similar image (fractal) [9].

The HRG paradigm is intended to suggest a morphological model for image representation and fusion suitable for machine vision applications. We do not intend to propose the HRG structure as a neurophysiological and psychophysical model of the HVS. However, it is appropriate to comment on the correspondence between the HRG representation and certain properties of the HVS. As visual recognition proceeds, low level processes (early vision) transition to higher order (cognitive) processes. In the HVS, the number of hierarchical levels used for low level (early vision) and high level (cognitive) representation will be a function of the maturity of the subject. In the HVS the average number of cognitive levels should be related to the storage capacity of short-term memory, which is a function of the physical maturity (age) of the subject.

Assume that object discrimination in the visual cortex is based upon a binary decision process. Following McLaughlin [10] and Barrett [11], an image class is defined by the presence or absence of attributes. At level 0, the demand is for the system to process $2^0 = 1$ concept (memory span equals 1). At level 1, there is a capacity to retain $2^1 = 2$ concepts simultaneously. Class concepts can be formed, but no distinction can be made between more than one object and its environment (which requires at least three concepts). The mature short-term memory capacity corresponds to level three with the ability to process up to $2^3 = 8$ concepts simultaneously (5-8 digits) [12].

3. Classification and Fusion of Static Images

In a machine vision application the the description of objects within a scene, the number of hierarchical levels, and the choice of spatial resolution at each level are problem dependent. However, the processes associated with HRG image representation, fusion and classification are generic. These processes are illustrated in block diagram form in Fig.3. Fig.3a shows the HRG data extraction process. This process consists of image processing, followed by spatial filtering at each hierarchical level. In Fig.3b. the HRG data is combined to produce a model (normal image). The classification of image data, which consists of measuring the distance between the data and a model, is illustrated in Fig.3c. In a machine vision application, these functions could be implemented by a multi-layer neural network computer similar to Fukushima's Neocognitron [13].

Summary

Motivated by the properties of the HVS, a methodology for fusion and classification of static images, based upon an HRG paradigm, has been introduced. On a single level basis, the HRG is similar to earlier concepts of knowledge representation based upon semantic nets [14]. However, the HRG approach, which is modeled after the HVS, incorporates a blend of high level-low level processing within a hierarchical structure. The HRG approach to image fusion may be extended to multi-sensor data fusion, provided the problem of sensor dimensionality can be solved. The HRG methodology may also have application to recognition and tracking of moving images, as well as in modeling of other biological systems, such as human speech recognition.

References

1. Gardner, S., Morphology of Neural Networks in the Mammalian Visual System, International Neural Network Society Annual Conference, Boston, Mass., Sept. 1988.

2. Gardner, S., Ultradiffusion, Scale Space Transformation, and the Morphology of Neural Networks, IEEE Annual International Conference on Neural Networks, San Diego, Cal., July 1988.

3. Daugman, J., Uncertainty Relation for Resolution in Space, Spatial Frequency, and Orientation Optimized by Two-Dimensional Visual Cortical Filters, J. Optical Society of America A., Vol. 2, No. 7, 1985.

4. Daugman,J., Two-Dimensional Spectral Analysis of Cortical Receptive Field Profiles, Vision Research, Vol.20, p. 847, 1979.

5. Daugman, J., Kammen, D., Image Statistics, Gases, and Visual Neural Primitives, Proceedings of the IEEE First International Conference on Neural Networks, M. Caudill, C. Butler, Eds. IEEETH00191-7, 1987.

6. Daugman, J., Complete Discrete 2-D Gabor Transforms by Neural Networks for Image Analysis and Compression, IEEE Trans. ASSP, Vol. 36, No. 7, p. 1169, July 1988.

7. Wechsler, H., Zimmerman L., 2-D Invariant Object Recognition using Distributed Associative Memory, IEEE PAMI, 1988 (to appear).

8. Malachi Char, J., Cherkassky V., Wechsler, H., Zimmerman G., Distributed and Fault-Tolerant Computation for Retrieval Tasks Using Distributed Associative Memories, IEEE Trans. Computers, Vol.37, No. 4, April 1988.

9. Gardner, S.(1987) Statistical Dynamics of Ultradiffusion in Hierarchical Systems, Proceedings of the IEEE First International Conference on Neural Networks, M. Caudill, C. Butler, Eds., IEEETH00191-7.

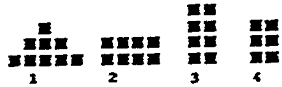
10.McLaughlin, G.H., Psycho-logic: A Possible Alternative to Piaget's Formulation, Brit. J. Ed. Psychol. Vol.33, p.61, 1963.

11.Barrett, T.W., Entropy and Symmetry: Their Relation to Thought Processes in a Biological System, Kybernetik, Vol.6, p.102, 1969.

12.Miller, G.A., The Magical Number Seven, Plus or Minus Two, Psychol. Rev. Vol. 63, p. 81, 1956.

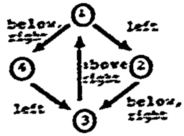
13 Fukushima K., Neocognitron: A Hierarchical Neural Network Capable of Visual Pattern Recognition, Neural Networks, Vol.1, No. 2, 1988,

14. N.V.Findler, Associative Networks: Representation and Use of Knowledge by Computers, Academic Press, 1979.



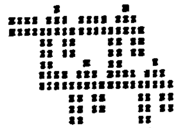
a. Simple Objects

b. Complex Object

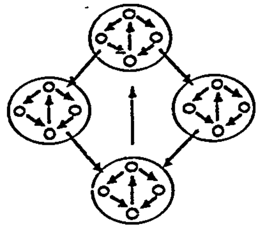


c. Relational Graph (RG)

Figure 1. Relational Graph Representation.

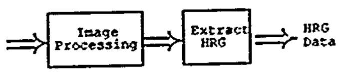


(a) Second level pixel representation.



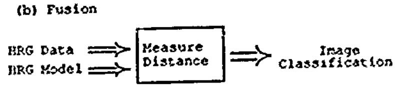
(b) Second level RG representation.

Figure 2. Hierarchical Relational Graph.



(a) Representation

(b) Fusion



(c) Classification

Figure 3. HRG Fusion, and Classification.

MULTITARGET TRACKING AND THREAT EVALUATION IN AIRBORNE SURVEILLANCE*

Thomas Kurien
Adam Caromicoli

ALPHATECH Inc.,
111 Middlesex Turnpike
Burlington, Massachusetts 01803

ABSTRACT

In this paper we define a processing architecture for an airborne surveillance system which simultaneously addresses the problems of multitarget tracking and threat evaluation. We show that the different nature of the two problems can be addressed by a processing architecture which supports both numerical and symbolic processing. Numerical processing algorithms based on the mathematics of estimation theory are applicable to the problem of multitarget tracking. Symbolic processing algorithms based on the domain knowledge of the airborne surveillance mission are applicable to the problem of threat evaluation. The multitarget tracking algorithm has been implemented and demonstrated in FORTRAN. Simulation results obtained with a prototype of the threat evaluation algorithm implemented in PROLOG show that the evaluated threat conditions are consistent with those actually simulated.

1. INTRODUCTION

The mission of the E-2C Hawkeye is to conduct airborne surveillance for a Naval battle group. To carry out this mission, the E 2C surveillance system has a suite of onboard sensors, communication systems linking it to other platforms in the battle group, onboard data processing capabilities , and a crew of E-2C surveillance operators. The onboard sensor suite includes a UHF radar, an identify-friend-or-foe (IFF) system which can interrogate friendly targets, and a passive detection system (PDS) which can detect and identify radars on board targets. Target track data available to other platforms in the battle group is accessible to the E-2C over the communication links. Using the sensor data and the track data over the communication links, the surveillance data processing computer has to track the position and velocity of all targets within the surveillance volume and identify the types of all tracked targets. Based on this track and identity data for all targets, the E-2C operator has to evaluate a measure of the threat faced by the battle group.

During the course of the past few years, the amount of surveillance data accessible to the E-2C has increased several fold. This has been a result of both the extended coverage regions of the sensors made possible by improvements in sensor and communication systems on board the E-2C, and the vast number of platforms with associated electronic warfare systems (e.g., ESM, ECM, and ECCM) present in modern tactical scenarios. The large amount of surveillance data has burdened the E-2C surveillance operators with tasks related to monitoring the results of the current data processing algorithms. Such tasks include maintaining track continuity of radar tracks and correlating data from different sensors. Not only does this cause the operator's attention to be consumed by the task of monitoring and correlating the vast number of tracks - a task for which he (or any human) is not really suited - but it also causes him to neglect his primary duty of evaluating threat to the battle group - a task for which he is better suited.

To keep pace with this data-rich environment, a High Speed Processor (HSP) which has large data processing capability has been developed and incorporated into the E-2C. This larger data processing capability on the E-2C presents an opportunity to incorporate advanced data processing algorithms which will better support the E-2C surveillance operators and thereby enhance the mission effectiveness of the E-2C system. The objective of our effort is to develop a *prototype* of such a data processing algorithm. Notice that the data processing algorithm is intended to serve as a decision aid to the E-2C surveillance operator and not as a replacement for him. We do not foresee that a human operator can be replaced for a function as critical as threat evaluation in actual tactical warfare - at least not with existing or near term algorithm technology.

The objective of this paper is to discuss the overall architecture of the decision aid and to examine in detail the threat evaluation algorithm. With this objective in mind, we have organized this paper as follows. Section 2 provides an overview of the architecture of the decision aid. The Correlation and Tracking, and the Identification algorithms are discussed briefly in Section 3. Details of the Threat Evaluation algorithm are provided in Section 4 and simulation results which demonstrate the effectiveness of this algorithm are provided in Section 5.

2. ARCHITECTURE OF THE DECISION AID

The first step in the design of a data processing algorithm is to clearly define all a priori knowledge available about the process. For the E-2C airborne surveillance problem, this entails two bodies of knowledge:

1. Physical laws that model well understood phenomena such as the dynamics of target motion (e.g., Newton's laws of motion), and the measurements from the E 2C sensors (e.g., electromagnetic propagation and reflection laws for the radar signal). These also include statistical models which account for the lack of precise knowledge of the dynamics (e.g., effect that wind has on aircraft motion) and measurements (e.g., measurement uncertainty introduced by randomly changing characteristics of the medium through which the radar signal propagates), and geometrical relationships which account for sensor-target geometry (e.g., Pythagorean theorem).

*Research supported by Office of Naval Research under Contract No. N00014-87-C-0757.

2. Man-made laws such as doctrines used in tactical warfare (e.g., flight corridors for friendly aircraft approaching an aircraft carrier), man-made rules for evaluating threat (a fighter aircraft has tactical air superiority over a bomber), and known facts (e.g., the BADGER is a Soviet bomber which carries a *type A* tail warning radar, has a maximum speed of *B knots*, and is equipped with *type C* antiship missiles which can be launched from a range of up to *D nautical miles*).

Data processing algorithms designed to utilize these two bodies of knowledge have different characteristics. For the former body of knowledge, algorithms may be designed which are based on a precise mathematical framework and which follow a fixed sequence of steps to process the data. Further, they are entirely *numerical* in that these algorithms process numerical data while carrying out transformations (dynamic, geometrical, averaging) to arrive at the result.

For the latter body of knowledge, one cannot formulate precise numerical relationships because either they do not exist or they are not known. Variables associated with this body of knowledge are described in a symbolic or aggregated form. One cannot formulate a fixed sequence of steps to arrive at the desired result, rather one has to process all the available data to derive a recognizable pattern from it. This form of processing is generally *symbolic* in nature. Since humans process data in a symbolic form, they prefer to interface with symbolic algorithms rather than with numerical algorithms.

Since the E-2C surveillance problem entails both bodies of knowledge and since the data processing algorithm has to interface with the E-2C operator, the design of the data processing algorithm requires both forms of processing. A conceptual architecture for the combined processing in the decision aid is shown in Fig. 1. The figure indicates a partitioning of the two forms of processing with the numerical processing algorithms on the one side and the symbolic processing algorithms on the other. The numerical algorithms are constructed based on models and associated parameters of the physical laws. They process the numerical data from sensor reports to evaluate numerical results. Symbolic algorithms utilize rules and facts to process the symbolic data. They evaluate the threat conditions for the battle group. The two forms of processing are linked by a common data base which contains intermediate and partial results produced by either form of processing and required by the other. Specifically, the numerical algorithms compute numerical results that enable certain rules to *fire* within the symbolic algorithms, and the symbolic algorithms will request new results to be evaluated by the numerical algorithms. Explanations for generated results are produced by the symbolic algorithms in response to E-2C operator queries.

A functional architecture of the decision aid is shown in Fig. 2. The two rectangles drawn with broken lines represent the numerical and symbolic processing partitions discussed above. Specifically, the algorithms for correlation and tracking, and target identification are numerical algorithms, the remaining algorithms are symbolic. We have chosen titles for the modules (Level 1, 2, and 3 Processing) and the submodules (Situation Abstraction, Situation Assessment, etc.) which are consistent with the lexicon defined by the Fusion Subpanel of the Joint Directors of Laboratories (JDL) C^3 Research and Technology Program [1]. The functions of each of the submodules are as follows. the (Multisensor Multitarget) Tracking and Correlation submodule correlates the sensor reports over time and sensors and evaluates the position and velocity of all targets, targets are identified based on the sensor reports by the Identification submodule, target characteristics relevant for threat evaluation are abstracted by the Situation Abstraction submodule, the Situation Assessment submodule evaluates the hostile target threat and the ability of friendly forces to engage the hostile targets effectively on a one-on-one basis, finally, the Threat Assessment

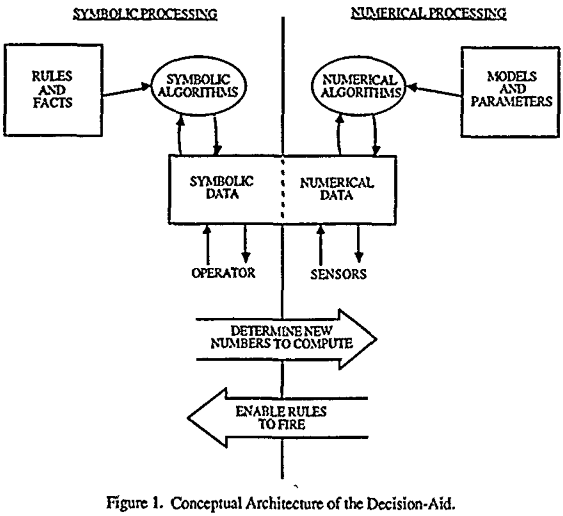


Figure 1. Conceptual Architecture of the Decision-Aid.

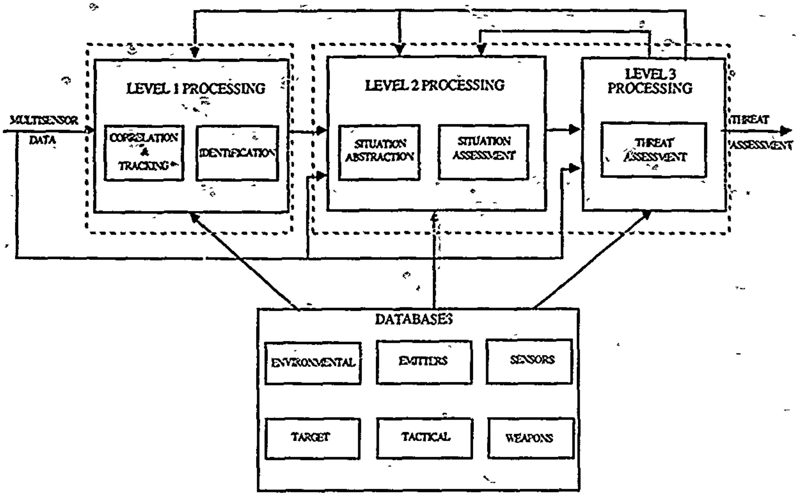


Figure 2. Functional Architecture of the Decision Aid.

submodule evaluates the collective hostile target threat and the ability of the battle group to engage the enemy effectively.

3 TARGET TRACK CORRELATION AND IDENTIFICATION

Both the Correlation and Tracking, and the Identification problems can be modeled fairly precisely. Based on these models, we have developed correlation and tracking, and identification algorithms which follow a fixed sequence of steps to process the sensor data. For reasons discussed in Section 2, we classify these as numerical algorithms. We will discuss these numerical algorithms briefly in this section.

3.1 Correlation and Tracking Algorithm

The multisensor multitarget correlation and tracking problem has been studied extensively in the past [2]. Some of these, especially those currently used in operational systems, are based on heuristic rules formulated using intuition and experience from actual surveillance scenarios. Such algorithms work well in specific situations; however, since they are not based on a general model, they fail to handle situations other than those considered during the design.

Other correlation and tracking algorithms are based on principles of statistical estimation theory [3-5]. These approaches define a precise mathematical model for the system, formulate the equivalent mathematical problem, and develop the algorithm to compute the optimal solution. Such approaches work well in most situations, but they generally have large computational requirements. However, carefully designed heuristic rules may be incorporated within the optimal solution which dramatically cut down the computational requirements of the optimal algorithm without significant loss in performance. In fact, such an algorithm has been designed and demonstrated in real time using real data [6]. The same algorithm has been incorporated in our decision aid. We will overview the mathematical approach and the heuristics used to develop this multisensor multitarget tracking algorithm here, the reader may refer to [5] for a detailed description of the algorithm.

The correlation and tracking algorithm uses the mathematical framework of Hybrid State Estimation to formulate the solution methodology. The general hybrid state model consists of continuous-valued states and discrete-valued states. Measurements related to the hybrid state are used to evaluate an optimal (minimum-mean-squared-error or maximum-a-posterior) estimate of the hybrid state. Variables in multisensor multitarget tracking can be identified with the generic hybrid model as follows: the state (position and velocity) of all targets constitute the continuous-valued state; indicators for target status (constant velocity model, maneuver model) and sensor report status (associated with target, false alarm) constitute the discrete-valued state, the noisy measurements of range, angle, etc. from the sensors constitute the measurements.

The optimal estimator for the hybrid state based on the measurements is easy to formulate, however, the postulation of all possible values of the discrete-valued state (referred to as global hypotheses) and computation of their likelihoods pose a difficult problem. In order to construct a practical algorithm all the unlikely global hypotheses have to be pruned away. The key techniques that we have used to prune unlikely hypotheses are outlined below.

N-scan Approximation. The optimal correlation and tracking algorithm requires that each track postulated for a target be associated with each sensor report since all such associations may be possible. In reality, we know that each target should have only one track corresponding to an association with the report it generates or to no association in the case it is not detected. The N-scan approximation w for N scans before it resolves multiple associations made for a target in a particular scan.

Gating : Gating is a screening technique that eliminates unlikely associations of sensor reports with targets The gating process consists of constructing a region (gate) around the predicted target position, and selecting only those reports which lie within this region to be associated with the target track. Gating proves to be very effective in cutting down the number of unlikely target-to-report associations and has been used in most of the tracking algorithms.

Classification : Another powerful screening technique that we have incorporated involves the selection of only a group of targets while forming global hypotheses. The selection is based on the criterion that the target track should have a likelihood greater than a threshold. Targets that satisfy this criterion are referred to as *Confirmed* targets. The remaining targets are grouped as either *Intermediate, Tentative* or *Born* targets and each group has its own likelihood threshold. This form of grouping is termed Classification.

Clustering . As mentioned earlier, the computational complexity of the correlation and tracking algorithm arises mainly due to the vast number of global hypotheses that may be formed. The number of global hypotheses is an exponential function of the the number of tracks postulated. If target tracks lie within different regions of the surveillance volume such that no common reports are assigned to them, then obviously there is no need to form global hypotheses across these tracks. Clustering is a grouping of target tracks which avoids the formation of such global hypotheses.

3.2 Identification Algorithm

The multisensor target identification problem has also been studied extensively in the past. Methodologies generally differ with regard to how they model and account for the uncertainty in the sensor measurements [7]. We have adopted a Bayesian method since a Bayesian identification algorithm will be compatible with the Bayesian correlation and tracking algorithm discussed in subsection 3.1.

E-2C sensor data that can be used to identify targets are the PDS emitter type measurements, and the IFF response measurements. Correlation and Tracking algorithm outputs of target position and velocity profiles also contain information useful for target identification. Ideally, we would like to identify targets at a type level (Bear, F-14, etc.); however, this may not always be possible due to the fact that some of the sensor data cannot discern targets at the type level since it is the same for a group of targets. Examples of such sensor data are: target height information which can only identify targets as being surface or air; IFF responses from a target which can identify whether a target is friendly or neutral; and speed profile measurements of a target which can identify whether a target is a fighter or a bomber. Lacking sufficient sensor information, it may be more appropriate to identify a target at a group level higher (more abstracted) than the type level. The four levels at which the identification algorithm can identify targets are Type (Bear, Badger, ..), Class (Bomber, Fighter, ..), Nature (Friendly, Neutral, Hostile), and Category (Air, Surface).

To illustrate the identification algorithm, consider the case where there are N types of targets ($T_1, \ldots\ldots T_N$). Assume there exist a total of M types of emitters and the conditional probabilities $P(E_j \mid T_i)$ of observing emitter E_j when target T_i is present are known. Then if a certain observed emitter E_j is associated with a target with prior type probabilities $P(T_i)$, the type probabilities can be updated using Bayes rule as follows

$$P(T_i \mid E_j) = \frac{P(E_j \mid T_i)\, P(T_i)}{\sum_{k=1}^{N} P(E_j \mid T_k)\, P(T_k)} \tag{1}$$

Note that ambiguities in emitter type measurements are modeled by a measurement confusion matrix, and emitter type switchings on any target are modeled through a dynamic model. Details of the identification algorithm which utilizes these models are provided in [8].

Sensor information related to a higher level of target identity (Category, Nature, Class) is incorporated at a lower level based on the assumption that for each level of abstraction, the subsets of identities at the lower level are mutually exclusive and exhaustive. This procedure can be illustrated with the following example Assume that there exist L target classes ($L \le N$). The mutually exclusive assumption requires that each target type is assigned to a unique target class Further assume that speed profiles provide information only at the class level, i e., speed profiles of all target types within each target class are identical (e.g., all bombers have a speed profile of S_b, all fighters have a speed profile of S_f, etc) This assumption implies that

$$P(S_j \mid Ch(i), T_i) = P(S_j \mid Ch(i)), \quad T_i \in Ch(i), \tag{2}$$

which is the notion of conditional independence introduced by Pearl [9]. Then if a certain speed profile S_j is observed, the type probabilities of targets may be updated as follows:

$$P(T_i \mid S_j) = \frac{P(S_j \mid C_{h(i)})\, P(T_i)}{\sum_{k=1}^{N} P(S_j \mid C_{h(k)})\, P(T_k)} \tag{3}$$

Probabilities for target identification at higher levels (class, nature, and category) may be calculated by summing the probabilities of the target members belonging to that group at the lower level. For example, the probability that a target belongs to a class can be computed by summing the probabilities of all target types belonging to that class.

$$P(C_h) = \sum_{T_i \in C_h} P(T_i) \tag{4}$$

4. THREAT EVALUATION

A brute force methodology for evaluating the hostile target threat is to enumerate all tactical scenarios encountered by Naval battle groups and assign a threat condition to each of them However, the tactical scenarios that today's Naval battle groups will confront are expected to involve a vast number of targets Since the number of scenarios is a combinatorial function of the number of targets, their identities, and their relative positions and velocities, it is virtually impossible to enumerate all possible tactical scenarios Consequently, it will not be possible to use this brute force methodology Perhaps, this also explains why there is no universally accepted procedure for quantitatively or even qualitatively measuring the threat projected by hostile forces against a Naval battle group

One way of avoiding the combinatorial problem is to evaluate the threat hierarchically. Using a hierarchical approach, target threat characteristics are grouped and aggregated at several levels. Such an approach is attractive from the viewpoint that it is similar to the approach used by experienced E-2C surveillance operators, and consequently they can guide the selection of the number of levels in the hierarchy, along with the grouping and aggregation at each level. A hierarchical approach has the added advantage that explanations for threat conditions evaluated by the decision aid may be provided in the same hierarchical order; such explanations will be easier for the E-2C operator to comprehend.

Target correlation and tracking, and identification may be viewed as the first level of processing in the hierarchical threat evaluation procedure. Abstraction of the relevant target characteristics and the evaluation of the individual threat conditions represent the second level of processing in this hierarchical procedure. Finally, the evaluation of the collective threat condition for the battle group represents the third level. As we have indicated earlier, the definition of threat at both Level 2 and 3 is highly subjective and so a *precise* relationship between the threat and the target characteristics may not have any practical significance. Accordingly, we define these relationships based on the lines of reasoning used by experienced E-2C operators. For reasons discussed in Section 2, we classify the resulting algorithms as symbolic. We will discuss these symbolic algorithms for Situation Abstraction, Situation Assessment, and Threat Assessment in this section.

4.1 Situation Abstraction

The function of the Situation Abstraction submodule is to abstract target characteristics relevant for threat evaluation. These characteristics include not only those evaluated by the Level 1 Processing module such as the target's position, velocity, and identity, but also those associated with the target's warfare capabilities which are available in the target data base. The former characteristics can be parameterized and measured quantitatively (e.g., the hostile target has a speed of 250 knots), whereas the latter characteristics can be parameterized only qualitatively (e.g., the friendly target has a defensive capability of, say, *low*). For the purpose of evaluating the threat condition, however, it will be more convenient to transform even the quantitative characteristics to a qualitative form. The reason for this is that a precise mathematical relationship between the quantitative characteristics and the threat condition is not known. Even if a precise mathematical relationship can be postulated (and such a relationship is likely to be quite nonlinear), there is very little empirical data that can be used to validate such a relationship. The best we might be able to do would be to use predictions of this relationship made by an experienced E-2C operator. Furthermore, an E-2C operator would, most likely, parameterize all characteristics in a qualitative form.

The characteristics of a friendly and a hostile target which are relevant for threat evaluation are as follows.

1. The military/economic value of the friendly target;
2. The category (air or surface) of the friendly target,
3. The air-to-air firepower of the friendly target (if it is an air target), or the surface-to-air firepower of the friendly target (if it is a surface target);
4. The location of the friendly target,
5. The velocity of the friendly target;
6. The mode in which the friendly target is operating,
7. The military/economic value of the hostile target;
8. The air-to-air firepower of the hostile target;
9. The air-to-surface firepower of the hostile target;
10. The location of the hostile target;
11. The velocity of the hostile target;
12. The mode in which the hostile target is operating.

The firepower represents both the offensive and defensive components. The mode or intent of the hostile target represents the operating state or condition of the target and is evaluated based on the emitters used by the target and the position of the target relative to the other targets.

Since the complexity of the threat evaluation algorithm is exponentially related to the number of targets in the scenario, it is beneficial to group (cluster) targets that act in *unison*. The selection of targets that will be members of each cluster is based on their *interactive* characteristics. Specifically, members of each cluster should have small position and velocity separations, a similar nature (friendly or hostile), and the same category (air or surface).

Since we view a cluster as a number of targets acting in unison, we may aggregate the *intrinsic* characteristics of the individual targets. The intrinsic characteristics of a target include

1. military / economic value,
2. air-to-air firepower,
3. air-to-surface / surface-to-air firepower, and
4. air / surface category.

The aggregation of each of these intrinsic characteristics for the targets in a cluster yield the intrinsic characteristics for the cluster. For example, the aggregation of the military / economic values of each of the targets yields the military / economic value of the cluster.

The grouping and aggregation performed by the Situation Abstraction submodule serve two purposes. First, by grouping the targets into clusters and treating them as single entities in the subsequent analysis, the computational burden of this subsequent analysis is significantly reduced. Secondly, by aggregating the intrinsic characteristics for a single cluster, the synergistic effects of targets acting in unison (such as those of a *tactical unit*) can be captured. For example, by acting in unison, two fighter aircraft may possess a greater defensive and/or offensive capability than two aircraft acting independently.

4.2 Situation Assessment

The function of the Situation Assessment submodule is to evaluate the hostile target threat and the ability of the battle group to engage the hostile targets effectively on a one-on-one basis. To perform this function, the Situation Assessment submodule forms *cluster-pairs*. Each cluster-pair consists of a friendly and a hostile cluster. For each such cluster-pair, the relative characteristics between the friendly and hostile clusters are evaluated (this represents the aggregation step). These relative characteristics include:

1. Value-Ratio,
2. Relative Air-to-Air and Air-to-Surface capabilities,
3. Proximity,
4. Air / Surface category of friendly cluster, and
5. Mode of the hostile cluster.

An example of the evaluation of these relative characteristics is the evaluation of the proximity of the friendly and hostile clusters. Our definition of the proximity of two clusters accounts for not only the distance between the clusters, but also the time to engage (which takes into account the weapon ranges), and their closest point of approach. To understand why this is necessary, consider two cases where the clusters have the same distance separation. In the first case, the clusters are heading directly away from each other, while in the second the targets are headed directly towards one another. In these cases, the proximity would account for the fact that their 'separation' in the first case is smaller in a tactical sense, because their distance separation is decreasing, and if their trajectories remain unchanged, will become very small in the future. In addition to the distance separation and the relative velocities of the clusters, the range of the weapons carried by the members of the clusters is included in the evaluation of proximity. This is required because two clusters which are separated by a significant distance may still be within the firing range of each others' weapons. Finally, the closest point of approach is also included, providing a measure of how close the targets could become if their velocities remain unchanged. A similar aggregation of intrinsic characteristics (i.e., weapon range in this case) and interactive characteristics (i.e. physical separation, closest point of approach and closing velocity in this case) can be used to evaluate the other relative characteristics listed above.

The next step is to combine the relative characteristics to evaluate a threat condition for each cluster-pair. As we have pointed out above, it may not be possible to derive a precise mathematical formula, and even if we did it is not likely to be meaningful. Therefore, it is more appropriate to relate the relative characteristics to the threat conditions in a tabular form. The results of the threat evaluation will be qualitative since this is the type of information that E-2C operators will provide (i.e., an operator would evaluate the threat in a given scenario as *serious* rather than a precise number such as *8.357*). In addition, we note that it is unlikely that the information required for threat analysis will be provided for the complete list of combinations of the parameters described above. Instead, it is likely that some variables will become important only when other variables take on specific values. For example, the relative air-to-surface capabilities will be irrelevant in cases where both the friendly and hostile clusters are comprised totally of aircraft, while the relative value and military capabilities of two clusters may be unimportant if the clusters are a long distance apart and not headed towards each other.

Benefits provided by the aggregation of threat characteristics in the Situation Assesment submodule are the same as in the case of the Situation Abstraction submodule. Specifically, as we have indicated earlier, the relationship between the cluster characteristics and threat condition must be summarized in tabular form. If the table is to have a realistic number of entries, the number of independent variables must be restricted since the number of entries will increase exponentially with the number of independent variables. The aggregation performed by the Situation Assessment submodule reduces the number of independent variables used to determine the threat condition from 12 to 5 (for example the proximity evaluation aggregates four parameters into one), thereby making a tabular approach feasible. An additional benefit that results from aggregation is related to the generation of explanations for evaluated threat conditions: it is unlikely that a threat condition evaluated for a pair of clusters based on twelve characteristics will be quickly and easily understood by an E-2C operator. Once the aggregation is performed, however, the explanation can be provided by the assessment module in a hierarchical fashion. This will also enable the E-2C operator to query explanations at any desired level and detail.

4.3 Threat Assessment

The function of the Threat Assessment submodule is to evaluate collective hostile target threat, and the ability of the friendly targets to engage the hostile targets effectively. The collective hostile target threat represents the overall threat faced by each friendly cluster based on all hostile clusters which might interact with the friendly cluster. A cluster is defined to interact with another cluster if either of them imposes on the other a cluster-pair threat greater than a threshold.

Evaluation of the collective hostile target threat is based on cluster-pair threats and the time to engage evaluated by the Situation Assessment submodule. This evaluation procedure also includes the steps of grouping and aggregation of threat characteristics. The grouping step combines all clusters that influence one another's interactions to form what we call *super-clusters*. The threat condition for each friendly cluster belonging to a particular super-cluster is evaluated based solely on the clusters belonging to the same super-cluster. The aggregation step combines the threat parameters associated with cluster-pairs included in a super-cluster to evaluate the overall threat for each cluster. Since each cluster within a super-cluster has an influence on the remaining clusters, the overall threat for a friendly cluster is related to the threat parameters of all clusters included in the super-cluster.

The evaluation of the net threat to a friendly cluster based on the cluster-pair threats of the clusters included in the super-cluster may be summarized as follows:

Let

C denote the number of clusters in the super-cluster;

T_{ij} denote the threat imposed by cluster i on cluster j ($i \neq j$, $i \in C$, $j \in C$);

t_{ij} denote the time for cluster i to reach region from which it can engage cluster j;

OT_j denote the overall threat imposed on cluster j.

Step 1. For all T_{ij} > Threshold, evaluate t_{ij} as follows.

$$t_{ij} = 0, \quad \text{if } d_{ij} < r_i;$$
$$t_{ij} = \infty, \quad \text{if } d_{ij} > r_i \text{ and } CPA_{ij} > r_i;$$
$$t_{ij} = (d_{ij} - r_i)/rdot_{ij}, \quad \text{if } d_{ij} > r_i \text{ and } CPA_{ij} < r_i.$$

where

d_{ij} denotes the distance between cluster i and cluster j,

r_i denotes the weapon range of cluster i,

$rdot_{ij}$ denotes the relative range-rate of cluster i with respect to cluster j;

CPA_{ij} denotes the closest point of approach of cluster i with respect to cluster j.

Step 2. Order the t_{ij} in ascending order.

Step 3. Evaluate the overall threat level for each cluster as follows:

Step 3a. Initialize the *Commitment Level* for each cluster within the super-cluster j to zero ($CL_j = 0$)

Step 3b. Starting from the smallest t_{ij}, recursively update the Commitment Levels for cluster j based on the previous Commitment Level and the threat imposed by cluster i on cluster j as follows:

$$CL_j(new) = f(T_{ij}, CL_j(old), CL_i)$$

Step 4. Set the overall threat for all friendly clusters as $OT_j = CL_j$.

5. SIMULATION RESULTS

The Threat Evaluation algorithm discussed in Section 4 has been implemented in PROLOG on a Micro Vax II Graphics workstation. We refer to this implementation as the Threat Evaluation Module (TEM). The preliminary implementation of TEM is deterministic in that we assume that target tracks and identification are known with probability one. This allows us to 'short circuit' the correlation and tracking, and the identification algorithms (discussed briefly in Section 3), and demonstrate the performance of the threat evaluation algorithm. We plan to implement the probabilistic version of TEM using the same hierarchical approach. This probabilistic version will be integrated with the correlation and tracking algorithms in the prototype E-2C decision aid.

The deterministic implementation of TEM required 31 PROLOG rules and 89 Prolog facts. In addition, four routines which carry out numerical computations are coded in FORTRAN. We predict that for the probabilistic version of TEM, the number of facts will increase to about 250, and the number of rules to about 50.

We have developed an E-2C simulation package to drive the E-2C tactical decision aid. This simulation package, which we refer to as REDGEN (REalistic Data GENerator), has the capability to generate E-2C sensor reports for fairly realistic scenarios that might be encountered by the E-2C. We have used REDGEN to simulate the following scenario to demonstrate the important capabilities of the Threat Evaluation algorithm.

The scenario comprises several friendly and hostile targets. The friendly targets forming the battle group consists of a single Aircraft Carrier (CV), two Destroyers (DD), six Combat Air Patrol (CAP) aircraft (F-14s), and the E-2C. The hostile targets consist of three bombers (Badgers) and six escort fighters (MiG-27s). Figure 3 shows the initial location and a track history for all targets for the duration of the scenario. For the sake of clarity in the figure, we have labeled clusters of targets as opposed to the individual targets; however, we have shown the individual target tracks for the entire duration of the scenario. The racetrack trajectory at the center of the figure represents the trajectory of the E-2C. The concentric circles around the E-2C are spaced 50 nautical miles and are used to indicate distance relative to the E-2C.

The scenario progresses through four stages corresponding to different threat levels imposed by the hostile targets. Threats evaluated by the decision aid for conditions that

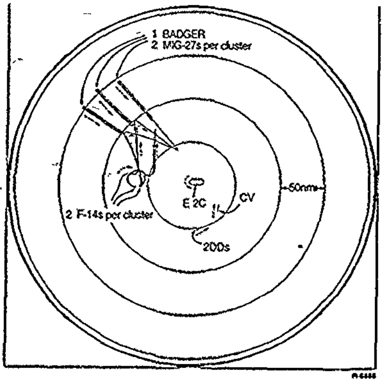


Figure 3. Overall Scenario.

are representative of these four stages are shown in Figs 4 - 7. In each of these figures, the PDS reports are shown using radial lines in the outer band of the figure, the target tracks for the recent past are shown by short line segments, and the nature of the target clusters (friendly, hostile,...) is represented by simple geometric symbols surrounding the target cluster. The meaning of the symbol is indicated in the Target Nature legend. Finally, the current threat posed against a target cluster is shown by a number placed next to the geometric symbol. The interpretation of this number is indicated in the Threat legend. (On the VR-290 color graphics terminal used to display the results of the decision aid, the threat levels are indicated using a color code).

Figure 4 shows the first stage of the scenario. The hostile targets are assumed to have entered the radar coverage region of the E-2C and are shown headed towards the Aircraft Carrier and the Destroyers (surface targets). CAP aircraft are shown maintaining station since, at this stage, they have not yet been assigned to intercept the hostile targets. The threat evaluation algorithm evaluates a *high* threat to the surface targets, a *moderate* threat to the CAP aircraft, and only a *low* threat to the hostile targets. (Note that the threat posed against the hostile aircraft is a measure of the counter-threat imposed by the friendly targets.)

The second stage of the scenario is shown in Fig 5. The CAP aircraft have been assigned to the hostile targets and are headed towards the hostile targets. The threat evaluation algorithm reduces the threat imposed on the surface targets to *none* based on the counter-threat provided by the CAP aircraft. Threat to the CAP aircraft themselves and the hostile targets have both been increased (as shown in the figure) based on the predicted air engagement.

Figure 6 shows the third stage of the scenario which follows the first set of air engagements. All but two of the CAP aircraft and one of the hostile aircraft are assumed destroyed during the engagement. The threat evaluation algorithm recognizes that one of the hostile targets has temporarily succeeded in avoiding the counter-threat provided by the CAP aircraft. Accordingly, it sets the threat posed to the surface targets to an increased value. In the final stage of the scenario, the surviving CAP aircraft have been reassigned and have managed to get on

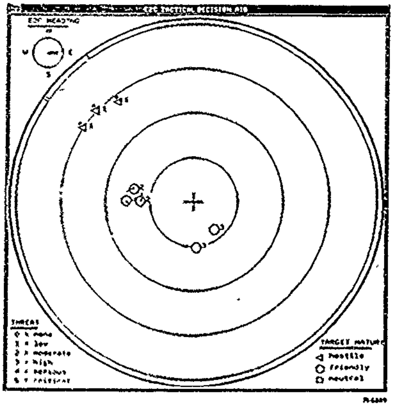


Figure 4. Stage 1 - Attacking Hostile Targets, CAP Unassigned.

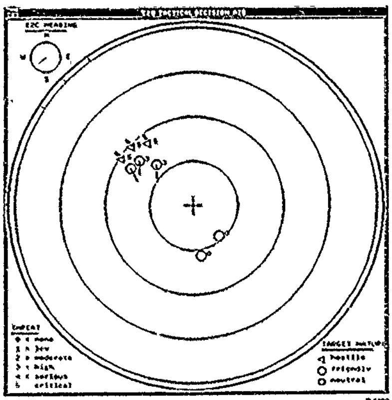


Figure 5. Stage 2 - CAP Patrol Assigned to Hostile Targets.

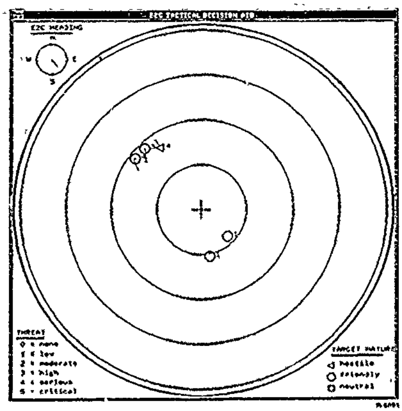


Figure 6. Stage 3 - One Hostile Target Survives Engagement.

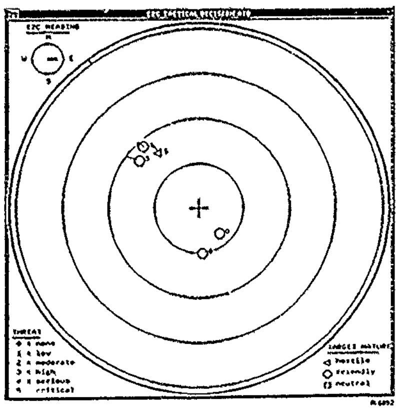


Figure 7. CAP Patrol Reassigned to Surviving Hostile Target.

the tail of the surviving hostile aircraft. Accordingly, the threat evaluation algorithm decreases the threat to the surface targets. However, as in stage 2 of the scenario, the threat against the CAP aircraft and the hostile targets are increased based on the predicted second engagement These threat levels are indicated in Fig. 7.

It can be seen that the threat evaluations made by TEM are consistent with those that are actually simulated. For the simple scenario that has been simulated, it may have been possible for an E-2C operator to predict the same set of threat conditions. However, for a more complex scenario involving hundreds of targets, it would be difficult for an E-2C operator to predict all potential engagements and the associated threats. In such situations, the threat evaluation algorithm will prove to be a significant aid to the E-2C operator.

6. SUMMARY

We have defined a processing architecture for an airborne surveillance system which simultaneously addresses the problems of multitarget tracking and threat evaluation. We have shown that the different nature of the two problems can be addressed by a processing architecture which supports both numerical and symbolic processing. Numerical processing algorithms are based on the mathematics of estimation theory and are applicable to the problem of multisensor multitarget target correlation and tracking, and identification. Symbolic processing algorithms are based on the domain knowledge of the E-2C surveillance mission and are applicable to the problem of threat evaluation.

A description of both the numerical and the symbolic algorithms has been provided. The numerical algorithms have been implemented and demonstrated in FORTRAN. Simulation results obtained with a prototype of the threat evaluation algorithm implemented in PROLOG have also been provided. These results have shown that threat evaluations made by the threat evaluation algorithm are consistent with those actually simulated.

In our ongoing effort, we plan to incorporate a more realistic data base for the sensors on board the E-2C surveillance aircraft and the targets which it has to track. We also plan to integrate both the numerical and symbolic processing algorithms in the prototype of the airborne surveillance decision aid. Finally, we plan to evaluate the computational requirements of the decision aid and indicate how it might be incorporated into an operational airborne surveillance system.

REFERENCES

1. Data Fusion Research Survey Report, 1988 Data Fusion Symposium, John Hopkins University Applied Physics Laboratory, Laurel, Maryland, 17-19 May 1988.

2. Blackman, S.S., Multiple Target Tracking with Radar Applications, Artech House, 1986.

3. Bar-Shalom, Y., 'Tracking Methods in a Multiobject Environment,' IEEE Transactions on Automatic Control, Vol. AC-23, August 1978, pp 618-626.

4. Reid, D.B., 'An Algorithm for Tracking Multiple Targets', IEEE Transactions on Automatic Control, Vol. AC-24, December 1979, pp. 843-854.

5. Kurien, T., and R.B. Washburn, 'Multiobject Tracking Using Passive Sensors,' Proceedings of the 1985 American Control Conference, Boston, MA, June 19-21, 1985, pp. 1032-1038.

6. Liggins, M.E., M.A. Gerber, S.W. Gully, T. Kurien, R.B. Washburn, and M.A. Weiner, 'Multispectral Multisensor Fusion Development for Enhanced Target Detection and Tracking,' 1988 Tri-Service Data Fusion Symposium, John Hopkins University Applied Physics Laboratory, Laurel, Maryland, May 17-19, 1988.

7. Stephanou, H.E., and A.P. Sage, 'Perspectives on Imperfect Information Processing,' IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-17, October 1987, pp. 780-798.

8. Caromicoli, A., and T. Kurien, 'Multisensor Target Identification in Airborne Surveillance,' ALPHATECH Technical Memorandum TM-299, ALPHATECH Inc., Burlington, MA, June 1988.

9. Pearl, J., 'On Evidential Reasoning in a Hierarchy of Hypotheses,' Artificial Intelligence 28, 1986, pp. 9-15.

AN ALGORITHMIC APPROACH TO MULTI-SOURCE IDENTIFICATION

Stephen L. Pendergast

HUGHES AIRCRAFT COMPANY, INC.

ABSTRACT

The purpose of this paper is to provide an overview of real-time automated identification processing algorithms as they will be implemented in the Advanced Combat Direction System (ACDS) Block 1 as described in ref[1]. The taxonomy, system architecture, and man-machine context in which ACDS will operate each strongly affected the overall design and algorithm trade-off decisions.

ACDS BLOCK 1 IDENTIFICATION REQUIREMENTS

ACDS Block 1 will replace NTDS on carriers and non-AEGIS cruisers as the data processing software that drives Combat Information Center(CIC) displays and generates data link messages. ACDS and AEGIS C&D-ADS are assigned the function of organic tracking and identification by ref[2]. Because ACDS will go to sea in the early 1990's, and will be the first system to automate the fusion of multi-source information into the tactical environment, it can be considered as the operational state-of-the-art for automated identification and classification processing of tracks.

ACDS is required combine real-time and non-real time track data to identify for immediate display and threat processing all tracks, cooperative and non-cooperative, in the Navy dispersed battle force surveillance volume using all available data sources in standard Navy computers (AN/UYK-43) using the Navy standard high order computer language (CMS-2) within a tactical real-time operational frame These requirements make efficient use of computer memory and processing mandatory and thus preclude some artificial intelligence techniques.

The ensuing track identification provides a level of detail that is useful for command decisions and is consistent with STANAG 1241 (ref [3]), the basis for the TADIL J taxonomy as implicitly defined in the fixed format messages of the JTIDP-TE, volume II, and volume V, (ref [4,5]). Data link protocols for ID precedence and difference resolution as defined in ref [6] are observed. The operator's principalrole is as arbitrator when conflict situations arise.

IDENTIFICATION SYSTEM OVERVIEW

The ACDS design shown in figure 1 provides for a multi-level identification process. At the initial fusion level, similar source integration (SSI) functions combine sensor data received from sets of independent sensor data processors with similar characteristics, and derive classification alternatives and confidences. At the higher level, after correlation by a Dissimilar Source Integration (DSI) function, the Multi-Source Identification (MSID) function combines classification and confidence data to arrive at fused alternatives and a "best" identification for the track. The two-tiered integration process has available to it a variety of stored data including: a priori presence probabilities, libraries, speed/altitude relationships, IFF code look-ups, and operator-modifiable doctrine statements.

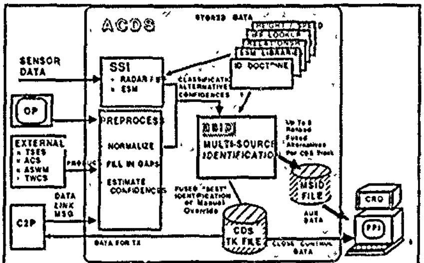


Figure 1. Identification Process Overview

ALGORITHM DEVELOPMENT

ACDS will be combining data from multiple sensors of different types, locally detected or communicated by tactical links (TADILS), to form an overall identification estimate of each track, together with an estimate of the identification quality. A standardized identification taxonomy and quality descriptor for source information, and a method of combining the information and ranking alternatives in a multi-level identification description is needed. The algorithms and analyses developed in this project show an approach to solving these problems.

The data from multiple sources and multiple sensors of different types on multiple platforms communicated by tactical distribution links (TADILS) need to be combined to form an overall identification estimate, together with an estimate of the identification quality. Different sensors require different libraries of object characteristics and distributions in order to determine possibilities. Furthermore, any library will be incomplete, and must allow for some level of ignorance. We wish to separate the sensor error characteristics from the library object characteristics. A standardized quality descriptor has to be developed so that information can be interchanged between systems. A method of combining the information and ranking alternatives in a multi-level identification description is needed.

SOURCE CONFIDENCE COMPUTATION.

A method of developing lists of possibilities and confidences which permits partitioning of raw sensor data and the libraries necessary to interpret it into specialized systems, while providing information which can be objectively evaluated and combined with information from other systems is necessary. Development of a computation algorithm which allows inclusion of sensor characteristics, alternative a-prioris, alternative parametric distributions, and library degree of completeness will be described. The ability of the algorithm to estimate relative degrees of confidence for different alternatives with widely varying a-priori alternative confidences, sensor accuracies, and inherent library ambiguities and known gaps in a-priori knowledge will be demonstrated.

An analysis will be described which, using ESM data as an example, demonstrated the ability of the algorithm to estimate relative degree of confidence in different alternatives in a library with widely varying a-priori alternative confidences, sensor accuracies, and inherent library ambiguities and a-priori library limitations and ignorance. Results of multiple cases of varying measurements with varying accuracies using different parameters and two different parametric libraries, and different sets of a-priori alternative confidences, each with multiple alternatives will be described.

The example libraries were constructed based on parameters in typical ESM operational geographically tailored libraries used in the AN/SLQ-17 to show different types of ambiguities and overlap, including wide and narrow regions of operation, and alternatives with partial overlap, and total containment. Measurements were selected to show precise and non-precise measurements with good fit and not-so-good fit to only one or several ambiguous alternatives, as well as good fit to none but the unknown alternative. Two sets of a-priori values were selected to show the effect of uniform and widely varying alternative a-priori probabilities.

The examples demonstrated the confidence computation algorithm's capability of estimating high confidence values for precision measurements with limited or no ambiguity in the library. Low accuracy measurements could still result in high confidence classification if the limit of accuracy fit within the parameter limits, and there were no ambiguous specific types. For low accuracy measurements with large regions of probability falling outside the library alternative class parametric limits, the confidence was allocated to the unknown alternative, as opposed to being force fit among the library alternatives as the classic Bayes technique would do. A further characteristic of the algorithm demonstrated by varying the a-priori alternative classification probabilities was that measurements matching a particular alternative resulted in high confidence attributed to that alternative, even though the a-priori probability was very small.

ESM Classification Algorithm Features Desired

Any algorithm to develop classifications and confidences from Electronic Support Measures (ESM) should account for sensor noise. We would expect precise sensor measurements of an emitter with unique characteristics to have less ambiguities and hence a higher confidence than measurements with a large variance. Conversely, precise measurements of emitters with widely varying operating points, and a large overlap of multiple similar emitters should result in lower confidence. We would further desire any ESM classifier algorithm to use knowledge of geographic area intelligence and sensor field of view to prune possibilities from the list of alternatives before any confidences are computed.

Another important aspect required of a realistic ESM classification algorithm is the need to account for incompleteness of our knowledge so that we assign more confidence to "unknown" when the library is sparse. Furthermore, when the emission characteristics are not unique, we need to collapse ambiguities upward when a lower level of detail is not justified. The algorithm should preserve all reasonable alternatives exceeding a specified level of probability, while at the same time thresholding out those which are impossible (are less than the threshold) and can be ruled out

ESM SSI-Fuses Force Passive Electromagnetic Sensor Classification Data.

The ESM SSI correlates and fuses track parametric and classification data from local, remote, & non-organic passive em sensors. Its geographic library relates measurements to specific types through emitter and mode linkage. It uses the intersection of multiple emitter possibility sets to successively reduce classification alternatives as data is integrated. It uses a modified Bayes with unknown to compute confidences to be assigned to possible alternatives.

ESM Measurement Sequence

We assume that the ESM sensor measures modulation parameters which define a sequence of measurement vectors $\bar{z}_k$ with associated covariance M_k of a detected electromagnetic emission which may include:

(1) Azimuth Bearing, (continuous, degrees)
(2) Elevation Angle, (continuous, degrees)
(3) Pulse Repetition Frequency (Continuous, Hz)
(4) Frequency (Continuous, MHz)
(5) Pulse Width (Continuous, μsec)
(6) Scan Period (of lowest scan modulation Rate, continuous, sec)
(7) Scan Frequency (of highest scan modulation , continuous, Hz)
(8) Scan Type (discrete description)

The statistics of the discrete measurements may be described with a probability matrix P_k with elements p_{ij} which describe the probability of i being estimated when j was present. We further assume that the statistics of the measurements of the continuous variables can be characterized by a multivariate Gaussian density function (ref 7), with a density conditioned on the measurement mean and covariance estimate given as:

$$p(z|\bar{z}_k, M_k) = \frac{1}{(2\pi)^{n/2}|M|^{1/2}} e^{-(z-\bar{z}_k)^t M_k^{-1}(z-\bar{z}_k)/2} \quad \ldots (1)$$

If the n individual parameters of measurement k are uncorrelated, then the multivariate density function becomes a product of univariate density functions:

$$p(z|\bar{z}_k, M_k) = \prod_{i=1}^{n} \frac{1}{\sqrt{2\pi m_{iik}}} e^{-(z-\bar{z}_{ik})^2/(2\, m_{iik}^2)} \quad \ldots\ldots\ldots (2)$$

ESM Library Characterization.

The objective of the ESM library is to describe the tactical frame of discernment in terms of the objects to be identified and their a-priori presence in the tactical surveillance volume, the electromagnetic emitters associated with them, the emitter modes and probabilities of operation, and the parameter distributions associated with the various modes of operation. With this information, the ESM measurements can be inverted to estimate the probability of each element in the tactical identification taxonomy having caused the observed data. Thus we need a relatively extensive, complex data base to interpret the measurements, including the following tables:

a. Specific Type a-priori presence potentials in the sensor surveillance volume, $P(\text{type}_p|\text{sensor}_q \text{ \& area})$.
b. Emitters associated with each Specific Type and Type Mod, $P(\text{emitter}_n|\text{type}_p)$.
c. Modes of operation and their a-priori likelihood associated with each emitter, $P(\text{mode}_m|\text{emitter}_n)$.
d. Mode parameter distributions for all the measurement parameters for each emitter mode, $p(z|\text{mode}_{mn})$.

The multiple libraries, linked together by pointers and indices create for each Specific Type and Type mod a complex, multidimensional electromagnetic fingerprint, simplified as shown in figure 5, where the three axes are frequency, pulse repetition frequency, and pulse width. Mathematical linking the libraries together, the specific type T_p conditioned density function of z is defined as:

$$p(z|T_p) = p(z|\text{mode}_{mn}) \cdot P(\text{mode}_{mn}|\text{emitter}_n) \cdot P(\text{emitter}_n|\text{Type}_p) \quad \ldots\ldots\ldots\ldots (3)$$

Building these libraries is a primary job of the intelligence community, which basically defines the identification taxonomy in terms of things it can detect. Since each sensor type has a different discrimination capability, the ESM library set of specific types tends to differ from the Radar, Visual, Acoustic, COMINT, and HUMINT sets. But from the multi-source identification point of view, we need to drive all sensor libraries to a common taxonomy.

In the naval operational context, where battle forces consisting of air, surface, subsurface, and amphibious category platforms can be moved to any point on 75% of the earth's surface, the basic set of specific types and a-priori presence probabilities need to be re-tailored to the current operating zone. Furthermore, the limitations of different sensor types and platforms needs to be considered for range and altitude limits defined by equipment sensitivity, horizon, and field of view.

In our system, these libraries are included for selected specific types of tactical significance, with additional entries for "UNKNOWN". UNKNOWN has a distribution and probabilities assigned to it uniform over all fields of view and parametric limits and discretes. The a-priori likelihood assigned to the UNKNOWN specific type is a function of the expected objects in the operational environment which are not contained in the set of defined specific types.

Emitter Mode Parametric distribution functions are described by upper and lower limits for each parameter, and probability values assigned to multiple steps within the interval, summing to one, with the density defined as uniform in each interval, the probability divided by the interval size. as illustrated in Figure 6.

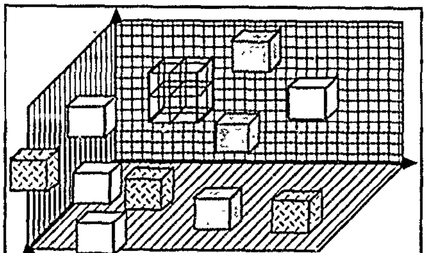


Figure 6. Platform ESM Signature.

Possibility Selection

Given a measurement vector z_k and its characterization with mean and covariance, as given in equation (1), and a set of a-priori type-mod presence potentials, and platform ESM signatures as defined in equation (3), we first define a possibility list about the measurement by defining a threshold parameter error window defined by the using the measurement mean and covariance and a specified likelihood of missing a possible alternative, say 0.001. An indexed search using bit mapped techniques provides rapid elimination of alternatives. As each successive measurement is correlated with a particular track, we only have to evaluate the previous possible list to trim it down. This process rapidly reduces the number of possibles as shown in table I, where four emissions associated with a track uniquely identify it. Even fewer emissions may suffice to select platform and category. Raw emission data must be retained through this process, so that if emissions are decorrelated due to spatial separation or elimination of all possibles, the process can be redone.

Table I. Multiple Correlated Emitters Intersect to Reduce Possibles.

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	A	A	A	A	A	A	A	A
18	Category		Air Specific Types																										Air Platforms						Air
19	Platform		A	A	A	A	A	B	B	B	B	B	B	B	B	C	C	C	C	C	C	F	F	F	F	F	F	F	A	B	C	D	E	F	
20	Spec Type		a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z							
21	Emission	1	0	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	0	1	1	3	4	3	0	0	3	17
22		2	0	0	1	0	0	0	0	1	0	1	1	0	1	1	0	1	1	1	1	1	0	0	0	1	0	1	1	4	4	0	0	3	12
23		3	1	0	1	0	0	0	0	1	0	0	1	0	1	0	1	0	0	1	0	0	0	1	1	1	1	1	2	3	2	0	0	5	12
24		4	1	0	1	1	1	1	1	1	0	1	0	1	0	0	0	1	0	0	1	1	1	1	0	0	0	0	4	5	2	0	0	3	14
25	Cumulative Intersection																																		
26		1	0	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	0	1	1	3	4	3	0	0	3	1
27		2	0	0	1	0	0	0	0	0	0	1	1	0	1	1	0	0	0	1	0	1	0	0	0	0	0	1	1	3	2	0	0	2	8
28		3	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0	0	0	1	0	0	0	0	0	0	0	1	1	2	1	0	0	1	5
29		4	0	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
30	Cumulative Union																																		
31		1	0	1	1	0	1	0	0	0	0	1	1	1	1	1	0	0	1	1	0	1	0	0	0	0	1	1	3	4	3	0	0	3	1
32		2	0	1	1	0	1	0	0	1	0	1	1	1	1	1	0	1	1	1	1	1	0	0	0	1	1	1	3	5	5	0	0	4	17
33		3	1	1	1	0	1	0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	4	5	6	0	0	6	2
34		4	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	5	7	6	[illegible]	0	7	22

	AK	AL	A	A	A	A	A	A	A	A	A	A	A	A	A	A	B	B	B	B	B	B	BG	BH
18	Category		Surface Specific Types														Surface Platfo						Sur	
19	Platform		A	A	A	B	C	C	D	D	D	D	D	F	F	F	A	B	C	D	E	F		
20	Spec Type		a	b	c	d	e	f	g	h	i	j	k	l	m	n								
21	Emission	1	1	0	1	1	1	0	0	0	1	1	0	0	1	1	2	1	1	2	0	2	8	21
22		2	0	0	0	1	0	1	1	1	0	1	1	0	1	0	0	1	1	4	0	1	7	19
23		3	1	0	0	1	0	1	1	1	0	0	0	1	0	0	1	1	1	2	0	1	6	18
24		4	0	0	1	0	1	1	1	1	1	0	1	0	1	0	1	0	2	4	0	1	8	22
25	Cumulative Intersection																							
26		1	1	0	1	1	1	0	0	0	1	1	0	0	1	1	2	1	1	2	0	2	8	21
27		2	0	0	0	1	0	0	0	0	0	1	0	0	1	0	0	1	0	1	0	1	3	11
28		3	0	0	0	1	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	6
29		4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
30	Cumulative Union																							
31		1	1	0	1	1	1	0	0	0	1	1	0	0	1	1	2	1	1	2	0	2	8	21
32		2	1	0	1	1	1	1	1	1	1	1	1	0	1	1	2	1	2	5	0	2	12	29
33		3	1	0	1	1	1	1	1	1	1	1	1	1	1	1	2	1	4	5	0	3	13	34
34		4	1	0	1	1	1	1	1	1	1	1	1	1	1	1	2	1	2	5	0	3	13	35

Modified Bayesian Inversion

Given a set of possible specific type mods created by the search described above, we then wish to estimate the confidence to be assigned to each possible alternative. This is done by computing a conditional probability function defined over the possible library alternative specific type mods, and integrating it over the measurement probability density function to yield the probability vector conditioned on the measurement. This is shown graphically in Figure 7 below. The conditional probability function for specific type mod T_q with a set of N possibles, to be detected by sensor S_q in area A is computed mathematically using Bayes law as:

$$P(T_q|z\&S_q\&A) = \frac{P(z|T_q)\cdot P(T_q|S_q\&A)}{\sum_{q=1}^{N} P(z|T_q)\cdot P(T_q|S_q\&A)} \quad ......(4)$$

Note that the conditional probability function is conditional on the sensor, area, and specific type set, not the actual measurement $\bar{z}_k$, or covariance, hence it could theoretically be computed in advance. The probability conditioned on the first measurement, k=1 can be computed by integrating the conditional probability function over the measurement conditional density function:

$$P(T_i|z\&S_i\&A\&\bar{z}_1\&M_1) = \int_{-\infty}^{\infty} P(T_i|z\&S_i\&A)\,p(z|\bar{z}_1,M_1)\,dz \qquad (5)$$

This process may be iterated in a process similar to a Kalman filter as each successive independent measurement k is received:

$$P(T_q|z\&S_p\&A\&\bar{z}_k\&M_k) = \int_{-\infty}^{\infty} P(T_q|z\&S_p\&A\&\bar{z}_{k-1}\&M_{k-1})\,p(z|\bar{z}_k,M_k)\,dz. \qquad (6)$$

An alternative approach would be to aggregate successive measurements into a larger observation vector, and the original possibility set binary search and repeat the Bayes inversion. This approach must be used if the measurements are not independent.

Some properties of this process match the desired properties enumerated above. For example, with an UNKNOWN value (denoted as T_0) defined for all z, if the measurement density function has finite area which does not match the parametric limits assigned to the library emitter modes, this area will be assigned to T_0. If an emitter mode has limited parametric range, it will require a measurement with covariance significantly smaller than the operating range to get a high confidence result. If emitters have a common range of operation, the confidence of a measurement in the range of common operation will be split between them proportionately to their individual total ranges of operation.

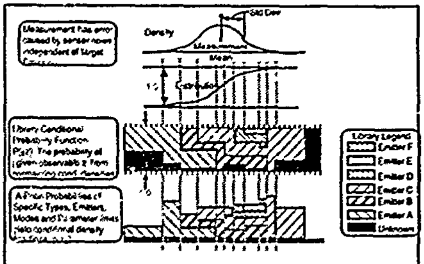


Figure 7.Modified Bayesian Inversion

This Bayes confidence computation algorithm could be used at a central node to process all raw sensor information through a global characterization process. This would probably be the optimum method so far as accuracy of the identification is concerned. However it would not permit compartmentalization of the parametric libraries, and the algorithm would not be readily broken up into multiple processors. As more and more sensor data was gathered, the amount of computation required at the central node would grow exponentially. Thus we wish to look at alternative methods which allow partitioning of the identification process, without suffering a large amount of degradation from the global Bayesian combining.

CONFIDENCE COMBINING ANALYSIS

A method of combining possible alternative and confidence information from different sources which takes into account unique characteristics of different sources in terms of ignorance, partial knowledge, and detectability constraints to fuse the information and determine source conflicts and ambiguities is necessary.

Demonstrations of the use of the Dempster-Shafer evidence combining rules will be described and compared with Bayes and another evidence combining algorithm, the diversity combining algorithm derived from signal processing. The confidence combining algorithm analysis demonstrated the capability of the Dempster-Shafer combining algorithm to approach the performance of global Bayesian combining, while allowing source migratability and library compartmentation, as well as including source ignorance(unknown,no statement) and the intersection of evidence in the combining process.

ESM SSI source confidences and Radar/SSI source confidences were computed by the modified Bayes source confidence algorithm described above using fictional ESM emission parametric and Target Kinematic/Altitude characteristic libraries with ambiguity characteristics similar to those expected in a tactical system. These were combined using the Dempster-Shafer algorithm, and as an alternative, the Diversity Combining algorithm, and compared with the results obtained with a global Bayes confidence computation algorithm (having access to all raw measurements and a global parametric library).

Performance metrics used in the comparison included:

a. Mean square confidence difference over the top five alternatives computed by each of the combining algorithms versus the global Bayes algorithm.

b. Mean square ordinal (rank) difference over the top five alternatives computed by each of the combining algorithms compared to the global Bayes combining algorithm.

Dempster-Shafer Algorithm.

The Dempster-Shafer algorithm for evidence combining was originally developed by Dempster (in refs [8,9,10]) and Shafer (in refs [11,12]). Application of it to tactical data fusion was suggested by Dillard (ref [13]) using the special case where knowledge sources assign probability masses only to the propositions T_i and to uncertainty Θ. This is the version of the combining law we use here, where the evidence assigned to T_o the Unknown, is applied as uncertainty. This as opposed to the complete implementation of Dempster's law of combination (Shafer Theorem 3.1) over the set of all non empty subsets A such that $2^{\Theta} \supset A$. the combination of two probability assignments m_1 and m_2 is given by:

$$m(A) = \frac{\sum\limits_{i;j\ A_i \cap B_j = A}^{2^{\Theta}} m_1(A_i)\, m_2(B_j)}{1 - \sum\limits_{i;j\ A_i \cap B_j = \emptyset}^{2^{\Theta}} m_1(A_i)\, m_2(B_j)} \quad \ldots\ldots\ldots\ldots\ldots (7)$$

With the number of alternatives to be considered, the set of subsets rapidly grows too big to contemplate. Thus the rule of combination for specific type mod t_n confidence derived from evidence z and y becomes:

$$m(t_n|z\&y) = \frac{m(t_n|y)m(t_n|z)+m(t_o|y)m(t_n|z)+m(t_n|y)m(t_o|z)}{\sum\limits_{i=1}^{N} m(t_n|y)m(t_n|z)+m(t_o|z)+m(t_o|y)-m(t_o|z)m(t_o|y)} \quad \ldots\ldots\ldots\ldots (8)$$

Diversity Combining Algorithm

An alternative formulation for combining evidence was proposed based on the algorithm for diversity combining of multiple channels used in communications. In this algorithm, confidence functions from K sources were converted to a likelihood type function, summed across the sources, and then renormalized, using the formula:

$$m(T_k|z_k, k=1..n) = \frac{\sum\limits_{k=1}^{K} \frac{m(T_i|z_k)}{1-m(T_i|z_k)}}{\sum\limits_{n=1}^{N} \sum\limits_{k=1}^{K} \frac{m(T_n|z_k)}{1-m(T_n|z_k)}} \quad \ldots\ldots (9)$$

This algorithm has several attractive features. For example, high confidence values will not be degraded when combined. No alternatives will be eliminated in the combining process.

Simple Combining Example.

In order to demonstrate the comparative performance of the different proposed algorithms, multiple examples of typical expected tactical measurements were used to generate two confidence vectors, and the results combined. Here we will describe a simple two variable measurement where each variable is used to compute a confidence vector and the results combined. The input measurement of variable v by source 2 and w by source one is shown in Figure 8 below. The different type A-E characteristics in v-w space are shown as shaded rectangles. The measurement and its covariance is shown by the ellipse.

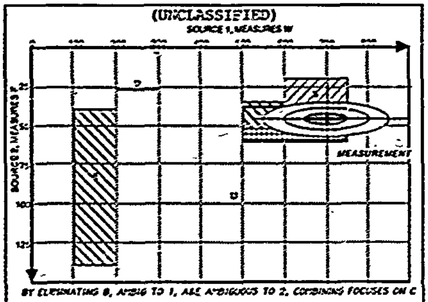


Figure 8. Two Variable, Two Source Input Example

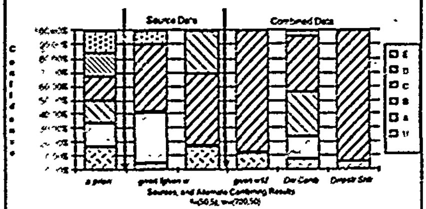


Figure 9. Example Evidence Combining Results

The confidence vectors generated by the two sources, and the results of combining the measurements using the Bayes inversion, and combining the source confidences using the special case Dempster-Shafer and Diversity Combining algorithms are shown in Figure 9 below. The leftmost column shows the a-priori probabilities of the different alternatives, in this case, uniform. The second two columns show the results of each source operating on its individual measurement using the Bayes algorithm. The last three columns show the results of combining the two sources using Global Bayes inversion of the raw sensor measurements, Diversity Combining, and Dempster-Shafer combining respectively. Notice that Dempster-Shafer and Bayes both focus on alternative C.

Detailed Performance Results.

Results for all of the examples were cumulated for the two confidence combining algorithms. The results showed that the Dempster-Shafer algorithm was closer to the global Bayes algorithm in 90% of the examples, with a mean square ordinal difference of less than 0.4 (one flip of adjacent alternatives), compared to the diversity combining ordinal difference of 1.4. Confidence accuracy was also better, with the Dempster-Shafer having an rms confidence difference of ≤9% compared to the Diversity Combining algorithm rms confidence difference of 20%.

Qualitative evaluation of the results also showed the Dempster-Shafer algorithm to be superior, in that it tended to eliminate alternatives which were obviously impossible when source measurements were combined, while the diversity combining algorithm formed a union set of all proposed alternatives, never eliminating anything. The area in which both algorithms seemed to be subject to error was in under-estimation of the confidence which should be attributed to the "Unknown" alternative.

ALTERNATIVE RANKING ANALYSIS.

Once the alternative confidences are computed, and mapped up to the multiple levels, a method of ranking alternative set descriptors of possibilities is needed which takes into account the information conveyed by a description, together with the confidence in a particular description when multiple taxonomic levels of description need to be compared against each other. Development of an entropy weighted ranking algorithm for comparing different level of detail and confidence identification alternatives will be described.

The alternative ranking algorithm analysis demonstrated the capability of the ranking algorithm to select the best identification alternative over several example data sets considering a multi-level identification taxonomy. The algorithm was implemented with several examples using a spreadsheet to compute and sort the alternatives.

Example Alternative Ranking Problem

An example showing the ambiguities which make a multilevel alternative ranking sort necessary is shown in table II, where sensor data is ambiguous as to Category as well as Platform and Specific Type.

Table II. Ambiguous Alternative Ranking Example.

CAT		PLATFORM		SP TYPE MOD	DIS CONF
AIR		FIGHTER		FITTER C	0.011
AIR		FIGHTER		TOMCAT F14A	0.032
AIR		FIGHTER		FISHBED J	0.011
AIR		FIGHTER		MIRAGE 3E	0.032
AIR		FIGHTER	0.122	UNKNOWN	0.036
AIR		BOMBER		BEAR	0.098
AIR		BOMBER	0.160	UNKNOWN	0.062
AIR		HELO		HORMONE	0.063
AIR		HELO	0.080	UNKNOWN	0.017
AIR	0 402	UNKNOWN	0.040	UNKNOWN	0.040
SURF		CRUISER		KRESTA 2	0.144
SURF		CRUISER		KRESTA 1	0.048
SURF		CRUISER	0.200	UNKNOWN	0 008
SURF	0.250	UNKNOWN	0.050	UNKNOWN	0.050
LAND		MISSILE SITE		EXOCET MM40	0 152
LAND		MISSILE SITE		SEPAL SSC1B	0 019
LAND		MISSILE SITE	0.190	UNKNOWN	0 019
LAND	0.200	UNKNOWN	0.010	UNKNOWN	0 010
SUBM		SSGN		ECHO 2	0 074
SUBM		SSGN	0 095	UNKNOWN	0 021
SUBM	0.100	UNKNOWN	0.005	UNKNOWN	0.005
UNKN	0.050	UNKNOWN	0.050	UNKNOWN	0.050

As shown below in table III, summing confidences at different levels and ranking results in different, incompatible results. The Entropy weighted rank scoring algorithm allows alternatives at different levels to be compared, and the level of detail justified by the data to be ranked highest.

The several ways sorting the alternatives shown in table III above do not result in a top 5 ranking which gives a clear measure of the best alternatives. As shown in graphically in the pie charts (Figures 10,11, 12, and 13) below, sorting by category, platform, or specific type mod disassembled confidences yields top alternatives which ommit a significant amount of information contained in the total picture conveyed by all the data in Table II. Entropy Weighted combined ranking, on the other hand, as shown in figure 14, selects a top five set which give a clearer picture of the degree and direction of the data ambiguities.

Table III. Results of Different Alternative Rankings

SORTED BY DISASSEMBLED CONFIDENCES							0 243
CAT		PLATFORM		SP TYPE MOD	DIS CONF		
LAND		MISSILE SITE		EXOCET MM40	0.152	15	
SURF		CRUISER		KRESTA 2	C.144	11	
AIR		BOMBER		BEAR	0 098	6	
SUBM		SSGN		ECHO 2	0.074	19	
AIR		HELO		HORMONE	0.063	8	
SORTED BY PLATFORM CONFIDENCES							
CAT		PLATFORM		SP TYPE MOD	DIS CONF		
SURF		CRUISER	0 200	UNKNOWN	0 008	13	
LAND		MISSILE SITE	0.190	UNKNOWN	0 019	17	
AIR		BOMBER	0.160	UNKNOWN	0 062	7	
AIR		FIGHTER	0.122	UNKNOWN	0.036	5	
SUBM		SSGN	0 095	UNKNOWN	0 021	20	
SORTED BY CATEGORY CONFIDENCES							
CAT		PLATFORM		SP TYPE MOD	DIS CONF		
AIR	0.402	UNKNOWN	0.040	UNKNOWN	0.040	10	
SURF	0 250	UNKNOWN	0.050	UNKNOWN	0 050	14	
LAND	0.200	UNKNOWN	0.010	UNKNOWN	0 010	18	
SUBM	0 100	UNKNOWN	0 005	UNKNOWN	0 005	21	
UNKN	0.050	UNKNOWN	0 050	UNKNOWN	0.050	22	
CAT		PLATFORM		SP TYPE MOD	DIS CONF		HSCORE
AIR	0.402	BOMBER	0.160	BEAR	0.098	6	1.698
SURF	0 250	CRUISER	0.200	KRESTA 2	0 144	11	1 693
LAND	0 200	MISSILE SITE	0.190	EXOCET MM40	0.152	15	1 592
AIR	0 402	BOMBER	0.160	UNKNOWN	0 062	7	1 555
AIR	0.402	FIGHTER	0 122	UNKNOWN	0 036	5	1.336

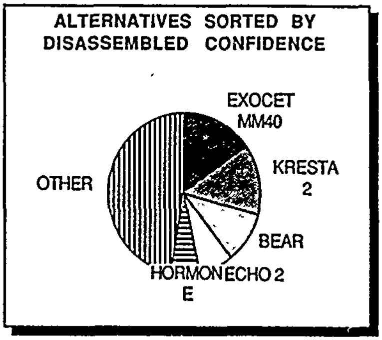


Figure 10. Sort by Disassembled Confidence at the Specific type level.

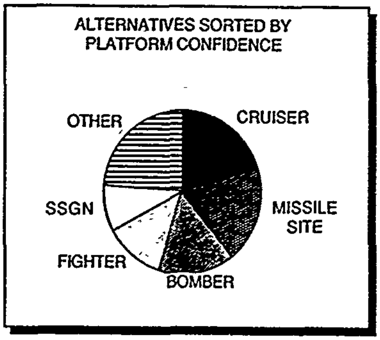


Figure 11. Sort by Assembled Confidence at the Platform level

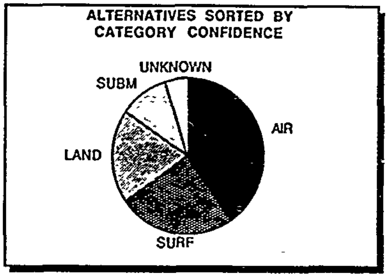


Figure 12. Sort by Assembled Confidence at the Category level

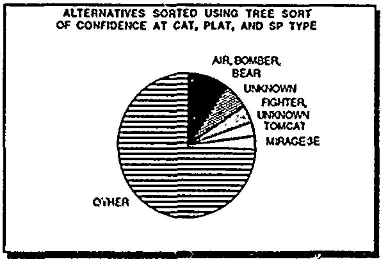


Figure 13. Tree Sort by Confidence at the Category, Platform, and Spec Type level

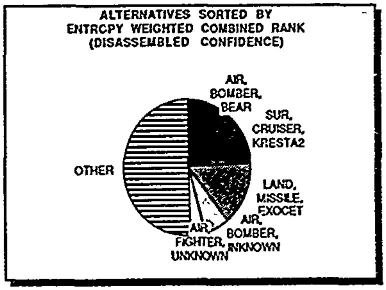


Figure 14. Sort by Entropy Weighted Score Combining Category, Platform, and Spec Type level

Entropy Weighted Ranking

The entropy weighted score is determined by weighting the confidence at each level of an alternative by the a-priori entropy (information) [14] available at that level, computed using the formula:

$$H = -\sum_{i=1}^{N} P(T_i|S_q\&A)*\log_2(P(T_i|S_q\&A)) \text{......} (10)$$

Then, using the H value at each level to compute the HSCORE as follows:

$$HSCORE = H_{Cat}*m(Cat) + H_{Plat}*m(Plat) + H_{SpTy}*m(SpTy) \text{........................} (11)$$

Results of Alternative Ranking Algorithm

The problem solved by the algorithm was that of ranking alternatives at different level of description: Category, Platform, and Specific Type/Type Mod. For example, it is necessary to rank the support for "Air-Fighter-F4C" versus the alternative "Air-No_Statement-No_Statement". The implementation demonstrated that for typical expected cases, the weighting coefficients for the different taxonomy levels could be adjusted so that with good support for a particular specific type, even though its absolute confidence value would be much less than that attributed to the unqualified Category value, could be selected as the "best" overall alternative.

DEVELOPMENT.

The set of algorithms described for identification fusion (MSID) serves as part of the central Multi-Source Track Management (MSTM) function for the Advanced Combat Direction System (ACDS) for major US Navy combatants including aircraft carriers and non-Aegis Cruisers. The multi-source identification function of ACDS is currently in the process of detailed algorithm validation, specification, and development, and will be operational in the mid 1990's, together with enhanced computers, displays, and data communications equipment.

REFERENCES

1. CDS MODEL 5 SYSTEM LEVEL REQUIREMENTS (U), Department of the Navy Naval Sea Systems Command, Washington, DC, rev 4, 30 September 1985 (Confidential).

2. BATTLE FORCE INFORMATION MANAGEMENT PHASE 1 INTERFACE REQUIREMENTS DOCUMENT (U), DEPARTMENT OF THE NAVY, SEA, SPAWAR, AND AIR SYSTEMS COMMANDS, 9 JAN 1986.

3. NATO STANDARD IDENTITY DSCRIPTION STRUCTURE FOR TACTICAL USE (U), NATO Standardization Agreement (STANAG 1241 EDITION 1) - NATO Military Agency for Standardization, 28 April 1982 (NATO Confidential).

4 JTIDS Technical Interface Design Plan - Test Edition [JTIDP-TE] Volume II, Interface Specification (Fixed Word Format) (U), Joint Tactical Command, Control and Communications Agency, Ft Monmouth, NJ, Rev 1, January 1986, (Confidential)

5 JTIDS Technical Interface Design Plan - Test Edition [JTIDP-TE] Volume V, Data Element Dictionary (U), Joint Tactical Command, Control and Communications Agency, Ft Monmouth, NJ, Rev 1, January 1986, (Confidential)

6 JTIDP-TE VOL II, PT 1, op.cit.

7 Seber, G.A.F., "Multivariate Observations", John Wiley & Sons, Inc., 1984.

8 Dempster, A.P., (1966), "New Approaches for Reasoning towards Posterior Distributions based on Sample Data", *Ann. Math. Statist.* 37, 355-374.

9 Dempster, A.P., (1967), "Upper and Lower Probabilities Induced by a Multivalued Mapping", *Ann. Math. Statist.* 38, 325-339.

10 Dempster, A.P., (1968), "A Generalization of Bayesian Inference"with discussion, *J Roy. Statist. Soc. B.* Vol 30, No. 2, 205-247

11 Shafer, Glenn, "a mathematical theory of evidence", Princeton University Press, 1976.

12 Shafer, G, "Belief Functions and Parametric Models", *J. Roy. Statist. Soc. B*, Vol. 44, No. 3, 322-352.

13 Dillard, R.A., "The Dempster-Shafer Theory Applied to Tactical Data Fusion", pp 170-174.

14 Shannon, C.E, "A Mathematical Theory of Communication", Bell System Tech Jounal, Vol 27, pp.379-423, July 1948.

DECISION PROCESSES FOR LARGE SCALE RESOURCE ALLOCATION PROBLEMS
Analysis of Problems with Random Release and Due Dates

Peter Breitkopf and Robert Walker*
Marc Diamond†

Abstract

We present initial work on the extension of the optimal resource allocation techniques currently under development to problems with random release dates and due dates. In the allocation of terminal homing illuminators in naval anti-air warfare engagements, this means considering threats whose velocities may vary with time. For this discussion we restrict ourselves to the analysis of a single threat scenario. Here the problem becomes one of determining when and/or at what distance to engage the threat. The problem definition and problem geometry for this case are established. Consideration is given furthermore to a definition for the stochastic process that describes the threat velocity. It is argued that a filtered Weiner process provides a reasonably good model for the threat and still yields a mathematically tractable problem that can be treated within the context of established theory. It is shown that under these conditions, the solution to the problem of determining when to engage the threat to maximize the probability of kill can be solved explicitly.

1 Introduction

This paper presents progress and continuing research toward further development of the optimal resource allocation techniques, with a particular focus on the optimal allocation of anti-air warfare resources. For scheduling shipboard illuminators to aid terminal engagement sequences in surface-to-air missile (SAM) defense, the analysis to date [1,2] has considered the incoming threat velocity as constant. In fact, the projected trajectory of incoming targets has considerable error introduced by possible maneuvers and uncertainty in the attacker's mission. Furthermore, the velocity derived from the radar can be considered to be noisy. Hence the arrival time of the "tasks" into the scheduling queue must now be represented as a random variable (i.e. we now have random release dates and due dates on the tasks). This affects the intercept timing and therefore has a significant influence on our past approach to modeling this problem, in which deterministic release dates and due dates on tasks were used explicity to derive resource requirement intervals and generate a discrete state-discrete time Markov decision model [1]. To account for the randomness over time in the target kinematics, a continuous state stochastic model is being incorporated into our problem.

We consider first a single threat scenario with no resource conflicts. The analysis of this case is necessary in order to provide the basis for an understanding of scenarios involving multiple threats with inherent resource (illuminator scheduling) conflicts, which is the subject of ongoing work. In the sequel, the problem geometry for a single threat attacking a ship is established under the assumption that we are committed to launch only a single salvo against it. In this restricted problem we are interested in determining when and/or at what distance the threat should be engaged in order to optimize the probability of kill. This arises as a problem, because of the fact that the threat velocity cannot be predicted with certainty means that, for a given

*FMC Corporation, Central Engineering Laboratories, Santa Clara, CA
†FMC Corporation, Advanced Systems Center, Minneapolis, MN

SAM launch time and threat distance one cannot guarantee that the threat will be intercepted while it is in the engagement envelope. The case in which a single observation of the threat distance is made is considered first and then generalized to the situation in which the threat is continuously observed. An appropriate definition for the velocity process is then examined. We argue that a filtered Weiner process provides the best available model for the threat. Finally, we conclude that when modeled this way, the problem of determining the optimal engagement opportunity becomes a function only of the threat distance and can thus be very easily be solved.

2 Problem Geometry

Consider first a single threat observed at time t_0 to be some distance, x_0, from a ship and assume that we are restricted to launching a single salvo against it. Assume that there is no error in the initial observed range, but possible error in the observed velocity. We want to determine when we should launch the defending surface-to-air missile so as to maximize the probability of destroying the threat. As per our assumption, we cannot predict with certainty the future location of the threat for $t \geq t_0$ but assume the the threat trajectory will be in some "trajectory cone" as shown in the distance-time plot of figure 1. The rays bounding this cone represent an assumed maximum and minimum velocity for the threat, which are respectively, 5/4 and 5/6 "distance units per time unit" for this example. Because we assume it to be constant (as a first order approximation), the trajectory of the SAM describes a straight line on this plot. Given a fixed launch time, T, the SAM will intercept the threat if its trajectory intersects the *actual* (unknown) trajectory of the threat within the engagement envelope. The probability of this event (which will depend on the assumptions about the distribution of the random variable representing the threat velocity as well as the launch time) will be denoted as $p_i(T)$, where the argument, T, makes explicit the dependence on the launch time. We assume that the probability of kill given intercept, $p_{k|i}$, is constant, and since

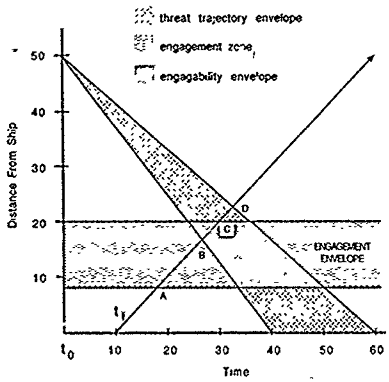


Figure 1: DISTANCE-TIME PLOT SHOWING ENGAGEABILITY OF A SINGLE THREAT DEPENDING ON LAUNCH TIME.

$$p_k(T) = p_{k|i}\, p_i(T) \qquad (1)$$

our qualified objective is clearly:

[P1] maximize: $p_i(T)$

subject to: $T \geq t_0$.

For any given launch time, T, the SAM trajectory intersects the threat trajectory cone at two points, (B and D of figure 1). It also intersects the engagement envelope at two points (A and C of figure 1). Let us assume, for the sake of illustration, that the intercept point of the threat trajectory is uniformly distributed between points A and B[1] so the probability of intercept for launch time t_1 is given as.

$$p_i(T) = \frac{d(B,C)}{d(B,D)} \qquad (2)$$

where d is some appropriate distance metric. Figure 2 provides a plot of $p_k(T)$ as a function of launch

[1]Implicit in this is an assumption about the stochastic process for the threat velocity as discussed further below. In fact, there may not be a "reasonable" physical process that will result in this assumption being valid. The assumption is made here for illustrative purposes only.

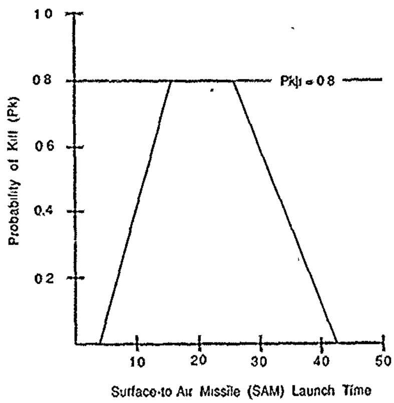


Figure 2: PROBABILITY OF KILL, p_k, AS A FUNCTIONOF LAUNCH TIME UNDER THE ASSUMPTION THAT THE THREAT POSITION WILL NOT BE OBSERVED AFTER THE INITIAL DETECTION AT TIME t_0.

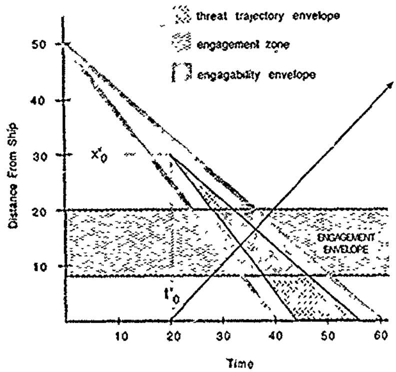


Figure 3: DISTANCE-TIME PLOT SHOWING THE TRANSLATION OF THE THREAT TRAJECTORY CONE FOR AN OBSERVATION MADE AT TIME t'_0.

time, T, for the conditions illustrated in figure 1, assuming that d is the standard Euclidean metric, and that we cannot observe the threat after the initial observation at time t_0. For this example, it is assumed that $p_{k|i} = 0.8$. In this case the solution to P1 is clear.

3 Problem Consideration for a Continuously Observed Threat

Assume now that we are allowed to observe the threat after its initial detection. If the velocity process is assumed to be stationary, an observation at time t'_0 of a threat distance x'_0 results in a translation of the trajectory cone so that its apex is at the point (t'_0, x'_0) (figure 3). If we are not going to be allowed any observations beyond time t'_0 then the problem becomes similar to that cited above. Otherwise, we can no longer ask *when* we should launch to maximize p_k, because the answer to this question will depend on the future behavior of the threat, which we can now observe. The dimension of threat distance, $x(T)$ must then be explicity represented in the problem:

[P2] maximize: $p_i(T, x(T))$

subject to: $T \geq t_0$.

In fact, the only reasonable way in which the problem can now be stated is, "given that I observe the threat at distance $x(T)$ at this point in time, shall I launch immediately, or wait and continue to observe it". In general this problem is very difficult to formulate and solve. For example, at a specific point in time T and an observed distance $x(T)$ we can calculate $p_k(T, x(T))$[2], but there may be some chance that the threat will pass into a region of higher p_k at some future point in time. Under some conditions, however, this problem becomes tractable For example, if we constrain ourselves to

[2]Note that once the SAM is launched further observation of the threat . avior is unimportant from the standpoint of the system control. Hence for $t \geq T$ the situation reverts, essentially, to the case of a single observation (made at time T). Hence, we can easily compute $p_k(T, x(T))$ for all feasible points $(T, x(T))$

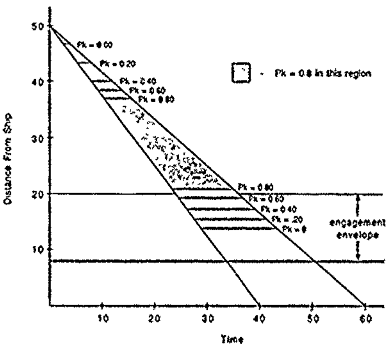


Figure 4: A CONTOUR PLOT FOR p_k AS A FUNCTION OF LAUNCH TIME, T AND OBSERVED DISTANCE AT LAUNCH TIME, $x(T)$ FOR THE EXAMPLE OF FIGURES 1, 2, AND 3.

make observations at (e.g., regular) discrete points in time (as opposed to making continuous observations), then a classical optimal stopping problem [3], which can be solved using dynamic programming, results. As another example, figure 4 provides a contour plot of $p_k(T, x(T))$, for the hypothetical situation characterized by figures 1 and 2 under the assumption of a continuously observed threat. Note that the contours run parallel to the time axis, as will necessarily be the case when the process is stationary. Note also that the problem P2 can readily be solved by a policy that says, in effect, "wait until the threat reaches a point, T such that $p_k(T, x(T))$ is optimum (i.e. within the shaded region), and launch the SAM." Because the p_k function is independent of time in this case, it is guaranteed that the system will reach such a state

4 Consideration of Velocity Process

It is very important when modeling the threat velocity as a random process that its mathematical description reflects as realistically as possible the actual nature of the underlying physical process. On the other hand it is desirable that the model be simple enough to derive results in closed form. The most simple model would be to consider the velocity at any point in time as a random variable that is uniformly distributed over a finite interval, in the same spirit as the discussion of the previous section. Although this process is mathematically easy to handle, it is not close to the actual physical reality in that it appears unlikely that there exist no physical process of a continuous time parameter that is consistent with the stated assumptions.

Another more realistic choice assumes the velocity of the incoming threat to be an independent Gaussian Process with non-zero mean and bounded variance. This allows for a fluctuation of the sample function of the integral process in time. The integral of such a process is closely related to a Wiener Process or Continuous Random Walk for which an extensive mathematical theory exists [4]. The Wiener process has properties sufficient to make the problem mathematically tractable. However, since the velocity is assumed to be independent and stationary, its power spectrum contains very high frequencies, which means that the acceleration becomes unbounded. To force a more predictable behavior on this process it is reasonable to perform a low pass filtering operation. This imposes a Markov structure by introducing statistical time dependencies.

To avoid unbounded sample functions and keeping the process realizations inside a cone emerging from the initial observation point in time the introduction of a stationary independent velocity process with uniform first density, appears to model the situation of a randomly varying threat most accurately. This process may also be low pass filtered. Applying the Law of Large Numbers one can show that the first order densities of the derived integral process become approximately Gaussian for large t and its dynamic behavior approaches that of a Wiener Process. Hence, it makes sense for our studies to assume that an unfiltered independent stationary Gaussian velocity process is a valid approximation and leads to meaningful results without extensive simulations.

5 Computation of the Probability of Intercept

As has been pointed out above, maximizing the probability of kill is equivalent to maximizing the probability of intercept $p_i(T)$. The stochastic process of threat position (integral of velocity process) is observed and the result of this observation is x_0 (figure 3). This observation restricts the number of possible time functions to a subset of realizations of the process that lie roughly in a cone that emerges from the observation parameters (t_0, x_0). A SAM with a constant velocity v_m may be launched at $T \geq t_0$. Interception is possible if there exists t such that $X(t) - v_m(t-T) = 0$ and $a \leq X(t) \leq b$. This means that the realization, $x(t)$, of the process of the threat position and the deterministic missile trajectory intersect within the engagement interval $[a, b]$. It is therefore reasonable to compute the probability of intercept p_i in the following manner:

$$p_i = P[X(t) - v_m(t-T) = 0, a \leq X(t) \leq b \mid X(t_0) = x_0] \tag{3}$$

The assumption of a gaussian velocity process with mean η and bounded variance leads to a Wiener Process (figure 4), whose first order density conditioned on the observation $x(t_0)$ can be described as

$$f_{X(t)|x(t_0)=x_0} = \frac{B}{\sqrt{2\pi\alpha(t-t_0)}} e^{-\frac{(x-(x_0-\eta(t-t_0)))^2}{2\alpha(t-t_0)}} \tag{4}$$

B is a normalization factor and α is proportional to the variance of the velocity process. As shown in figure 11, if we start from the observation point (t_0, x_0), the first order conditional density of this process is centered around a linearly varying mean $E[X(t)] = x_0 - \eta(t - t_0)$ with a linearly increasing variance. This position process is clearly not stationary.

If $x(t)$ is replaced by $v_m(t-T)$, a new function

$$g(x_0, t_0, T, t) = \frac{B}{\sqrt{2\pi\alpha(t-t_0)}} e^{-\frac{[v_m(t-T)-x_0+\eta(t-t_0)]^2}{2\alpha(t-t_0)}} \tag{5}$$

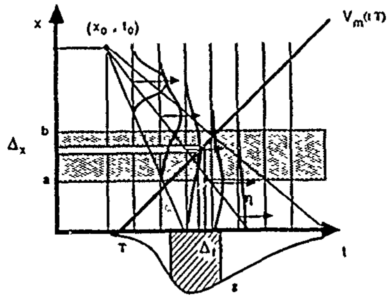


Figure 5: CALCULATION OF p_i ASSUMING A GAUSSIAN VELOCITY PROCESS.

can be constructed. It can be shown that its integral has a least upper bound M. Normalization of the function $g(x_0, t_0, T, t)$ using M leads to a new function $\bar{g}(x_0, t_0, T, t)$ that has the properties of a probability density function. Using equation (2) the probability of intercept can be computed by integrating the function $\bar{g}(x_0, t_0, T, t)$ from $\frac{a}{v_m} + T$ to $\frac{b}{v_m} + T$. This leads to the following optimization problem:

$$\text{max:} \quad p_i(x_0, t_0, T) = \int_{\frac{a}{v_m}+T}^{\frac{b}{v_m}+T} \bar{g}(x_0, t_0, T, t)\,dt$$

$$\text{subject to: } x_0 \geq 0, \quad T \geq t_0 \geq 0$$

which will be considered here.

6 Results

It is very important to show that the derived probability of intercept contains enough information about the underlying random process' dynamics to serve as a useful measure for optimization. Some of the results will therefore show the variation of p_i

dependent on different process parameters, which includes the special case of a deterministic threat trajectory.

Figure 6 shows the variation of SAM speed. Considering p_i as a function of v_m/η attains a maximum which allows one to determine an optimal launch velocity for each α. This function degenerates to a rectangular function for $\alpha = 0$, which corresponds to the deterministic case. Figure 7 shows that as α decreases to 0, p_i becomes 1 if the theoretical interception occurs within the engagement interval and 0 otherwise. The case where the missile velocity is held constant and α is increased is shown in figure 8. Here again it is shown that if theoretical interception occurs within $[a, b]$ the probability p_i will approach 1 for α approaching 0 (upper curve in figure 9). If a theoretical interception occurs outside $[a, b]$, p_i tends to go to 0. Therefore both results contain the deterministic case. Figure 10 shows the situation where the threat process is observed at $t_0 = 0$ and a missile launch is attempted after a delay time T. Given one observation, figure 11 shows a curve that attains a maximum. This allows us to determine an optimal launch time T that optimizes p_i. If continuous observations (t_0, x_0) are allowed and the launch of the missile occurs at the time of observation, $p_i(t_0, x_0)$ will remain constant as t_0 changes. This is clearly shown in figure 12. The value p_i therefore is only dependent on the observation distance x_0 and attains a maximum which allows one to determine the optimal distance of the threat where immediate launch will yield the highest probability of intercept (figure 13).

7 Conclusions

From these studies one can conclude that to maximize the probability of intercepting a one dimensional threat whose velocity is an independent stationary Gaussian random process with non-zero mean and bounded variance it suffices:

a. to observe the treat once and then launch the missile after an optimal delay time T (figures 10 and 11), or

b. to observe the threat continuously until the optimal threat distance (figure 12 and 13) is

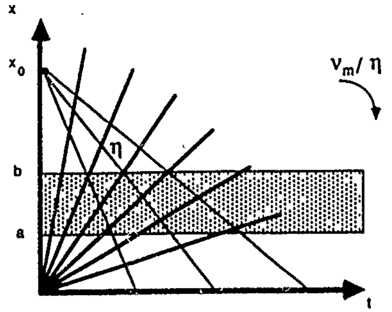


Figure 6: VARIATION OF MISSILE SPEED.

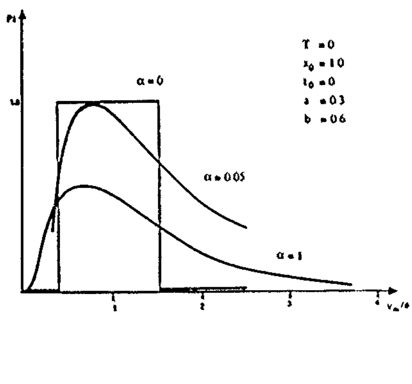


Figure 7. A PLOT OF p_i FOR VARYING MISSILE SPEEDS.

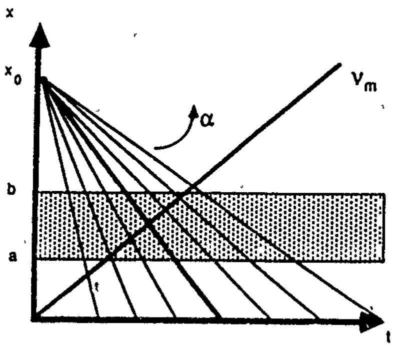


Figure 8: CHANGE IN THE VARIANCE OF THE VELOCITY PROCESS.

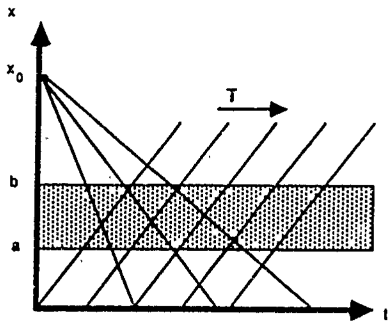


Figure 10: VARIATION OF DELAY IN LAUNCH TIME AFTER AN INITIAL OBSERVATION.

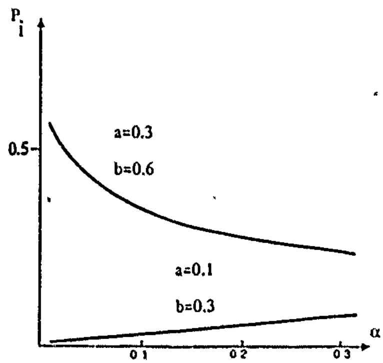


Figure 9: THE VALUE OF p_i FOR DIFFERENT VARIANCES OF THE VELOCITY PROCESS.

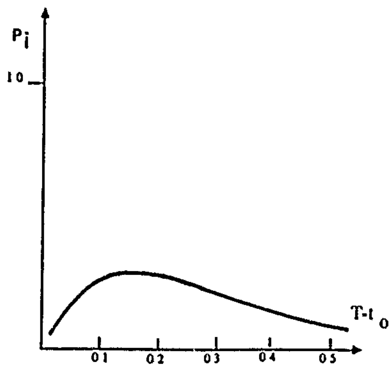


Figure 11: THE VALUE OF p_i FOR VARYING LAUNCH TIME DELAYS.

reached and then launch immediately.

Since p_i is not dependent on the observation time t_0, each realization of the random process will reach this point eventually[3].

Possible ways to continue this work would be to look at a more realistic stochastic description of the velocity process allowing non-Gaussian first order densities with final support (i.e. hard limitation of the velocity) and impose a linear Markovian structure by filtering. Another direction to extend this work is to use the results for the Gaussian density presented here to describe the probability of intercept of multiple launches employing a form of "shoot and look" doctrine. It seems reasonable that a solution to the problem of optimizing a multiple salvo engagement against a single threat can be solved using dynamic programming. For example, the optimal first launch opportunity in a two salvo engagement would have the form:

$$\max: 1 - p_m(t, x(t)) \int_A p_{o|t}(t'_0) f(t'_0) dt$$

subject to: $t \geq t_0$

Here, p_m(probability of miss) $= 1 - p_k$, $p_{o|t}(t')$ is the probability density for the outcome time of an engagement initiated with a SAM launch at time t (that is, the time at which we know whether or not the SAM successfully destroyed the threat), and $f(t)$ is defined by:

[P3] $f(t'_0) = \min p_m(T, x(T))$

subject to: $T \geq t'_0$.

Obviously, further details and the tractability of this problem have yet to be established.

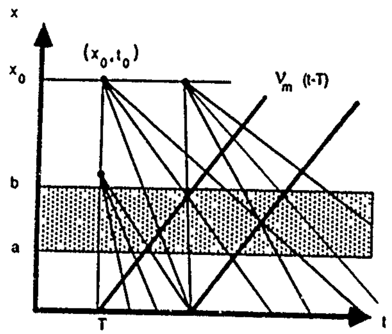


Figure 12: CONTINUOUS OBSERVATION AT (t_0, x_0) OF THE INTEGRATED VELOCITY PROCESS.

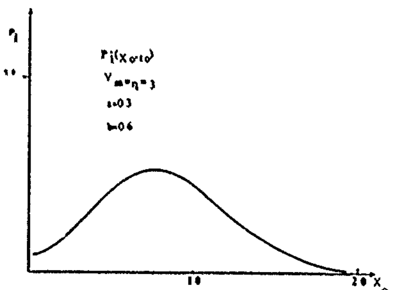


Figure 13: THE VALUE OF p_i FOR VARYING OBSERVATION DISTANCES.

References

[1] M. D. Diamond and O. M. Carducci, "Decision processes for large scale resource allocation problems," in *Proceedings of the 8th MIT/ONR Workshop on C^3I Systems*, (A. Levis and M. Athans, eds.), (Cambridge, Mass.), pp. 153–160, M.I.T./O.N.R., M.I.T. Press, July 1984.

[3]Note the similarity with the general case of a stationary process discussed in section 3 above.

[2] M. D. Diamond, J. A. Newhouse, and J. C. Kimbel, "Decision processes for large scale resource allocation problems: extensions and results," in *Proceedings of the 1987 Command and Control Research Symposium*, (J. T. Miller, ed.), (Washington, D.C.), pp. 163–170, Joint Directors of Laboratories, Science Applications International Corporation, June 1987.

[3] S. M. Ross, *Introduction to Stochastic Dynamic Programming.* Reading, Mass.: Academic Press, 1983.

[4] D. Kannan, *An Introduction to Stochastic Processes. North Holland Series in Probability and Applied Mathematics*, New York, NY: North Holland, 1979.

OPTIMUM CONFIGURATION FOR DISTRIBUTED TEAMS OF TWO DECISION-MAKERS

by

Jason D. Papastavrou and Michael Athans

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

This paper deals with problems of quantitative organizational design. We show that the optimal architecture of even a very simple team of two decision makers (DMs) performing binary hypothesis testing depends on variables external to the team. On the other hand, there exist particular probability distributions for the observations which lead to unambiguous optimal architectures in which the "better" decision maker makes the final team decision based upon finite-bit messages from the "worse" decision maker. But, even in these cases the results are difficult to generalize for teams with three or more DMs, because of the complexity of the problem. A heuristic algorithm for organization design is presented.

1. INTRODUCTION AND MOTIVATION

Our main research goal is to develop basic understanding of decision making in distributed organizations. As we shall see such problems can become very complicated because of the distributed (decentralized) decision process. In order to gain understanding into the basic fundamental issues we need a paradigm which represents simple decision making, and whose centralized version is easy to formulate, solve, and compute. We have adopted the problem of hypothesis testing as our basic paradigm; see references [1] to [4] for related prior research in this area.

The classic decision problem in this setting relates to the design a team to perform target detection (no target vs target present) using several distributed sensors. Suppose that each sensor has significant computational capability to process his raw returns and can perform local target detection. Because of the unreliability and uncertainty of the observations there is a high probability of error associated with each sensor's decision when he operates in isolation. Thus, it is desirable to have many sensors to operate together as a team to decrease the error probability. In order to achieve this, we have to define the architecture of the organization (i.e. which sensor communicates with whom) and derive a decision protocol to fuse the "tentative" sensor decisions into a global team decision.

We employ a binary hypothesis testing model, which can be generalized to more general hypothesis testing problems. These are indeed generic in the situation assessment C^2 function. We would like to develop a quantitative design methodology to deal with them.

We examine several important issues of these problems, starting with real-time decision making rules for distributed hypothesis testing. The team architecture is the way the DMs of the team are set-up. We want to obtain the performance of a given team architecture (say the probability of error) and compare the performance of alternative architectures. We also seek to design an organization to meet some global team performance specifications and study the effect of adding a new DM to the team. Finally, we would like to understand and develop the theoretical aspects and computational complexity associated with this class of problems.

Suppose that a team consists of N DMs. Evidently, the team may have many alternative architectures and communication protocols. For example, if N = 3, we can see two different architectures in Figure 1a and 1b. The environment consists of several hypotheses. Each DM receives a conditionally independent observation and makes a tentative decision, based upon his own measurement and the decisions of the other processors which have been transmitted to him, according to some specified communication protocols. The final team decision has some costs associated with it. We would like to determine somehow which configuration results into superior performance; moreover, given three DMs and a particular configuration, we would like to determine which DM should be employed in each position. We would also like to test the effects of different communication protocols. Another type of problem is illustrated in Figure 1c. Given a team of DMs which does not meet certain specifications, we would like to determine what DM should be introduced to the team and in what position, for the team to meet the posed specifications.

In this paper all the hypothesis testing performed is assumed binary. In Section 2, we will discuss the optimum configuration of a team consisting of two DMs. In Section 3, we examine the same problem for the special case where the observations of the DMs are described by Gaussian distributions with different variances under each hypothesis. In Section 4, we present a problem of organizational design and an algorithm to solve it. Finally in Section 5, we present some concluding remarks and suggestions for future research.

2. TWO DM ORGANIZATIONS

2.1 General Remarks

Since organizations w..h [illegible] DMs are key building blocks for larger organizations, [illegible] objective is to study them extensively and analyze them [illegible]pletely. There are two alternative architectures for this type of team: fusion and tandem (Figure 2). Since the DMs in the tandem architecture can always employ the decision rules of the DMs in the fusion architecture (hence even the optimal decision rules for the fusion architecture), the performance of the tandem architecture is always at least as good as the performance of the fusion architecture. Thus, we will restrict ourselves to the study of the tandem architecture.

FIGURE 1
LONG RANGE OBJECTIVES

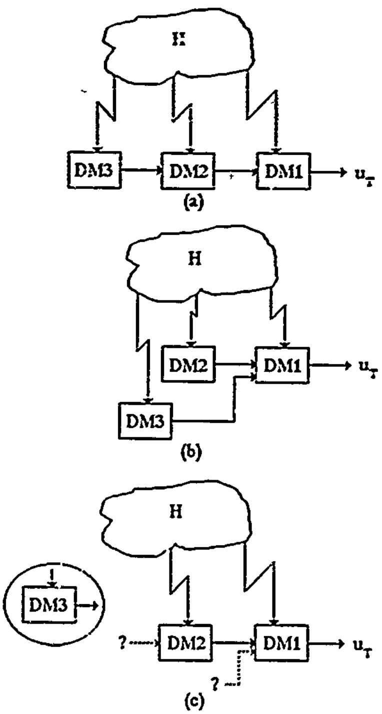
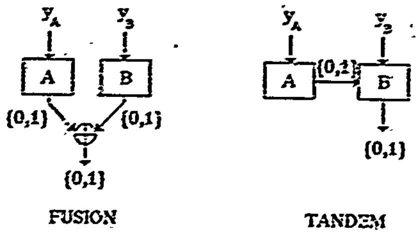


FIGURE 2
TWO DM ORGANIZATIONS



2.2 The ROC Curve

In the binary hypothesis testing problem each DM can be characterized by his Receiver Operating Characteristic (ROC) Curve. This curve plots the probability of detection as a function of the probability of false alarm.

The probability of detection, P_D, is the probability that the DM decides $u = 1$ (indicating that H_1 is the true hypothesis) when H_1 is indeed true and is defined by

$$P_D = \int_n^{\infty} P_{\Lambda|H_1}(\Lambda \mid H_1)\, d\Lambda \qquad (1)$$

where

$$\Lambda(y) = \frac{P(y \mid H_1)}{P(y \mid H_0)}$$

is the likelihood ratio and n represents the decision threshold.

The probability of false alarm, P_F, is the probability that the DM decides $u = 1$ when H_0 is the true hypothesis and is defined by

$$P_F = \int_n^{\infty} P_{\Lambda|H_0}(\Lambda \mid H_0)\, d\Lambda \qquad (2)$$

Thus, the ROC curve is expressed by two parametric equations, with the threshold parameter n varying from zero to infinity; in general, can not be expressed in a closed form. The ROC curve is <u>concave</u> and it has another useful property; suppose that by substituting n^* in equations (1) and (2), the point (P_F^*, P_D^*) of the ROC curve is obtained. Then, the slope of the tangent to the ROC curve at (P_F^*, P_D^*) is n^* (Figure 3). Consequently, if a DM performs detection with some given n^*, his optimal operating point is the point of the ROC curve where the slope of the tangent is n^*.

In our research, we use the ROC curve to quantify the relative expertise of different DMs. Moreover, since the team of DMs also performs binary hypothesis testing, team performance can also be quantified by the team ROC curve. If the ROC curve of DM A is higher than the ROC curve of

DM B, then we say that A is a better DM than B, because for the same level of probability of false alarm, A will have a higher probability of detection (Figure 4a). But, the DMs can not always be ranked globally because sometimes their ROC curves intersect (Figure 4b).

FIGURE 3
THE ROC CURVE

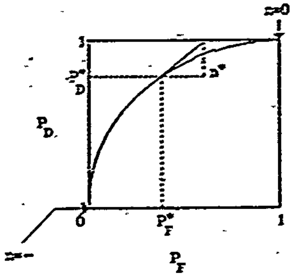
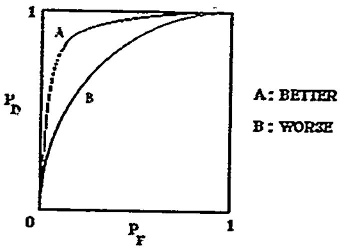
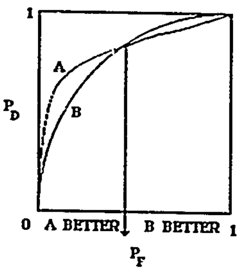


FIGURE 4
RANKING DECISION MAKERS



(a)



(b)

2.3 The Problem

Consider a team consisting of two DMs in tandem architecture, which performs binary hypothesis testing (Figure 5a). The prior probabilities ($P_i = P(H_i)$ for $i = 0,1$) are assumed known, as well as the costs $J(u,H)$ which are incurred by the team when it decides u and H is the true hypothesis. It is assumed that it is more costly for the team to err than to be correct. The team objective is to minimize the expected cost incurred by the team.

Each DM receives a conditionally independent observation. One DM, called the consultant DM, makes a binary decision ($u_c = 0$ or $u_c = 1$) based on his measurement, y_c, and transmits it to the othe DM, called the primary DM. Then, the primary DM has to make the team decision (based upon his own measurement y_p and the message from the consultant) which has to be either $u_p = 0$ or $u_p = 1$ indicating that the corresponding hypothesis is considered to be true.

The optimal solution for the decision rules of the two DMs is given by likelihood ratio test with constant thresholds [3]. For the primary DM:

$$\text{If } u_c = 0: \qquad \Lambda(y_p) \underset{u_p=0}{\overset{u_p=1}{\gtrless}} \frac{1-P_F^c}{1-P_D^c}\, n \qquad (3)$$

$$\text{If } u_c = 1: \qquad \Lambda(y_p) \underset{u_p=0}{\overset{u_p=1}{\gtrless}} \frac{P_F^c}{P_D^c}\, n \qquad (4)$$

For the consultant DM:

$$\Lambda(y_c) \underset{u_c=0}{\overset{u_c=1}{\gtrless}} \frac{P_F^1-P_F^0}{P_D^1-P_D^0}\, n \qquad (5)$$

where

$$n = \frac{P_0}{P_1}\,\frac{J(1,H_0)-J(0,H_0)}{J(0,H_1)-J(1,H_1)}$$

and P_D^i (P_F^i) is the probability of detection (probability of false alarm) for the primary DM when $u_c = i$ was received by the consultant ($i = 0,1$) and P_D^c (P_F^c) is the probability of detection (probability of false alarm) for the consultant DM, when both DMs are operated according to the optimal decision rules of eqs.(3)-(5). For example,

$$P_F^0 = \Pr\left(\Lambda(y_p) \ge \frac{1-P_F^c}{1-P_D^c}\, n \mid H_0\right) \qquad (6)$$

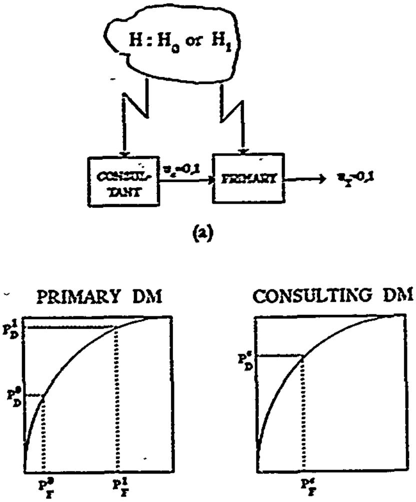
Figure 5b demonstrates the form of the operating points.

The ROC curve of the team as a whole can be computed and is given by:

$$P_F^T = (1-P_F^c)\,P_F^0 + P_F^c\,P_F^1 \qquad (7)$$
$$P_D^T = (1-P_D^c)\,P_D^0 + P_D^c\,P_D^1 \qquad (8)$$

FIGURE 5

THE PROBLEM AND ITS SOLUTION



(a)

(b)

Note that the team ROC depends not only upon the characteristics ("expertise") of the individual DMs, but also on the particular way that they have been constrained to interact (the team or organization architecture).

2.4 Architecture Comparisons

Suppose that one of the two DMs is "better" than the other, i.e. his ROC curve is higher than the ROC curve of the other DM. There exist two candidate architectures for the team; either make the "better" DM the primary DM or make the "better" DM the consultant DM. Recall that the primary DM makes the final team decision. We would like to determine which of the two architectures yields better performance than the other for all values of n, that is whether the optimal architecture is independent of the external parameters of the problem (details of cost function, prior probabilities) which determine the value of n.

The architecture with the better DM as the primary DM was conjectured [3] to be better. This conjecture is appealing from an intuitive point of view; given two DMs one would like to have the "better" DM make the final decision, independent of the prior probabilities and the cost assignments. If this were the case, then the optimal way of organizing two DMs would not change, say, as the prior probabilities of the underlying hypotheses vary. Unfortunately, as we show below, this conjecture can be false.

2.5 A Counterexample to the Conjecture

In Figure 6, we present the ROC curves of two DMs, one better than the other according to our prior definition. Table 1 contains the discrete distributions of their observations; the elements in the matrix denote probabilities. For example, the "worse" DM will observe y = 1 with probability 0.1 if H_0 is true and with probability 0.5 if H_1 is true. From Table 1 we can then see that the better DM has as good or better discrimination of the two hypotheses, and of course this is reflected in the dominance of his ROC curve in Figure 6.

In order to establish the counterexample we compared the two architectures using tedious, albeit straight forward calculations of the probability of error. The results are illustrated in Table 2, which contains the probability of error for two different values of n for each architecture -- B denotes the "better" DM, while W denotes the "worse" one. For n = 1.0 having the better DM as the consultant is optimal while for n = 0.38 having the better DM as the primary is optimal. This can be also verified by deriving the team ROC curves for each architecture (Figure 7a). As the close-up of Figure 7b shows the two ROC curves intersect near P_F= 0.3. Thus, in this special example, the optimal team

FIGURE 6
THE ROC CURVES

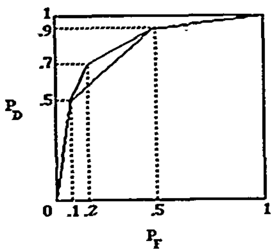


TABLE 1
DESCRIPTION OF DMs

"WORSE" DM

y \ H	H_0	H_1
1	0.1	0.5
2	0.4	0.4
3	0.5	0.1
4	0	0

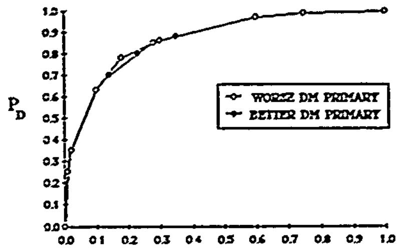
"BETTER" DM

y \ H	H_0	H_1
1	0.1	0.5
2	0.1	0.2
3	0.3	0.2
4	0.5	0.1

TABLE 2
COMPARISONS OF
PROB. OF ERROR

	$\boxed{B} \to \boxed{W} \to$	$\boxed{W} \to \boxed{B} \to$
n = 1.00	0.200 (optimal)	0.215
n = 0.38	0.1840	0.1833 (optimal)

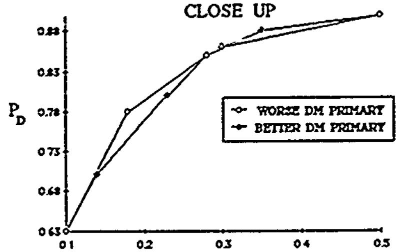
FIGURE 7
TEAM ROC CURVES



P_F

(a)

CLOSE UP



P_F

(b)

architecture depends on the value of n (i.e. the numerical values of the prior probabilities and costs). On the other hand, for this example, both architectures have very similar performance, since their ROC curves are quite close (Figure 7a).

3. COMPARING GAUSSIAN VARIANCES

3.1 General Remarks

Consider special case of the problem presented in Section 2.3 above, where each DM receives two independent observations distributed with the Gaussian distribution with different variance under each hypothesis. The ROC curves in this case are simple and given in a closed form [1]. A summary of this case is given in Table 3.

3.2 The First Architecture

Suppose that the better DM is made the primary. Then, from the solution of the problem and the property of the ROC curve

$$n^c = \frac{P_F^1 - P_F^0}{P_D^1 - P_D^0}\, n = \frac{\partial P_D^B}{\partial P_F}\Big|_{(P_F^c,P_D^c)} = \frac{\sigma_0^2}{N\sigma_1^2}\,\frac{P_D^c}{P_F^c} \qquad (9)$$

$$n^0 = \frac{1 - P_F^c}{1 - P_D^c}\, n = \frac{\partial P_D^W}{\partial P_F}\Big|_{(P_F^0,P_D^0)} = \frac{\sigma_0^2}{\sigma_1^2}\,\frac{P_D^0}{P_F^0} \qquad (10)$$

$$n^1 = \frac{P_F^c}{P_D^c}\, n = \frac{\partial P_D^W}{\partial P_F}\Big|_{(P_F^1,P_D^1)} = \frac{\sigma_0^2}{\sigma_1^2}\,\frac{P_D^1}{P_F^1} \qquad (11)$$

where the superscripts B (better) and W (worse) indicate which DMs ROC curve is being differentiated. Solving the system of eqs.(9)-(11) and recalling the concavity of the ROC curve, we obtain that in this case

$$(P_F^1, P_D^1) = (1,1)$$

which implies that whenever $u_c = 1$ is received from the consultant, the primary decides $u_p = 1$ independent of his own observation. Substituting into (7) and (8), we obtain that the team ROC curve in this case is given by:

$$P_F^T = P_F^0 + P_F^c - P_F^0 P_F^1 \qquad (12)$$
$$P_D^T = P_D^0 + P_D^c - P_D^0 P_D^1 \qquad (13)$$

for some (P_F^0, P_D^0) [(P_F^c, P_D^c)] in the ROC curve of the worse [better] DM.

TABLE 3
COMPARING GAUSSIAN VARIANCES

WORSE DM :	BETTER DM :
$Y_1, Y_2 \sim N(0, \sigma^2)$	$Y_1, Y_2 \sim N(0, \sigma^2)$
$H_0 : \sigma^2 = \sigma_0^2$	$H_0 : \sigma^2 = \sigma_0^2$
$H_1 : \sigma^2 = \sigma_1^2$	$H_1 : \sigma^2 = N\sigma_1^2$
$P_D = P_F^{\frac{\sigma_0^2}{\sigma_1^2}}$	$P_D = P_F^{\frac{\sigma_0^2}{N\sigma_1^2}}$

with : $\sigma_0^2 < \sigma_1^2$; $N > 1$

3.3 The Second Architecture

Suppose that now the better DM is made the primary. Then, we can arbitrarily assign to the DMs the following operating points:

(P_F^0, P_D^0): to the Consultant (Worse) DM
(P_F^c, P_D^c): to the Primary (Better) DM when $u_c = 0$ is received
(1,1) : to the Primary (Better) DM when $u_c = 1$ is received

Substituting into eqs. (7) and (8), we obtain eqs. (12) and (13) again. Since for this arbitrary assignment of operating points, the architecture with the better DM as the primary can achieve performance equal to the optimal performance of the other architecture, the <u>better DM should always be the primary DM.</u>

3.3 Obtaining the Team ROC Curve

Suppose that the better DM is the consultant. Then, from the system of eqs. (9)-(11), we can solve for P_F^c to obtain:

$$P_F^c = \left[N\sigma_1^2(1-P_D^c)n \frac{[\sigma_0^2(1-P_D^c)]^{-\frac{\sigma_1^2}{\sigma_1^2-\sigma_0^2}} - [\sigma_1^2(1-P_F^c)n]^{-\frac{\sigma_1^2}{\sigma_1^2-\sigma_0^2}}}{[\sigma_0^2(1-P_D^c)]^{-\frac{\sigma_0^2}{\sigma_1^2-\sigma_0^2}} - [\sigma_1^2(1-P_F^c)n]^{-\frac{\sigma_0^2}{\sigma_1^2-\sigma_0^2}}} \right]^{\frac{N\sigma_1^2}{N\sigma_1^2-\sigma_0^2}}$$

This is an equation of just P_F^c. We could have substituted for P_D^c from the equation of the ROC curve of the consultant (better) DM, but did not do it because of space limitations. If the equation is solved P_F^c is obtained. Moreover:

$$P_F^0 = \left[\frac{\sigma_1^2}{\sigma_0^2}\frac{1-P_F^c}{1-P_D^c}n\right]^{-\frac{\sigma_1^2}{\sigma_1^2-\sigma_0^2}}$$

By substituting into the equation of the ROC curve of the primary (worse) DM, P_D^0 is obtained. Finally by substituting for all the probabilities into equations (7) and (8), the team ROC curve is obtained as a function of n, the variances of the DMs and N.

It should be clear that the team ROC curve will not be of the same form as the ROC curves of the individual DMs. In fact, it is not even given by a closed form expression. Thus, we cannot easily extend the result to the case of three DMs in a tandem architecture.

4. DESIGNING ORGANIZATIONS

4.1 General Remarks

Suppose that we are given a team of DMs and a set of requirements on the team performance, which are not met. We could perform several changes in the team, such as adding or deleting a DM or changing the team interconnections, or redesigning the communication protocols, to make the team meet posed performance requirements. Presently, we are employing a trial and error approach because of the mathematical complexity of the problems; we hope for analytical insight from our future research.

4.2 Adding a New DM

By introducing the "perfect" DM to the team, that is a DM who always knows which is the true hypothesis (i.e. his ROC curve goes through $(P_F, P_D) = (0,1)$), the team probability of error will be reduced to zero. Hence, specifications no matter how strict can always be met.

We would like to introduce a trade off between the team performance and the quality of the DM to be introduced. To measure quality we need to rank the DMs even in cases of ambiguity (Figure 4b). The measure we will employ is the area under the ROC curve. This measure is scalar and preserves the ranking of unambiguous situations (Figure 4a), the "perfect" DM has a measure of 1 and the "worst" DM (the DM who is equally likely to choose between either hypothesis independent of his observation) has a measure of 0.5.

The design problem will now be to find the "cheapest" DM which will enable the team to meet the requirements, by cheapest meaning the DM with the smallest area under the ROC curve.

4.3 A Sample Problem

Suppose we are given a DM ("old") with ROC curve:

$$P_D = P_F^{\frac{\sigma_0^2}{\sigma_1^2}}$$

and a set of requirements for team performance (i.e. minimum levels of probability of detection for specified levels of probability of false alarm). We want to find the "cheapest" DM ("new") with omiomorphic ROC curve to the old DM, that is:

$$P_D = P_F^{\frac{\sigma_0^2}{K\sigma_1^2}} \quad ; K \geq 0$$

which will make the team satisfy the requirements. In this case, the <u>smaller</u> the value of the constant K the <u>cheaper</u> the DM.

The problem is the same as the one described in Section 2.3 above. The two possible architectures are to use the new DM as the consultant or to use the new DM as the primary.

In the following algorithm we use our theoretical analysis which suggests that <u>the better DM should be the primary</u> to avoid a completely our trial and error approach.

4.4 The Algorithm

STEP 0: Start with two identical "old" DMs

STEP 1 If the requirements are met, then the team is too good. Thus, the new DM can be worse than what he is, which implies that the K of the new DM can and should decrease. From our theoretical analysis we know that the new DM should be the consultant Thus, we decrease the consultant's K and go to STEP 3.

STEP 2: If the requirements are not met, then the team is too weak. Thus, the new DM should be better than what he is, which implies that the K of the new DM can and should increase. From our theoretical analysis we know that the new DM should be the primary. Thus, we decrease the primary's K and go to *STEP 4.*

STEP 3: If all the requirements are met and one is met exactly, we stop. If the requirements are met then we decrease the K of the consultant. If the requirements are not met we increase the K of the consultant. We then repeat STEP 3.

STEP 4. If all the requirements are met and one is met exactly, we stop. If the requirements are met then we decrease the K of the primary. If the requirements are not met we increase the K of the primary. We then repeat STEP 4.

Our theoretical analysis indicated whether the new DM should be the primary or the consultant. Using educated choises for the values of K in our trial and error approach our problem will be solved efficiently.

5. CONCLUSIONS

By a counterexample we have shown that the optimal team architecture may depend on parameters external to the team (prior probabilities, cost structure etc). Hence, we can have ambiguity of whether a particular architecture is optimal for all values of the external parameters. It is possible, however, to use the area under the team ROC curve to remove the ambiguity.

Special distributions lead to architectural comparisons that are unambiguous. We demonstrated this in the case of comparing gaussian variances, in which the better DM should always be the primary DM. Computer simulations (not reported here) indicated that this result holds true for comparisons of means of gaussian distributions, but the inherent complexity of the equations prohibited us from obtaining analytical results.

Even if the individual DM ROC curves are analytical, the team ROC curve is not. Thus, it is hard to generalize our results to teams with more than two DMs. We hope to obtain some novel results to help us design more complex organizations; but, it is not clear whether such results exist.

Finally, we plan to study the effects on the team performance of different communication protocols as well as of more complex (non-binary) hypotheses.

ACKNOWLEDGMENT

This research was supported by the Distributed Tactical Decision Making program of the Office of Naval Research under contract ONR/N00014-85-K-0329 (NR 649-003).

REFERENCES

[1] Van Trees,H.L.,"Detection, Estimation and Modulation Theory, vol. I" New York, Wiley, 1969

[2] Tenney ,R.R., and Sandell, N.R., "Detection with Distributed Sensors" IEEE Trans. of Aerospace and Electronic Systems, AES-17, July 1981, pp. 501-509

[3] Ekchian, L.K., "Optimal Design of Distributed Detection Networks" Ph.D. Dissertation, Dept. of Elec .Eng. and Computer Science Mass. Inst. of Technology, Cambridge, Mass., 1982

[4] Ekchian, L.K., and Tenney R.R., "Detection Networks" Proceedings of the 21st IEEE Conference on Decision and Control, 1982, pp. 686-691

[5] Kushner, H.J., and Pacut, A., "A Simulation Study of a Decentralized Detection Problem" IEEE Trans. on Automatic Control, 27, Oct. 1982, pp. 1116-1119

[6] Chair, Z., and Varshney, P.K., "Optimal Data Fusion in Multiple Sensor Detection Systems" IEEE Trans.on Aerospace and Electronic Systems, 21 January 1986, pp. 98-101

[7] Tsitsiklis, J.N., "Problems in Decentralized Decision Making and Computation" Ph.D. Dissertation, Dept. of Elec. Eng. and Computer Science Mass. Inst. of Technology, Cambridge, Mass , 1984

[8] Tsitsiklis J.N., and Athans, M., "On the Complexity of Decentralized Decision Making and Detection Problems" IEEE Trans. on Automatic Control, 30, May 1985, pp. 440-446

[9] Tsitsiklis J.N., "Decentralized Detection by a Large Number of Sensors" Mathematics of Control, Signals and Systems, Vol.1, #2, 1988

[10] Pearl J., "Fusion Propagation, and Structuring in Bayesian Networks" Technical Report, Computer Science Dept. Univ. of Calif. at Los Angeles, Ca. 90024

[11] Papastavrou J.D., "Distributed Detection with Selective Communications" M.S. Dissertation, Dept. of Elec. Eng. and Computer Science Mass. Inst. of Technology, Cambridge. Mass., 1986

[12] Thomopoulos S.C.A., Viswanathan R., and Bougoulias D.C., "Optimal Decision Fusion in Multiple Sensor Systems" IEEE Trans.on Aerospace and Electronic Systems, 23 September 1987, pp. 644-653

[13] Reibman A.R., anb Nolte L.W., "Optimal Detection and Performance of Distributed Sensor Systems" IEEE Trans.on Aerospace and Electronic Systems, 23 January 1987, pp. 24-30

[14] Reibman A.R., anb Nolte L.W., "Design and Performance Comparisons of Distributed Detection Networks" IEEE Trans.on Aerospace and Electronic Systems, 23 November 1987, pp. 789-797

THE DYNAMICS OF A HUMAN DECISION MAKER IN A SUDDENLY CHANGING ENVIRONMENT

G.A. Clapp

Naval Oceans Systems Center
Code 808
San Diego, CA 92152-5000

K.S. Haaland D.D. Sworder

Department of AMES
University of California, San Diego
La Jolla, CA 92093

ABSTRACT

In most C3 architectures, the human decision maker has a central position. He is frequently given a role in which he is required to bring some semblance of order to a rapidly evolving encounter which involves groups of both hostile and friendly vehicles. On the basis of his understanding of the implications of different events, he must make rational judgments on resource allocation and strategy.

This paper presents a study of a model which portrays the response of a person given a situation-assessment task. The model is phrased in terms of stochastic differential equations in order that it be compatible with the usual modelling paradigm for describing dynamically varying encounters. This normative-descriptive model permits the intimate interrelation of the operator and the rest of the C3 subsystems to be expressed.

The measured response of a test subject has been compared with the sample response of the model. A time varying scene was presented to a subject who was asked to indicate the level of confidence that a target was within the field of view. This response, along with the actual locations of targets and decoys, was recorded and compared with the model response to the same scenario. The model response did not reproduce the empirical behavior pattern any more than two humans would respond identically to the same stimuli. Nevertheless, the model does manifest the "human" indecisiveness found in the experiment.

1. INTRODUCTION

A human has a unique talent for making decisions and allocating resources in a dynamically varying encounter which is subject to significant uncertainty. For example, if he is able to view the field of action, he can distinguish the highest priority target in a cluster of like objects even in an environment permeated with a high level of structured clutter. Additionally, he is able to bring some semblance of order to observations of the motion of interacting groups of hostile and friendly vehicles. On the basis of his understanding of the implications of the perceived events, he makes rational judgments on suitable strategies. Indeed, the ubiquitous role of the human in BM/C3 testifies to his distinctive aptitudes.

Notwithstanding this general sentiment, the precise delineation of human tasks is a controversial theme. Indeed, the potential role of algorithmic surrogates has expanded as sensor and computer technology have advanced. In many analyses of C3 systems, the human is not thought of as an integral part of the system architecture, but is instead given an external position as a "user" of data or an "input" to the rest of the system. This extrinsic view of the human has led some investigators to relegate human activities to a very limited domain.

While all military BM/C3 systems share numerous generic similarities, different applications tend to emphasize the importance of certain aspects of the system design problem. To see why this is so, some insight into the basic issues which guide system design is required. The functions of Battle Management (BM) and Command, Control and Communications (C3) are performed hierarchically at many levels of military command and operations. These levels range from individual weapons platforms at the lowest to the Command Authorities (CA) at the highest. The functions are carried out through organizational structures and procedures spanning the military force structure. Implementation is accomplished by the incorporation of equipment, algorithms and communications into various categories and levels of subsystems which range from space-based sensors to field radios.

Even when the particular needs of a specific application are studied, virtually the entire domain of system components and locations must be considered. Consider, for example, the BM/C3 functions required for the operation of an individual weapons platform. Data from multiple sensors, both on board and remote, must be processed in concert with weapon status information to determine whether and where to launch interceptors. More generally, a panoramic view of BM/C3 includes not only the individual subsystems, but also the communication links and the support that they provide in supplying the requisite information to the control and management centers.

In the global system there are numerous interactions to be considered and trade-offs to be made in the allocation of BM/C3 functions to the different levels within the hierarchy and to different nodes within each level. This allocation is made on the basis of considerations of cost, reduced sensitivity to countermeasures, and uncertainties in the way that the encounter will evolve. A flexible architecture is essential to permit a rational maturation through the inclusion of advanced subsystems and components.

The utilization of human decision makers does not alter this fundamental view of BM/C3. Many of the BM/C3 functions can be performed either by a human or by an algorithmic surrogate. Clearly, any task involving the expeditious processing of large quantities of data is of necessity outside the human purview. A judicious aggregation of the data, however, can enable a human to adroitly make those critical decisions upon which the success of the C3 system depends. The making of changes in rules of engagement, and the placement of the system on alert status are inherently human decisions that must be referred to the CA acting in a direct management role.

To better understand the role of BM/C3 a broad description is useful. An epigrammatical definition is that BM/C3 is the mapping of assets into capabilities. This definition emphasizes the fact that BM/C3 is manifest in the system architecture. The arguments on the extent of BM/C3 (does it comprise everything from detectors to debris, or is it confined to the decision making environment of the CA) can then be seen to be disputations about the domain and range of the mapping, rather than about the fundamental character of BM/C3.

A recent workshop was held to explore the generic problems encountered in civil and military systems which share many features with C3, e.g., a wide geographical dispersal of activities, high communication data rates, uninterrupted operation for extended periods of time, etc [1] Not surprisingly, civil systems do share a range of challenges with military BM/C3 systems, but the range of permissible reactions to these challenges is much less restricted in the civilian setting. For example, the American Airlines SABRE system has a brittle architecture, and has had to be completely shut down for brief periods in the past. It has evolved over the years that it has been in service, but this evolution has tended to be based upon an ad hoc response to new needs or unexpected predicaments. Such a behavioral characteristic is unacceptable in the context of the most military C3 systems. Such systems will remain in place for a long period. A purposefully open architecture which permits the exploitation of advances in technology in sensors, communications, and processors is essential if the system is not to be outmoded long before its design lifetime.

In the presentations made at the workshop, it was observed that the function of the human decision maker kept reappearing in various contexts. The need for a human presence is a controversial theme in discussions of complex systems, both military and civil. For example, the SABRE system is under the direct control of human dispatchers, and they are in turn supervised by a "super" dispatcher, thus creating a classical hierarchical management structure which involves both human supervision and algorithmic data manipulation. Similarly, the space shuttle is under human control except for phases of ascent and descent In NASA, there is such a lack of trust in the ability of computers to automatically detect off nominal operating conditions, that computers are not permitted to turn off other computers. This can only be done by a person with the responsibility to make such decisions.

The ambiguous role of the human manifested itself again in the NORAD system. The principal operating decisions are all made by humans working individually or in teams Although the anecdotal evidence suggests that some of the most untoward system errors have been caused by misguided human intervention, there is no indication of any intent to reduce the human presence in the future. Instead, the proposals from both NASA and NORAD would increase the human presence by providing aids that would enhance his effectiveness.

Rigorous reasons for this fixation on giving a human an on-line role are frequently difficult to articulate. One might infer from this that the justification rests upon simple anthropocentric apologetics. Indeed, in many discussions of complex engineering systems the human is given an external position. This extrinsic view of the human has led some investigators to relegate the human to a very limited position in one of the major C3 systems of current interest, the SDI system In [2], for example, it is suggested in a discussion of the SDS that "the tactical system will have to operate in an automated mode simply because there is no time for humans to evaluate the huge amounts of sensor information and to arrive at superior weapon engagement in the short time available."

This circumscribed place of the human is unsatisfactory in the SDI mission and in other C3 systems as well. There has been a consensus from the beginning that human involvement in strategic defense decisions is essential. Admittedly, there are significant practical limits on the role that can be played by humans. The basic issue is how to define and support the role of human decision making, given the characteristics and limitations of the human information processing capabilities. If meaningful human roles are to be established in a C3 system, early incorporation of human factors analysis is indispensible.

It would appear that any precisely defined tasks given to a human could be performed automatically by an algorithm which mimics his input-output relationship. In truth, such algorithmic emulation has proven to be unsatisfactory in many respects. To clearly display the unique contribution that can be played by the human, a clear understanding of his areas of proficiency is essential To provide a description of operator behavior that will aid in defining the appropriate human role in the system architecture, it is essential to have a behavioral model which is both relatively simple and compatible with the other constituents of the encounter.

Analytical models of human response have a long history, with careful development beginning in the 1940's More recently, as the tasks originally assigned to the human have come to overlap more with those of the inanimate elements of the system, humans have assumed more complex roles and increasingly sophisticated models have been required In [3] Johannsen and Rouse propose a framework within which human activities can be organized. Their hierarchical perspective is amenable to a quantitative computer-like interpretation of human functions, but at the same time accounts for higher level psychological and intellectual activities such as reflection and planning. At the lowest level of activity, the operator behaves in an essentially automatic way. Indeed, in highly trained operators, proper behaviors, once learned, become reflexive and are probably performed at the level of the cerebellum. Johannsen and Rouse point out that the events which elicit these activities tend to occur relatively frequently and the response becomes instinctive.

Rasumssen [4] continued the trend toward a hierarchical representation of human control and decision making behaviors The lowest level of activity, which Rasmussen refers to as "skill based," occurs when the operator is quite familiar with the environment in which the encounter takes place Thus, an experienced driver smoothly and automatically adjusts his steering and acceleration to maintain a desired position on the road Even the abrupt appearance of a obstacle in the path elicits a reflexive response.

When concern centers on such composite problems as tracking of a dynamic target in clutter, the Optimal Control Model (OCM) of Baron, Kleinman and Levison has proven quite useful (see [5] for a clear description of this approach along with numerous references) Normative descriptive models of which the OCM is a notable example have as their rationale the fact that "the motivated expert decisionmakers strive for optimality but are constrained from achieving it by inherent human perceptual limitations and cognitive biases.[6] When applied to situations in which actions must be taken in response to an evolving encounter, the OCM and its more recent counterparts have been phrased within the Stimulus/Hypothesis/Options/Response (SHOR) paradigm of human decision making (see [7] and a recent example of its application [8]).

In this paper, the relationship between stimulus and hypothesis evaluation is explored in some detail within the context recognition and decision making tasks that are encountered in C3 applications. In [9] Wohl, et al. discuss some of the generic issues which must be addressed in the formulation of the Stimulus and Hypothesis Evaluation portion of the operator model. Paraphrasing these themes in a prototypical C3 task, reconnaissance and situation evaluation, it is evident that the decision maker may encounter a variety of different eventualities as he monitors a dynamic encounter. There may be important targets of various types, decoys or target-like objects of little significance, open areas containing nothing of interest, etc. The operator classifies these alternatives as different hypotheses, and makes a choice between them on the basis of his observations. If this decision is made contingent on the motion patterns of the observed objects, then the situation assessment block in the operator model becomes a bank of Kalman filters tuned to the various dynamic hypotheses, along with a suitable combination of the outputs to generate the conditional likelihoods of the various hypotheses (this is shown in Figure 3-10 of [8]).

The application which motivated this work involved a study of a system built around the Naval Oceans Systems Center (NOSC) teleoperated vehicle (TOV) (see Figure 1.1). In this system a remote operator is able to project his presence into a hostile environment through the intermediary of an anthropomorphic robot (Figure 1.2). The remote operator becomes the central focus in this C3 architecture, and a quantification of his action pattern is essential in determining the adequacy of the system. In this application as in most C3 systems, the hypotheses which define the evolution of the encounter are more clearly distinguished by their panoramic features than they are by their local motion attributes. For example, a tank (high priority) may maneuver in concert with other vehicles of less worth. The remote operator will tend to identify the relevant object in the field of view on the basis of its visual signature rather than by its motion pattern.

Figure 1.1 The NOSC Teleoperated Vehicle

In this paper, a recognition model somewhat different than that found so useful in the indicated references is proposed. It is a nonlinear stochastic differential equation which evidences the uncertainty that an operator faces in recognizing the relevant features in a changing scene containing a high level of visual clutter. A test which compares the response of the model with that of a human subject suggests that "human" indecisiveness is well captured by the model.

Figure 1.2 The NOSC Teleoperated Robot

2. DECISION MAKER RESPONSE MODELLING

Since reconnaissance and situation evaluation are important roles for a decision maker in C3, it is useful to be able to describe an operator's proficiency in situation recognition. The ability to infer the mode of evolution of an encounter requires both a panoramic view and an ability to place the observations within a well defined pattern. The human role becomes preeminent when the encounter involves sudden and unpredictable changes in the operational environment. The appearance of a target, the change in aspect of that target in a manner which indicates a threat to decision maker's resources, an abrupt change in the terrain over which the encounter takes place are all possible events that the decision maker may experience and to which he must respond. These are represented by a distinct set of hypotheses in [8]. In contrast to the structure developed in that reference, the currently realized hypothesis may change in time; i.e., the target may become obscured, the terrain may change again.

Suppose that the operative hypothesis is denoted by an integer valued "feature" of the encounter. Indicate the feature process by $\{r_t\}$ where the state space of r_t is $S=\{1,\ldots,s\}$; i.e., there are s different possible features and r_t is an indicator of the current one. A specific example of this structure which will be discussed in more detail in the sequel can be described as follows. Suppose that the decision maker is scanning a region looking for high priority targets. Within the field of view there is either (1) a target, (2) a target like decoy or (3) open terrain. The appropriate options for the decision maker are contingent on the realized feature. The most important event is that a target is within the operator's field of view, and this is indicated by $r_t=1$. The events $r_t=2$ or 3 are interpreted in an analogous fashion and $S=\{1,2,3\}$

For calculations, it is expedient to introduce an alternative notation for the feature process. If $r_t=i$, let $\phi_t=e_i$.[2] Then $\{\phi_t\}$ is a unit vector which indicates the current feature. Assume that $\{r_t\}$ is a Markov process with transition matrix $Q=[q_{ij}]$

$$Pr(r_{t+\delta}=j|r_t=i)=\begin{cases}1+q_{ii}\delta+o(\delta);\ i=j\\ q_{ij}\delta+o(\delta);\ i\neq j\end{cases} \tag{2.1}$$

Then $\{\phi_t\}$ satisfies a stochastic differential equation

$$d\phi_t=Q'\phi_t dt+dm_t \tag{2.2}$$

where $\{m_t\}$ is a purely discontinuous martingale with respect to the natural filtration.

Equation (2.2) is a simple model of the feature process, but it will suffice for this introductory study. Feature variation manifests itself in panoramic changes in sensory data, and the operator attempts to identify feature changes from stimuli provided by the suite of sensors available to him. Such observations may be quite cluttered, and it is difficult to eliminate the ambiguity which naturally surrounds the interpretation of the scene. To be specific, suppose that each feature has a distinct sensory signature. In keeping with the referenced analytical models of human response, this signature has little to do with the actual physiological processes in the human, but is rather a pseudo-stimulus used as the input in the normative-descriptive model of operator response. Let the signature of feature i be given by the real number h_i. The operator receives a cluttered measurement, $\{y_t\}$, of the current feature. This measurement will be represented by the stochastic differential equation

$$dy_t=h'\phi_t dt+dn_t \tag{2.3}$$

where h is the indicated s-vector, and $\{n_t\}$ is Brownian motion

$$(dn_t)^2=dt \tag{2.4}$$

Equation (2.3) relates the operator's observation to a signal $h'\phi_t$ generated by the feature. This modal indication is contaminated by an exogenous wide-band clutter. On the basis of $\{y_t\}$, the decision maker infers the likelihood of the various possible features. This structure fits well with the SHOR paradigm, but since the motions give little information, the structure of the hypothesis estimator is significantly different.

Let $\{Y_t\}$ be the natural filtration associated with $\{y_t\}$, and denote the conditional expectation with respect to $\{Y_t\}$ by "^". Then the operator's conception of the relative probabilities of different features is given by the "information state" $\{\hat{\phi}_t\}$

$$\{\hat{\phi}_t\}=E\{\phi_t|Y_t\} \tag{2.5}$$
$$=[Pr(\phi_t=e_i|Y_t)]$$

The equations of evolution of $\{\hat{\phi}_t\}$ are given in [10], and they represent the first portion of the Decision Directed Model (DDM) used to describe the response of a remote operator engaged in directing the TOV in a multitask environment [11]

$$d\hat{\phi}_t=Q'\hat{\phi}_t dt+(\mathrm{diag}\, h-\hat{\phi}_t' h I)\hat{\phi}_t\, dv_t \tag{2.6}$$

where $\{v_t\}$ is the innovations process

$$dv_t=dy_t-h'\hat{\phi}_t dt \tag{2.7}$$

Equation (2.6) is a nonlinear stochastic differential equation. It quantifies the way in which the information state changes as the operator processes new observations. The components of $\{\hat{\phi}_t\}$ are the conditional likelihoods of the various hypotheses concerning the encounter. It is on the basis of these quantities that strategy is formulated and resources allocated.

Equation (2.6) is an Ito equation, and considerable care must be exercised in its interpretation. In Section 3 the flexibility of the operator model is explored. Different "personalities" are created as model parameters are changed. The effect that the rate of change in the encounter, the feature discernibility, and the level of exogenous clutter have on hypothesis evaluation are exhibited in the sample behavior of the DDM.

In Section 4, the measured response characteristics of an operator is compared with the simulated behavior of the model. A time varying scene was presented to a subject who was asked to indicate a level of confidence that a target (a tank) was within the field of view. This response along with the actual locations of targets, decoys and open regions was recorded. A sample function of the information state $\{\hat{\phi}_t\}$ generated from (2.6) for the same feature sequence was also recorded. For different sample functions of the observation noise, the operator's perception of the scenario features will change. As a consequence, $\{\hat{\phi}_t\}$ is a random process even for a predetermined scenario $\{\phi_t\}$. Thus one would not expect the model to reproduce a specific human response any more than a human would be unaffected by the realization of the exogenous visual clutter, or indeed that two humans would respond identically to the same stimuli. With this caveat, (2.6) does indicate an appropriate indecisiveness in identifying feature changes.

3. THE PLIANCY OF THE DDM

The model given in (2.6) is intended to represent the behavior of a decision maker in a variety of different situations, and to do this it must posses the "human" quality of formability. Thus, when the scenario is slowly varying and the clutter level is low, the DDM should respond surely and expeditiously. Alternatively, when the features which distinguish different hypotheses lack contrast, and when there is considerable clutter, the DDM may be confused and tepid in its response.

The operator model has two multidimensional parameters, and each is related to a different aspect of the encounter. The rate at which the underlying features of the scenario change is quantified by Q. If hypothesis i has a short mean lifetime, then q_{ii} is large. If there are targets of different classes in close proximity, then the transition rates q_{ij} would be large for appropriate i,j and so on. The correct specification of Q is dependent upon the dynamic structure of the encounter, and it parameterizes the drift term in (2.6); i.e.

$$E\{d\phi_t|Y_t\}=Q'\hat{\phi}_t dt \tag{3.1}$$

In the same way that Q quantifies the encounter dynamics, h quantifies both the sensory acuity and the clutter. Equation (2.3) is phrased in terms of a normalized observation signal $\{y_t\}$ with the noise term $\{n_t\}$ having unit intensity (see (2.4)). The scale of h is then contingent on the actual intensity of the wide band clutter in the sense that if the exogenous disturbance is increased by a power factor of four, this same effect can be induced into (2.3) by mapping h-->h/2.

To study the flexibility of the DDM it is expedient to generalize (2.3)-(2.4) slightly, even though this means over parameterizing the system. Thus, instead of (2.4), suppose that

$$(dn_t)^2=r\, dt \tag{3.2}$$

Then r is a direct indicator of the intensity of the observation noise. Although redundant, (3.2) permits the effects of varying clutter levels to be separated from variations in the distinguishability of the various modal hypotheses. The equation for $\{\hat{\phi}_t\}$ must be modified in an obvious way, and this has been done in what follows. It was further noted that the numerical stability of (2.6) was inadequate for the simulation task. A change of

[2] Denote the unit vector in the i'th direction in R^s by e_i.

variables transforms (2.6) into the Zakai form which is a linear stochastic differential equation for an unnormalized version of $\{\phi_t\}$. This equation proved to be superior in this application, and the solutions were renormalized for presentation here.

As pointed out above, there are three basic constituents of an encounter, each with a natural parameterization in the DDM: encounter dynamics (Q), exogenous disturbance (r), and operator acuity (h). To illustrate the behavior of the model as a function of the observation noise, consider a situation in which there are only two hypotheses (S={1,2}) and h=(1,-1). Hence the observation structure is

$$dy_t = \begin{array}{ll} 1dt + \text{noise} & \text{if } r_t=1 \\ -1dt + \text{noise} & \text{if } r_t=2 \end{array} \quad (3.3)$$

The signature of hypothesis 1 is the constant 1, and if observed long enough the operator would deduce the correct response by averaging; i.e., $\phi_t \rightarrow e_1$. The operator response must, however, be expeditious and further, the modal variable may change anyway. Suppose that the scenario dynamics are given by

$$Q = \begin{bmatrix} -0.1 & 0.1 \\ 0.9 & -0.9 \end{bmatrix} \quad (3.4)$$

The mean sojourn in mode 1 is 10 sec (1/0.1) while the mean sojourn in mode 2 is 1.1 sec (1/0.9). Equation (3.4) implies that the system is reluctant to leave e_1, and the operator knows it.

The decision maker's response is clearly dependent upon the clutter. The sharp edges of modal transitions are masked by noise. The operator realizes that these changes are inevitable, but the DDM may have difficulty discerning actual transitions from anomalous variation in the observation noise. Figure 3.1 shows a sample of the output of the DDM when the scenario has a single jump to e_1,

$$\phi_t = \begin{array}{ll} e_1 & \text{when } t \in A \\ e_2 & \text{when } t \in A^c \end{array} \quad (3.5)$$

where A=[0.5,1.0). This interval is indicated in Figure 3.1 along with the DDM response.[3] The noise level is small (r=0.01), and the response is unequivocal. The delay in identifying an $e_2 \rightarrow e_1$ transition is negligible.[4] There are no significant false alarms, but the false indication of change are more evident during sojourns in e_2. This is a natural consequence of fact that the mean lifetime in e_2 will be far shorter than that in e_1.

As the noise is increased (r=0.1), hypotheses evaluation becomes more challenging (Figure 3.2). The delay in detecting a transition is now on the order of 0.1 sec. The volatility in mode 2 increases significantly. If the amplitude of the noise is increased still further (see Figure 3.3), the hypothesis evaluation block no longer effectively responds to the stimuli at all, and tends toward the stationary distribution for $\{\phi_t\}$ which would be obtained in the absence of observations.

Figures 3.1-3.3 give a quantitative indication of how an increase in exogenous clutter manifests itself in increased indecisiveness in the DDM. The DDM moves in a continuous fashion from assured response to capriciousness as r increases. These figures also indicate that the volatility during the midst of a sojourn time is a direct function of the expected lifetime of the associated feature in all but the highest noise case.

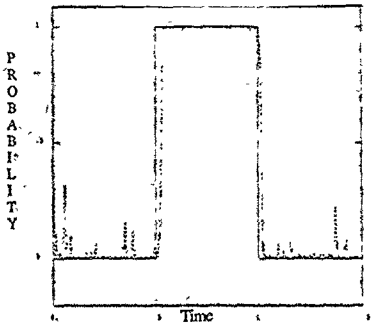


Figure 3.1 Sample function of $\hat{\phi}_1$ when the noise intensity is low (r = .01)

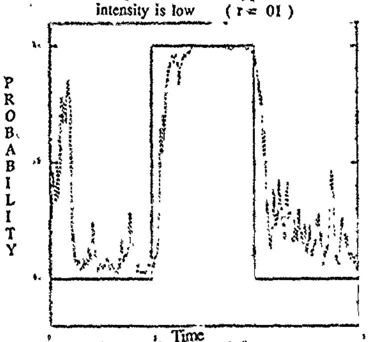


Figure 3.2 Sample function of $\hat{\phi}_1$ when the noise intensity is moderate (r = 0.1)

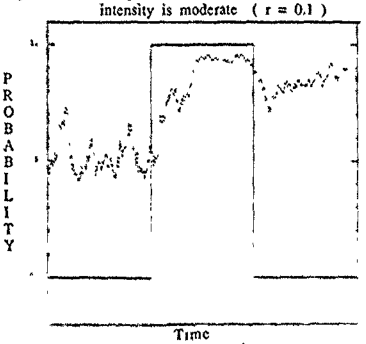


Figure 3.3 Sample function of $\hat{\phi}_1$ when the noise intensity is high (r = 1.0)

[3] A plot of $(\phi_t)_1$ is shown as the solid rectangle, and $(\hat{\phi}_t)_1$ is shown dotted

[4] A natural index for scaling clutter is the ratio of the size of the modal jump over the $\sqrt{r}$. In this example, this index would be 2/0.1=20.

To study the way in which encounter dynamics influence the personality of the DDM, consider Figures 3.2, 3.4, and 3.5. These figures show the response of the DDM to changes in Q with a fixed value of r (r=0.1) and h (h =(1,-1)). In addition to the Q matrix given in (3.4), two other matrices were studied

$$Q=\begin{bmatrix}-0.5 & 0.5\\ 0.9 & -0.9\end{bmatrix} \quad \text{Figure 3.4} \tag{3.6}$$

and

$$Q=\begin{bmatrix}-0.5 & 0.5\\ 5.0 & -5.0\end{bmatrix} \quad \text{Figure 3.5} \tag{3.7}$$

The behavior when $\phi_t=e_1$ does not vary radically with the changes in Q. In each case e_1 is the long lived mode, and the detection of e_1 is made expeditiously. There is an increase in uncertainty in detecting e_2 when the time scale of the problem is changed (compare Figs. 3.2 with 3.5 for a time scale variation of five). A short lived mode is difficult to detect in wide-band noise.

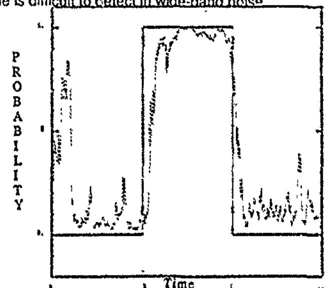


Figure 3.4 Sample function of $\hat{\phi}_1$ when the lifetime in mode 1 is shorter (τ_1 = 2.0 , τ_2 = 1.1)

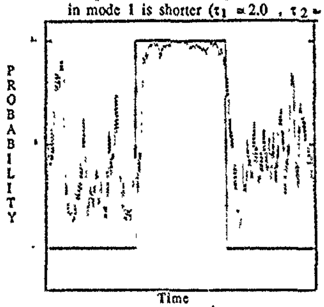


Figure 3.5 Sample function of $\hat{\phi}_1$ when the lifetime in mode 2 is shorter (τ_1 = 2.0 , τ_2 = 0.2)

The behavior of the DDM as image discernibility changes is of fundamental interest. Expensive operator enhancements are justified by an improved ability to differentiate hypotheses that lack definition in the image. Such sensor-processor modifications directly influence h, and through this intermediary they are reflected in the response of the model. To study hypothesis distinguish ability as distinct from simple changes in exogenous clutter, at least three distinct hypotheses must be considered, i.e., the dimension of ϕ_t must be greater than two. With two hypotheses, changes in h can be mimicked by changes in r.

To investigate separability of the hypotheses, the feature state was three dimensional and the scenario dynamics were fixed,[5]

$$Q=\begin{bmatrix}-1.0 & 0.2 & 0.8\\ 0.2 & -1.0 & 0.8\\ .25 & .25 & -0.5\end{bmatrix} \tag{3.8}$$

The noise had a constant intensity (r=0.01). Operator acuity is delineated by h. Figures 3.6 through 3.8 show the decision maker's response for different h-vectors

$$\begin{aligned} h = \ & (-1.0,0.0,1.0) \text{ in Fig. 3.6} \\ & (-1.0,-0.5,1.0) \text{ in Fig. 3.7} \\ & (-1.0,-0.9,1.0) \text{ in Fig. 3.8} \end{aligned} \tag{3.9}$$

In all of the cases indicated in (3.9), the features which define hypotheses e_1 and e_3 are clearly marked (h_1=-1.0 and h_3=1.0). Based upon the results presented in Fig. 3.1, these two alternatives are clearly distinguishable in clutter of the intensity indicated. It is the behavior of the DDM as a function of the signature of the second hypothesis e_2 (h_2) that is of most concern here. In Fig. 3.6, each of the hypotheses has an easily perceived marker. The scenario is that given by (3.4) with the renumbering of hypotheses e_2-->e_3. When an e_3-->e_1 transition takes place, both $\hat{\phi}_1$ and $\hat{\phi}_2$ increase.[6] The mean drift rate in (2.3) has changed from -1.0 to 1.0. The hypotheses evaluation block quickly recognizes that a change in drift has occurred (in approximately 0.25 sec.), but can't tell which of the alternatives is the correct one since both are identified with an increased drift. It takes another quarter of a second for the hypothesis e_2 to be discarded. After identifying e_1, the quiescent interval is relatively uneventful.

The e_1-->e_3 transition at 1.0 sec. is interesting because the DDM responds so expeditiously. This would not be expected on the basis of (3.7) since e_1 is not a particularly short lived state. The prompt response is due primarily to chance fluctuation in the exogenous noise. The sample function of $\{n_t\}$ is such that $\hat{\phi}_1$ has notably descended at t=0.96. This decrease is reflected in a corresponding increase in the likelihood that e_3 is the operative hypothesis. Even though this change was incorrect because the ostensible transition had not yet occurred, the DDM began to move in the direction of hypothesis 3, and was thus well situated for the actual event. Under these fortuitous circumstances, the DDM appears to be omniscient.

In Figure 3.7, the distinguishability of e_1 and e_2 has been reduced. The same broad outlines of the DDM response again manifest themselves, with the confusion surrounding the e_3-->e_1 transition increasing significantly. It now takes 0.3 sec. for the DDM to moderate. The same sample function for $\{n_t\}$ was used in all of these tests. Hence, the anomalous behavior near t=1.0 reoccurs throughout these simulations.

As the values of h_1 and h_2 become closer, the distinguishability of e_1 and e_2 decreases with easily predicted consequences. Figure 3.8 shows the change in the DDM brought about by this narrowing in the differentiation of the hypotheses. When $|h_1-h_2|<\sqrt{r}$ as it is in Fig. 3.8, the probability of correct detection of the transition to e_1 becomes too low to be acceptable even when the interval over which observation takes place is relatively long. Collaterally, the identification of a transition to e_3 becomes even faster when the DDM has not clearly identified the earlier transition.

[5] S={1,2,3} and e_i is in R^3.

[6] In Figs. 3.6-3.9, $\hat{\phi}_1$ is shown solid and $\hat{\phi}_2$ is shown dotted.

In this section a simple parametric study of the hypothesis evaluation block of the DDM has been presented. An indication of the behavioral flexibility of the DDM is apparent. The next section gives an indication of how this flexibility can be used to describe the response of a test subject.

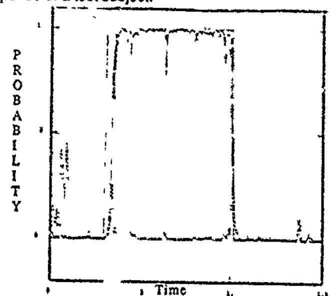


Figure 3.6 Sample function of $\hat{\phi}_1$ (solid) and $\hat{\phi}_2$ (dashed) when modal signatures are quite distinct ($h^T = [-1.0 \;\; 0.0 \;\; 1.0]$)

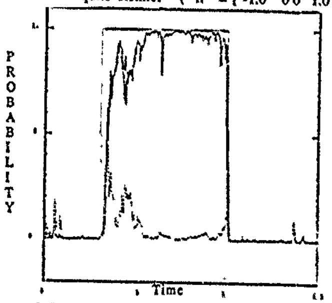


Figure 3.7 Sample function of $\hat{\phi}_1$ (solid) and $\hat{\phi}_2$ (dashed) when modal signatures are less distinct ($h^T = [-1.0 \;\; -0.5 \;\; 1.0]$)

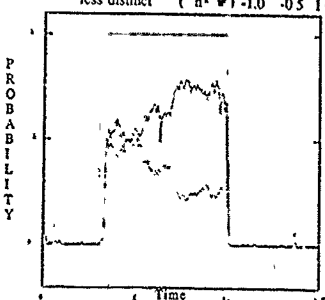


Figure 3.8 Sample function of $\hat{\phi}_1$ (solid) and $\hat{\phi}_2$ (dashed) when modal signatures are indistinct ($h^T = [-1.0 \;\; -0.9 \;\; 1.0]$)

4 OPERATOR RESPONSE CHARACTERISTICS

As stated earlier, the adequacy of the normative-descriptive model of a human decision maker must be established empirically. The previous section has indicated in some detail the performance attributes of the input-output model, (2.3)-(2.6). In the SHOR paradigm the decision maker responds to new data by modifying the likelihoods of the various hypotheses which delineate the encounter. The DDM provides an algorithmic description of the precise way in which this is done.

The hypothesis evaluation portion of the DDM is difficult to verify because this appraisal is an internal activity of the decision maker, and is normally reflected only indirectly by his actions. Nevertheless, to better understand the limitations of the DDM, a scene recognition task of the type to be encountered in the TOV application was given to a representative operator. The experimental facilities were rudimentary, and the operational environment of the TOV operator could not be reproduced. The experimental protocol can be described as follows. A set of photographs were taken at the Camp Pendleton Marine Base in California. They were appropriately juxtaposed to from a panoramic view of the terrain as seen from a fixed location. The scene consisted of rather open terrain containing scrub brush along with a scattering of discrete objects at various ranges. The actual object set contained jeeps, Land Vehicle, Tracked (LVTs), out buildings, tanks, and various natural structures similar in appearance to the foregoing. The vehicles had different aspect angles with respect to the viewer and had a variety of visual appearances.

To test the dynamic response of a decision maker, a single category of objects was made that of primary concern. In the experiment, the operator was asked to identify the presence of a tank in a changing scene, and to consider all other events as being inconsequential. At one level then the operator could be viewed as comparing a simple hypothesis (tank present) with a single alternative (tank absent). Actually both hypotheses are composite, with the latter containing many well defined subsidiary hypotheses. Hence, this simple structure with S={1,2} is not sufficiently rich to capture the scenario as the decision maker perceives it.

There was considerable visual ambiguity since other object classes share many features with a tank when viewed at a distance. Indeed, anything with an angular shape could be confused with the primary target class at first glance. It was necessary, therefore, to introduce a third operative hypothesis (secondary object present) to account for the decoys. Surprisingly, given the intricacy of the scene, further additions to S were not found to be necessary. It should be noted that the DDM permits augmentation to the modal set in a direct manner if the actions of the decision maker warrant such an increase in dimension.

To inject scene dynamics into what is fundamentally a static encounter, a movable camera was made to pan a horizontal slice in the picture at a fixed angular rate. A local image was displayed on a monitor for the operator to view. Thus, static objects appeared in the monitor at random times with random clarity but with duration fixed by the linear dimension. The scan rate thus determined the time scale of changes in the events to which the operator responded. A frequent modal change is achieved at a high scan rate. At low scan rates, the operator has more time to contemplate and distinguish the relevant objects from clutter.

The subject was given a joy stick with one degree of freedom, and was asked to position the rod to indicate her level of confidence that a target was contained in the image displayed on the monitor. The output voltage from the joy stick was proportional to the angular position with the lowest voltage representing certainty that there was not a target within the field of view, and maximum voltage representing certainty that a tank was displayed on the monitor. An intermediate position was appropriate when the scene was ambiguous.

Figure 4.1 displays a sample function of the subject's response. As mentioned earlier, the event sequence can be viewed as consisting of an alternation between distinct modal hypotheses. The actual time intervals during which a target is on the monitor are indicated in the figure by the dotted rectangles. Let the event that a tank is being observed by r_t=1. Decoys are similarly distinguished by the dashed rectangles with the corresponding event denoted by r_t=2. The remainder of the interval consisted of open fields and unstructured clutter, and this is denoted by r_t=3.

From a cursory review of the picture, the broad outlines of the decision maker's behavioral peculiarities shown in Figure 4.1 are easily predicted. During sojourns in regions of relatively open terrain, there is little operator confusion. The output of the joy stick should be essentially zero during such phases.

As the picture was scanned, there were regions in which isolated and distinguishable objects were encountered, and the subject attempted to identify them. The scanning direction is such that an object enters the subject's visual field from the right of the screen. When an object appears on the monitor, the operator will initially be uncertain as to the object classification. If the object has a distinct, angular shape that separates it from the more diffuse background, the operator will realize that a modal change has taken place but may delay making a confident identification. The subject indicates this confusion by moving the rod to an intermediate position which is roughly proportional to the probability that the object is in the primary category. In an object sparse region, the operator will concentrate on an unusual shape as it moves across the screen, and make more definitive judgments on the proper classification. Some of the things in the scene are quite deceptive. Careful study is required to distinguish these decoys from targets. The presence of these "near targets" produces significant false alarms.

In another region in the picture, a group of similar objects are in close proximity. While there is never a situation in which more than one object is within the field of view of the camera, when objects occur in rapid succession, the observer never has the time to focus clearly on the object at hand. Instead, the decision maker becomes preoccupied with the sequence itself, and lapses into an unsettled state in which the objects are not clearly differentiated.

In Figure 4.1 the operator response is shown along with the regions of target and decoys. The operator response is related to $\{a_t\}$ in the notation of (2.6). The anticipated peculiarities of the decision maker's response characteristics appear in the sample function. Open areas are clearly recognized as such for the most part, although some natural objects do cause confusion at certain light angles. What is more interesting is the response of the operator to targets and putative targets. Predictably, the operator has much more difficulty in differentiating the hypotheses when the objects appear more frequently than when they are separated by extended intervals.

A simulated response of the hypothesis evaluation portion of the DDM is shown in Figure 4.2. The most notable difference between the simulated and the actual response is that the simulated sample function is locally much more volatile than is the human. This is not surprising. Equation (2.6) is an Ito equation, and solutions to such equations are quite irregular on any short time scale. The decision maker's evaluation of a situation is probably not nearly so changeable, and even if it were, the physiological lags which occur in translating a mental view of a scene into a shaft position would smooth the volatility and preclude its measurement. Furthermore, a pointing system under the direction of the operator has a low pass character as well. Hence, the local volatility is not an important issue in operator modeling.

With this caveat, the response of the DDM has the "human" peculiarities that were predicted and measured. This similarity of response manifests itself clearly in the object rich portion of the scan. The rapid modal variation is represented in the DDM by a Q matrix with high modal transition rates. As the modal variation becomes more frequent, the response of the DDM becomes more indecisive.

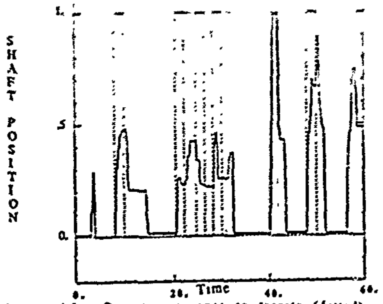


Figure 4.1 Operator response to targets (dotted) and decoys (dashed)

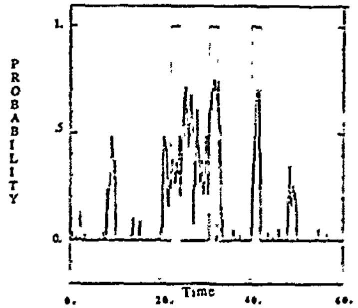


Figure 4.2 Sample function for the test scenario

5. CONCLUSIONS

A human decision maker frequently plays a central role in a C3 architecture. Unfortunately it is difficult to delineate his characteristics in an analytically tractable manner. He is capable of so many different behaviors that it is hard to capture his persona in an abstract form. This paper considers one aspect of his response characteristics, and indicates the capability and versatility of the hypothesis evaluation portion of the DDM. Some very human peculiarities have been observed in tests utilizing the model. The indecisiveness and uncertainty under which the decision maker must perform his assigned tasks manifests itself in the simulation study. As has been pointed out, the proper response of the DDM is dependent on the correct selection of the model parameters. The simulations indicate that considerable freedom exists in shaping the response characteristics. Indecisiveness in a rapidly changing scenario can be produced by increasing the size of the elements in the Q matrix. If two or more hypotheses are difficult to differentiate, the associated elements of the h vector should be made close.

Situation evaluation is only one component of the operator's function in the C3 system. The complete DDM requires a characterization of the Options/Response portion of the model. This block translates the vector of likelihoods provided by the hypothesis evaluation block into an appropriate action. This block was studied in [11], but was incompletely delineated. This is a topic of current study.

REFERENCES

1. Report of the Organizing Committee of the Innovative Science and Technology (IST) Workshop in SDI/BM/C3, IDA Memorandum Rept. Jan.1988.

2. M. Athans, "Command and Control (C2) Theory: A Challenge to Control Science," IEEE Trans. on Automatic Control, Vol. AC-32, April 1987, PP. 286-293.

3. G. Johannsen and W. Rouse, "Prospects on a Mathematical Theory of Human Behavior in Complex Man-Machine System Tasks," 14th Conf. on Manual Control, 1978.

4. J. Rasmussen, "The Human as a Systems Component," in Human Interaction With Computers, Ed. H.T. Smith and T.R. Green, Academic Press, 1980.

5. S. Baron, "A Control Theoretic Approach to Modeling the Human Supervisory Control of Dynamic Systems," in Advances in Man Machine Systems, Ed. W.B. Rouse, Vol. 1, 1984.

6. D. Serfaty, L.G. Bushnell, and D.L. Kleinman, "Distributed Human Information Processing: A Normative-Descriptive Approach," Proc. of the Fifth Symposium on Empirical Foundations of Information and Software Science, Roskilde, Denmark, Nov. 1987.

7. J.G. Wohl, "Force Management Requirements for Air Force Tactical Command and Control," IEEE Trans. on Systems, Man, and Cybernetics, Vol. SMC-11, Sept. 1981, pp.618-639.

8. J.G. Wohl, et.al. "Human Cognitive Performance in ASW Data Fusion," Proc. of the First Tri-Service Data Fusion Symposium, John Hopkins University, June 1987.

9. J.G. Wohl, E.E. Entrin and J.S. Eterno, "Modeling Human Decision Processes in Command and Control," Alphatech Tech. Rept. TR-137, Jan. 1983.

10. R.J. Elliott, "Filtering and Control for Point Process Observations," Rept. No. 9.86.1, Dept of Applied Probability and Statistics, Univ. of Alberta, Sept. 1986.

11. D.D. Sworder, and K.S. Haaland, "Algorithms for the Design of Teleoperated Systems," Control and Dynamic Systems, Advances in Theory and Applications, Vol. [illegible] Academic Press, 1988.

APPROXIMATE REASONING IN NAVAL COMMAND AND CONTROL

R. Pratila

Space and Naval Warfare Systems Command
Washington, D.C. 20363-5100

ABSTRACT

Good information structures imply a proper framework for analysis. In the classical method of analysis one must elicit numerical inputs of probability and utility with a high degree of precision. This is often perceived as a disadvantage. Through approximate reasoning the need for precise inputs may be relaxed. The deduction of imprecise, yet meaningful, conclusions can be made from a set of imprecise premises. Such reasoning is neither very exact nor very inexact. It can; however, play a basic role in decision making by providing a way of dealing with complex problems. This role is examined in a simple example where one of two possible decisions is to be made.

SCENARIO

Consider a Naval force preparing for an air strike against a land target, which is a hostile cruise missile site blocking a friendly strategic sea lane. The site is embedded within a U-shaped mountain range. The elevation and ruggedness of the mountains provide the missile site a natural protective barrier on three sides. The mountains open to a rugged, arid plain strewn with large boulders and rock formations.

Defenses located on the plain are estimated as moderate to heavy. At least 5 well-placed surface-to-air missile sites are known to exist. There may be more. The defenses located in the mountains are estimated, with much uncertainty, to be light, i.e., they consist of possibly 2 missile sites.

The mission planners decide that for the mission to be successful 75% or more of the target must be destroyed. The strike will involve 33 attack aircraft, each equipped with 4 air-to-surface missiles. By approaching the target from over the mountains there is a good chance of avoiding defenses located on the plain and encountering only the light defenses in the mountains; losses could be light. However, there is uncertainty about the success of the mission. The target will be obscured for all possible attack angles; direct hits will be difficult to achieve. By approaching the target via the plain or range opening there is a good chance of encountering moderate to heavy defenses. Aircraft losses could be moderate to heavy. The probability of destroying the target is better since the attack angles pose a lesser visibility problem to the pilots. The question to be resolved by the mission planners is, WHICH ROUTE OFFERS THE BEST CHANCE OF MISSION SUCCESS AND FEWEST LOSSES?

FUZZY DECISION ANALYSIS

We make use of 3 propositions from the calculus of fuzzy sets. They are:

(a) The degree to which element a belongs to set A and element b belongs to set B is equal to the smaller of the individual degrees of membership.

$$u_{A \wedge B}(a,b)=\min[u_A(a),u_B(b)] \quad (1)$$

(b) The degree to which element a belongs to set A or element b belongs to set B is equal to the larger of the individual degrees of membership.

$$u_{A \vee B}(a,b)=\max[u_A(a),u_B(b)] \quad (2)$$

(c) The degree to which a belongs to not A is 1 minus the degree to which a belongs to A.

$$u_{-A}(a)=1-u_A(a) \quad (3)$$

The above relations are used to deduce the fuzziness on an output given the fuzziness on an input. In general, if u(y) is the degree to which y belongs to the set of possible outputs and $u(x_i)$ is the degree to which x_i belongs to the possible set of inputs, then by (1) and (2)

$$u(y)=\max_{y=f(x)}\{\min[u_1(x_1),u_2(x_2,\ldots,u_n(d_n)]\} \quad (4)$$

The bracketed expression in (4) is the extent to which the set of inputs x belongs to the possible set of such inputs. It is only as large as the smallest of the separate degrees of membership of the components of x. Since there may be a different degree of membership for each x satisfying f(x)=y, the degree to which y belongs to the possible set of output values is the maximum degree of membership of x over all such x.

Shown in Figures 1 and 2 are two decision analysis diagrams which support the mission scenario described above. Figure 1 depicts the classical method of analysis, which subscribes to precise estimates of probability and utility. Figure 2 exemplifies the fuzzy set method of analysis. Probabilities are reasoned as small chance, good chance, etc. Utilities are reasoned as heavy losses, mission failure, etc.

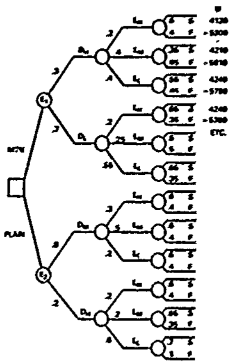


Fig.1.

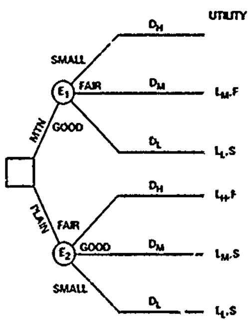


Fig.2.

We begin the fuzzy set analysis by quantifying the descriptors of Figure 2. Membership functions for small chance $u_P(p_S)$ and good chance $u_P(p_G)$ are reasoned as shown in Figure 3. By (3) fair chance $u_P(p_F)$ is,

$$u_P(p_F)=1-[u_P(p_S)+u_P(p_G)].$$

Similarly, the membership functions for losses L_H, L_M, L_L and success S are estimated as shown in Figures 4 and 5, respectively.

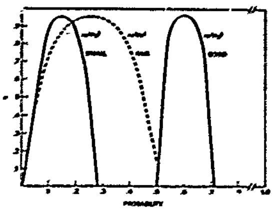


Fig.3.

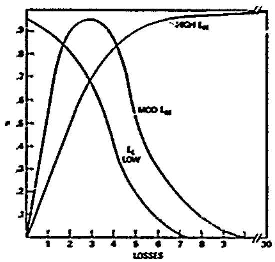


Fig.4.

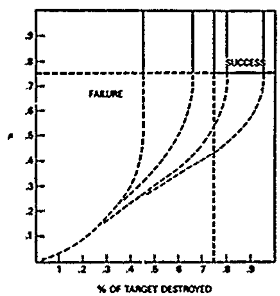


Fig.5.

Using (4) the membership functions for the expected values E_1 and E_2 are computed in terms of aircraft losses and mission success. For inputs L_H, L_M and L_L we have,

$$u(e)=\max\{\min[u_P(p_S),u_P(p_G),u_H(l_H),u_M(l_M),u_L(l_L)]\}$$

$$e_1=p_S l_H+p_G l_L+(1-p_S-p_G)l_M$$

$$e_2=p_S l_L+p_G l_M+(1-p_S-p_G)l_H$$

where p_S and p_G are shorthand notations for $u_P(p_S)$ and $u_P(p_G)$, respectively. In the computations sample values of the variables p and l are selected from the curves of Figures 3 and 4. All possible combinations of the selected values form the input set. The min-max rules then determine the respective membership functions of the computed outputs. The expected values E_1 and E_2, in terms of aircraft losses, are shown in Figure 6.

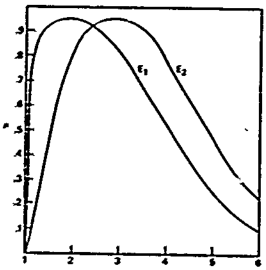


Fig.6.

Similarly, the membership functions for E_1 and E_2 with p_S and p_G modifying inputs s and f are determined from

$$u(e)=\max\{\min[u_P(p_S),u_P(p_G),u_S(s),u_P(f)]\}$$

$$e_1=p_S f+p_G s+(1-p_S-p_G)f$$

$$e_2=p_S s+p_G s+(1-p_S-p_G)f$$

Sample values of p, s and f are taken from Figures 3 and 5 where, again, all possible input combinations are formed. The resultant outputs, in terms of mission success (or failure), are shown in Figure 7.

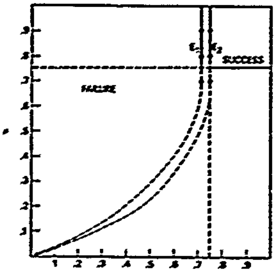


Fig.7.

DISCUSSION

Referring to Figure 6 the highest membership for E_1 and E_2, respectively, shows 2 and 3 aircraft losses. In terms of mission success (Figure 7) the membership function for expected value E_1 is interpreted to mean that about 72% of the target will be destroyed. This may also be interpreted as mission failure. The success suggested by E_2 is only slightly better, i.e., 75%. It appears that the plain route to the target offers the possibility of slightly greater mission success but at a cost of one more aircraft.

Computations involving the probabilities shown in Figure 1 yielded results nearly identical to those computed by the fuzzy set method. However, a sensitivity analysis of the estimated probability inputs of Figure 1 showed a marked change in outputs. Changes in probability estimates as small as ± 0.1 in several branches resulted in outputs showing mission failure where previously mission success was the probable output. The fuzzy set method provided for a kind of sensitivity analysis at the outset by allowing the estimated inputs to take on a range of values. All possible combinations of these values were considered as the input set. Clearly, linguistic quantifiers as prescribed by the fuzzy set method of analysis can play a basic role in dealing with complex problems.

ERROR BOUNDS AND ASYMPTOTIC PERFORMANCE UNDER MISMATCH OF MULTISENSOR DETECTION SYSTEMS

D. Kazakos
Electrical Engineering Department
University of Virginia
Charlottesville, VA 22901

ABSTRACT

We consider a binary detection problem, when the data are obtained from m distant sensors, and modelled as stationary Gaussion processes, with different spectra. We also assume that inaccurate versions of the true statistical models are utilized, and we develop upper bounds to the probability of error, based on the Chernoff bounding approach. Those bounds also converge to the asymptotic probability of error as the number n of data points increases to ∞. Conditions for sustaining, in spite of mismatch, exponential convergence of the error probability to zero with n are determined.

INTRODUCTION

There has been much interest recently in the signal detection problem for data available from multiple sensors [1] - [7]. In this paper we develop bounds to the probability of error for binary detection from multisensor data, when inaccurate versions of the actual statistics are incorporated into the decision rule. We specifically investigate the error probability for detection in Gaussian, stationary processes with inaccurately known spectra.

SUMMARY OF RESULTS

Suppose that m sensors are utilized for deciding between two hypotheses, H_1, H_0. Let $x_k^n = (x_{k1} \cdots x_{kn})$ be the data vector for the kth sensor, distributed according to the probability density function $f_{jk}(x_k^n)$ for j=0,1 under H_0, H_1, correspondingly. Suppose, also, that $g_{jk}(x_k^n)$ is the "inaccurate" version of $f_{jk}(x_k^n)$ that is used in the decision rule. We will assume throughout this paper that the data from different sensors are statistically independent. This assumption will be removed in a subsequent paper. Because of the independence between distinct sensors, the likelihood ratio test is:

$$\text{Decide } H_1 \text{ if } m^{-1} \cdot \sum_{k=1}^{m} q_k(x_k^n) > T \tag{1}$$

where:

$$q_k(x_k^n) = n^{-1} \log \frac{g_{1k}(x_k^n)}{g_{0k}(x_k^n)} \tag{2}$$

is the "mismatched" log likelihood function of the kth sensor.

Let, for j=0,1:

$P_j(f,g,n) =$ Probability of erroneously deciding H_j using $\{g_{jk}\}$ and based on n measurements.

We will utilize the Chernoff bounding technique, and subsequently relate the bound to "large deviation theory." [8], [9], as $n \to \infty$.

The basic Chernoff bound is:

$$P_0(f,g,n) = \Pr[m^{-1} \sum_{k=1}^{m} z_k > T | H_o]$$

$$\leq E_o \exp[t(m^{-1} \sum_{k=1}^{m} z_k - T)] =$$

$$= \prod_{k=1}^{m} E_o \exp[tm^{-1}(z_k - T)] \tag{3}$$

$$P_1(f,g,n) = \Pr[m^{-1} \sum_{k=1}^{m} z_k \leq T | H_1]$$

$$\leq E_1 \exp[-t(m^{-1} \sum_{k=1}^{m} z_k - T)] =$$

$$= \prod_{k=1}^{m} E_1 \exp[-tm^{-1}(z_k - T)] \tag{4}$$

for $t \geq 0$, E_j = expectation under H_j. Note that the bound (3) is less that 1 for some $t \geq 0$, if and only if:

$$m^{-1} \sum_{k=1}^{m} E_o z_k < T \tag{5}$$

Similarly, (4) is less than 1 for some $t \geq 0$, if and only if

$$m^{-1} \sum_{k=1}^{m} E_1 z_k > T \tag{6}$$

Let, now:

$$z_k = n^{-1} \log [g_{1k}(x_k^n)/g_{0k}(x_k^n)] \tag{7}$$

be the mismatched log-likelihood function for the kth sensor, and $s = t\, n^{-1} \geq 0$. Straightforward calculation provides us with the following expressions:

$$E_0 \exp[tm^{-1}(z_k - T)] = \int f_{0k}(x_k^n)[g_{1k}(x_k^n)/g_{0k}(x_k^n)]^{s/m}$$

This research has been supported by a contract from Rome Air Development Center, monitored by Dr Vincent Vannicola, of the Signal Processing Division.

$$dx_k^n \cdot \exp(-snT/m). \quad (8)$$

$$E_1 \exp[-tm^{-1}(z_k-T)] = \int f_{1k}(x_k^n)[g_{1k}(x_k^n)/g_{0k}(x_k^n)]^{-s/m} \cdot$$

$$dx_k^n \cdot \exp(snT/m) \quad (9)$$

$$E_0 z_k = n^{-1} \int f_{0k}(x_k^n) \log [g_{1k}(x_k^n)/g_{0k}(x_k^n)] dx_k^n \quad (10)$$

$$E_1 z_k = n^{-1} \int f_{1k}(x_k^n) \log [g_{1k}(x_k^n)/g_{0k}(x_k^n)] dx_k^n \quad (11)$$

Define the functionals:

$$G_k [f, g_{1k}, g_{0k}, n, s] \stackrel{\Delta}{=}$$

$$n^{-1} \log \int f(x_k^n)[g_{1k}(x_k^n)/g_{0k}(x_k^n)]^{s/m} dx_k^n \quad (12)$$

Taking logarithms of (3), (4) and using (8), (9), (12), we find the bounds:

$$n^{-1} \log P_0(f,g,n) \le \sum_{k=1}^{m} G_k[f_{0k}, g_{1k}, g_{0k}, n, s] - sT. \quad (13)$$

$$n^{-1} \log P_1(f,g,n) \le \sum_{k=1}^{m} G_k[f_{1k}, g_{1k}, g_{0k}, n, -s] + sT. \quad (14)$$

We observe that if G_k^1 denotes the derivative of G_k with respect to s, then, from (10), (12), (7):

$$G_k^1[f_{0k}, g_{1k}, g_{0k}, n, 0] = m^{-1} E_0 z_k. \quad (15)$$

$$G_k^1[f_{1k}, g_{1k}, g_{0k}, n, 0] = m^{-1} E_1 z_k. \quad (16)$$

Hence, if (5) is satisfied, the slope of the upper bound (13) at s=0 is negative, and the bound is zero at s=0. Similarly, if (6) is satisfied, the slope of the upper bound (14) at s=0 is negative, and the bound is zero at s=0. Also, both bounds (13) and (14) are convex with respect to s.

The tightened Chernoff bounds (13), (14) are:

$$n^{-1} \log P_0(f,g,n) \le \inf_s \sum_{k=1}^{m} G_k[f_{0k}, g_{1k}, g_{0k}, n, s] - sT. \quad (17)$$

$$n^{-1} \log P_1(f,g,n) \le \inf_s \sum_{k=1}^{m} G_k[f_{1k}, g_{1k}, g_{0k}, n, -s] + sT \quad (18)$$

Suppose, now, that for the class of statistical models we consider, the limits:

$$\lim_{n\to\infty} G_k[f_{jk}, g_{1k}, g_{0k}, n, s] \stackrel{\Delta}{=} G_k[f_{jk}, g_{1k}, g_{0k}, \infty, s] \quad (19)$$

exist, for k=1,...,m, j=0,1

Suppose, also that:

$$\lim_{n\to\infty} n^{-1} \sum_{k=1}^{m} \int f_{0k}(x_k^n) \log [g_{1k}(x_k^n)/g_{0k}(x_k^n)] dx_k^n < T \quad (20)$$

$$\lim_{n\to\infty} n^{-1} \sum_{k=1}^{m} \int f_{1k}(x_k^n) \log [g_{1k}(x_k^n)/g_{0k}(x_k^n)] dx_k^n > T \quad (21)$$

Then, utilizing the results of Large Deviation Theory, [8], [9], we find:

$$\lim_{n\to\infty} n^{-1} \log P_0(f,g,n) =$$

$$\inf_s \sum_{k=1}^{m} G_k[f_{0k}, g_{1k}, g_{0k}, \infty, s] - sT. \quad (22)$$

$$\lim_{n\to\infty} n^{-1} \log P_1(f,g,n) =$$

$$\inf_s \sum_{k=1}^{m} G_k[f_{1k}, g_{1k}, g_{0k}, \infty, -s] + sT \quad (23)$$

It is interesting to observe that the bounds (17), (18) become the exact asymptotic error probabilities as $n\to\infty$. This is the essence of Large Deviation Theory [8], [9].

We will now concentrate on a major class of statistical models for which the limits (19), (20), (21) exist. This is the class of Gaussian, stationary random processes with zero means and different spectra. Consider three multivariate Gaussian probability density functions $f(x^n)$, $g(x^n)$, $g_0(x^n)$ with zero means and covariance matrices F, C_1, C_0. We calculate the integral of eq (12):

$$G[f, g_1, g_0, n, \theta] = n^{-1} \log \int f(x^n)[g_1(x^n)/g_0(x^n)]^{\theta} dx^n =$$

$$= -\frac{1}{2}\Big[n^{-1} \log |F_1^{-1} + \theta C_1 - \theta C_0^{-1}| - n^{-1} \log |F_1^{-1}| -$$

$$- \theta\, n^{-1} \log |C_1^{-1}| + \theta\, n^{-1} \log |C_0^{-1}|\Big] \quad (24)$$

Taking the limit as $n\to\infty$, and using the results of [10], [11], we find:

$$-2G[f, g_1, g_0, \infty, \theta] = (2\pi)^{-1} \int_{-\pi}^{\pi} \{\log[f^{-1}(\lambda) + \theta c_1^{-1}(\lambda) - \theta c_0^{-1}(\lambda)] -$$

$$- \log f^{-1}(\lambda) - \theta \log c_1^{-1}(\lambda) + \theta \log c_0^{-1}(\lambda)\} d\lambda \quad (25)$$

where, $f(\lambda), c_1(\lambda), c_0(\lambda)$ are the three spectra corresponding to f, g_1, g_0, and it is assumed that they are strictly positive for all $\lambda \in [-\pi, \pi]$. Using (25) for k=1,...,m, and assuming that for the kth sensor the true spectra are $\{f_{0k}(\lambda), f_{1k}(\lambda)\}$ under H_0, H_1, and the assumed spectra are $\{g_{0k}(\lambda), g_{1k}(\lambda)\}$ under H_0, H_1, we can evaluate the rates in (22), (23).

The necessary and sufficient conditions in order for $P_0(f,g,n)$ and $P_1(f,g,n)$ to converge exponentially to zero, are that the derivatives of (22), (23) at s=0 are negative, which are equivalent to (20), (21). The derivative of G with respect to θ at θ=0, is:

$$-2G'[f, g_1, g_0, \infty, \theta=0] = (2\pi)^{-1} \int_{-\pi}^{\pi} \{f(\lambda)[c_1^{-1}(\lambda) - c_0^{-1}(\lambda)] -$$

$$- \log c_0(\lambda) c_1^{-1}(\lambda)\} d\lambda. \quad (26)$$

The condition of negative slope at s=0 of (22) is.

$$(2\pi)^{-1} \int_{-\pi}^{\pi} \sum_{k=1}^{m} \{f_{0k}(\lambda)[g_{1k}^{-1}(\lambda) - g_{0k}^{-1}(\lambda)] -$$

$$\log g_{0k}(\lambda) g_{1k}^{-1}(\lambda)\} d\lambda + 2T > 0 \quad (27)$$

The condition of negative slope at $s=0$ of (23) is:

$$(2\pi)^{-1} \int_{-\pi}^{\pi} \sum_{k=1}^{m} \{f_{1k}(\lambda)[g_{1k}^{-1}(\lambda) - g_{0k}^{-1}(\lambda)] -$$

$$\log g_{0k}(\lambda) g_{1k}^{-1}(\lambda)\} d\lambda - 2T > 0, \tag{28}$$

Conditions (27), (28) guarantee exponential convergence of P_0, P_1 to zero, correspondingly. It can be shown that if (27) is not satisfied, then P_0 converges to 1 as $n \to \infty$. Similarly, if (28) is not satisfied, P_1 converges to 1. The proof of the latter fact is not given here, but it will appear in the expanded version of the paper.

Let us define the spectral distance measure:

$$I(f,g) = (2\pi)^{-1} \int_{-\pi}^{\pi} \{f(\lambda) g^{-1}(\lambda) - 1 -$$

$$\log f(\lambda)\, g^{-1}(\lambda)\} d\lambda \tag{29}$$

Because of the identity $x-1 \geq \log x$, it is seen that $I(f,g) \geq 0$ with equality if and only if $f(\lambda) = g(\lambda)$ for almost all $\lambda \varepsilon [-\pi,\pi]$. After some algebraic manipulations, condition (27) is expressed as:

$$\sum_{k=1}^{m} I(f_{0k}, g_{1k}) - I(f_{0k}, g_{0k}) + 2T > 0 \tag{30}$$

Similarly, condition (28) is expressed as:

$$\sum_{k=1}^{m} I(f_{1k}, g_{1k}) - (f_{1k}, g_{0k}) - 2T > 0 \tag{31}$$

Combining (30) and (31), we see that the necessary and sufficient condition for exponential convergence of the error rate to zero, is the satisfaction of the double inequality:

$$2^{-1} \sum_{k=1}^{m} I(f_{1k}, g_{1k}) - I(f_{1k}, g_{0k}) > T >$$

$$> 2^{-1} \sum_{k=1}^{m} I(f_{0k}, g_{0k}) - I(f_{0k}, g_{1k}). \tag{32}$$

As long as the leftmost side of (32) is larger than the rightmost side, we can always pick a threshold T between those two numbers to achieve asymptotic convergence of the error rates to zero. For the special case of matched statistics, we have $f_{1k} = g_{1k}$, $f_{0k} = g_{0k}$, and the condition (32) becomes:

$$2^{-1} \sum_{k=1}^{m} I(f_{1k}, f_{0k}) > T > -2^{-1} \sum_{k=1}^{m} I(f_{0k}, f_{1k}) \tag{33}$$

We note that (33) can always be satisfied for some T, because the leftmost side is nonnegative, and the rightmost side nonpositive. Thus, in the "matched" case, exponential convergence hinges only upon the choice of T.

To evaluate the actual rates of convergence, we need to use expression (25) for $k=1,\ldots,m$ into (22), (23) and minimize with respect to s.

CONCLUSIONS

We have obtained the rates of convergence of error probabilities in multisensor detection for a binary hypothesis, and we determined the conditions of exponential convergence in the presence of mismatch. The conditions are necessary and sufficient.

REFERENCES

[1] Tenney, R. R. and Sandell, N. R. (1981) "Detection with distributed sensors," IEEE Transactions on AES, July 1981, pp. 501-510.

[2] Teneketzis, D. (1982) "The decentralized quickest detection problem," Proceedings of the 21st IEEE Conf. on Decision and Control.

[3] Srinivasan, R (1986), "Distributed Radar Detection Theory," IEE Proceedings, vol 133, ptF, Feb 1986, pp. 55-60.

[4] Chair, Z and Varshney, P. (1987) "Neymann-Pearson Hypothesis Testing in Distributed Networks," 1987 Conf. on Decision and Control.

[5] Reibman, A. R. and Nolte, L. W. (1987) 'Optimal Detection and Performance of Distributed Sensor Systems," IEEE Trans. on AES, vol 23, no 1, pp. 24-30.

[6] Viswanathan, R. Thomopoulos, S.C A., Tumuluri, R., "Spatial Serial Distributed Decision Fusion," 1987 IEEE Conf. on Decision and Control.

[7] Reibman, A. R and Nolte, L. W., "Design and Performance Comparison of Distributed Detecton Networks," IEEE Trans. on AES, Vol 23, no 6, Nov 1987, pp 789-797.

[8] Chernoff, H. (1952), "A measure of asymptotic efficiency for tests of a hypothesis based on the sum of observations," Ann. of Math. Stat, vol 23, pp. 493-507, 1952.

[9] Ellis, R. S. (1984), "Large Deviations for a General Class of Random Vectors," Annals of Prob., vol 12, pp 1-12, 1984.

[10] Kazakos, D. and Papantoni-Kazakos, P. (1980), "Spectral Distance Measures Between Gaussian Processes," IEEE Trans. on Aut. Control, vol AC-25, no 5, October 1980, pp 950-959.

[11] Kazakos, D. (1982), "Statistical Discrimination Using Inaccurate Models," IEEE Trans. on Infor. Theory, Sept 1982, vol IT-28.

Section III

C^3 Systems and Components

Working Group Chairman:

Prof. John H. Lehoczky
Carnegie Mellon University

SECURE DISTRIBUTED SYSTEMS FOR COMMAND AND CONTROL

Emilie J. Siarkiewicz and Joseph V. Giordano

Rome Air Development Center (COTD)
Griffiss AFB NY 13441-5700

ABSTRACT

Command and Control (C2) systems are likely to consist of hierarchically-organized processing clusters requiring interoperability among heterogeneous components, decentralized real-time processing control, very large database processing, survivability through reconstitution and reconfiguration, and some level of security assurance. To address these concerns, RADC has an on-going research and development program in the area of secure distributed systems with the premise that the functionality and security of the C2 systems can be enhanced through the use of formal specification and verification technology. The program's notable achievements during the last year are presented in a tabloid fashion.

INTRODUCTION

Why do we need secure, or trusted, distributed systems? Command and control systems of the future are likely to consist of hierarchically organized processing clusters requiring interoperability among heterogeneous components, decentralized real-time processing control, very large database processing, survivability through redundancy, reconfiguration, and reconstitution, and some level of security assurance.

With increased hacker and insider threats, running system high is no longer good enough. Furthermore, security engineers have found, through experience, that the higher levels of assurance must be designed in; good security mechanisms cannot simply be added in later phases of system development.

What do we mean when we say a system is multilevel secure (MLS)? A multilevel secure system is one for which assurance of the adequacy of its hardware and software mechanisms, or controls, is sufficient to enforce its policy, that is, the rules governing its use for simultaneously processing multiple classifications and/or categories of information by users with various clearances and needs-to-know.

The set of mechanisms is referred to collectively as the TCB, or trusted computing base. Assurance is a measure of an evaluator's confidence that the security features are correctly implemented and protected from tampering.

Assurance is achieved by means of some combination of hardware protection features, advanced software design and implementation technology, project management and source control procedures, code inspection, penetration testing, and formal verification. Functionality is usually thought to be synonymous with features; but in a secure, or trusted system, assurance is at least as important a part of functionality as features are.

The degree, or level, of assurance is designated by an alphanumeric code as defined in DOD 5200.28-STD, "Trusted Computer System Evaluation Criteria" [1]. Level D denotes minimal or no protection. Levels C1 and C2 denote varying degrees of discretionary (need-to-know) protection assurance. Levels B1, B2, and B3 denote varying degrees of mandatory (classification/category) protection assurance. A1 is currently the highest rating, denoting Verified Design. For each of these evaluation classes, the Criteria specifies a detailed set of assurance requirements, as well as a set of required security features. Thus "B2 assurance" means that which is achieved by following the assurance requirements in the B2 section of the Criteria.

RADC's research and development program in secure distributed systems has three thrusts: distributed operating systems design and prototyping, secure distributed systems specification and verification techniques and methodologies, and secure database management.

The rest of the paper describes some of the work on-going or planned in each of these areas. More detail can be found in the references.

DISTRIBUTED OPERATING SYSTEMS

To address the problem of decentralized processing control and interoperability of heterogeneous components, work is on-going in the area of specification and verification of secure distributed operating systems (SDOSs). In September 1987, the first (Concept Exploration) phase in the development of a secure distributed operating system was completed, in which multilevel security issues, as they relate to distributed operating system design, were investigated [2, 3, 4]. The Cronus DOS, which is described in a companion paper [5], was used as a baseline.

The goals of the project were two-fold. First, there was the desire to understand better the problems associated with developing an SDOS that demonstrates a set of high-level requirements, such as availability and scalability. There is a large set of problems that need to be addressed in this context, many of which are interrelated and cannot be considered in isolation. The intent was to qualitatively evaluate the relative importance and difficulty of addressing each of these problems using four criteria: performance, functionality, feasibility, and usability. Such an analysis was expected to yield insight into the trade-offs faced by the designer of any SDOS.

This project was viewed as a first step in the development of an SDOS that incorporates advanced distributed system technology, including high-level resource management and structured software architectures. Therefore, the second goal of the project was to produce preliminary security documentation in accordance with the A1 (Verified Design) evaluation criteria and to explore possible implementation strategies. Thus the contractor produced a security policy, a formal mathematical model of the security policy, a functional description of the security features and DOS support mechanisms, a detailed top-level specification of the TCB, and portions of the formal top-level specification of the TCB, using the Gypsy specification language.

SDOS POLICY

The purpose of a system's security policy is to define the meaning of security within the system. That is, the policy defines the requirements for the prevention of unauthorized disclosure of sensitive information (i.e., confidentiality), the protection of the accuracy of critical system hardware, software, and data (i.e., integrity), and prevention of denial of service or loss of critical services when required (i.e., assured service). Since every host in a heterogeneous system will have its own definition and implementation of security, a global security policy is necessary as a foundation for allowing different hosts to interact securely. The task of connecting a host into SDOS involves reconciling the local security policy with the global security policy, determining the assuredness of software running on the host, and selecting the appropriate security attributes of the host.

A security policy may be defined at many different levels of abstraction. The more abstract the level, the closer the policy corresponds to one's intuitive understanding of the meaning of security; the less abstract the level, the greater the policy's complexity and the greater the correspondence between it and the secure system's implementation. Since both understandability and detail are important, a policy was produced which has two parts: (1) a high-level, generic part that is independent of any particular system, and (2) a lower-level, system-specific instantiation, applying the generic part to the particular case of SDOS, and including other system-specific requirements.

The generic policy is based on the proposition that message passing is an appropriate model for describing interactions in a secure distributed computer system. The abstract nature of the generic policy is useful for gaining an intuitive understanding of secure information flow. The core of the generic policy is based on the property of restrictiveness. It places a constraint on possible information flow passing through MLS entities in terms of histories of message-passing operations. Entities that satisfy the restrictiveness property separately are assured to satisfy the property as a group of interacting entities. This composability property allows the system policy to be decomposed into the policy on individual components of the system.

The composability property is desirable for command and control systems for several reasons. First of all, as mentioned earlier, as the command and control function becomes both decentralized and integrated, there is a desire to protect the investment in existing software and hardware by networking systems together.

Conversely, it is difficult to verify large, complex systems. Being able to divide the system into smaller components makes the job of specification and verification easier.

Thirdly, a characteristic of distributed systems is the presence of concurrent processes, which leads to a non-deterministic state. Current specification and verification methods depend on being able to treat the system as a state machine. Being able to view the system as a composite of states makes proving the security of the system possible.

Until recently, there have been two basic approaches to stating that a system is secure. The first, referred to as the non-interference property, requires that state changes initiated by a given user not affect the outputs received by another user.

The second approach, referred to as non-deducibility, requires that it be impossible for a user, through observing events visible to him, to deduce anything about the sequence of inputs made by a second user unless the level of the second user is lower than the level of the first.

The non-interference property is only composable for (1) forward branching architectures (that is, a process may not receive input from more than one source and there is no feedback), (2) deterministic systems, (3) uninterruptable systems, and (4) systems with unbounded buffers.

Deducibly-secure systems are composable only if unsolicited write-ups are disallowed. (An unsolicited write-up occurs whenever the system makes a high-level output at a time when there has been no high-level input requesting it.)

The new restrictiveness property [6, 7] overcomes the composability limitations of the non-interference and non-deducibility properties. The restrictiveness property requires that even if one pooled all information based on the behavior of the process at level 1 or below, one's ability to deduce the existence or non-existence of inputs at higher (or incomparable) levels is limited. That is, given a particular set of inputs at higher levels, there is at least one other trace, or history of a process' interaction with its environment, with identical behavior at level 1 or below which masks the existence of the higher-level inputs.

The instantiated security policy rules are divided roughly into three groups: mandatory, discretionary, and configuration. The mandatory policy is a straight-forward application of the generic policy to the entities defined for the SDOS in that it controls the flow of information on the basis of DOD security levels. Thus every subject and object in the system has a label associated with it. The discretionary policy is designed to control the use of the SDOS's abstract operations on the basis of client identities, or a user's need-to-know. The configuration policy defines the system's security "configuration" in terms of a set of "policy parameters" (such as whether the policy is to be tamperproof or experimental and whether auditing is enabled or disabled). The configuration policy also controls the modification of those parameters, both by system users and by changes to the underlying network connectivity. Thus the security policy can be tailored to produce a family of security policies to cover various site requirements, system development phases, or mission phases.

Another aspect of host integration is the implementation of SDOS services necessary to allow the host and the distributed system to be interconnected. These services must be implemented with sufficient assurance to provide multilevel security. This means they must have a B2 assurance rating at a minimum. The assurance of these services determines the assurance of the SDOS. Thus, B3 or A1 are more desirable goals.

In a distributed system, whose components are implemented on many different computers, it is useful to talk about the assurance level of each security feature, or the assurance of the individual components that implement each security feature, rather than the assurance of the system as a whole.

It is a fundamental property of assurance that the assurance of a component, C, can be no greater than the assurance of the components "below" C. Below is loosely defined, such that a component is below C if (1) C depends on it for proper operation (e.g., C calls it), or (2) it is able to access C's databases (e.g., it is in a lower, more privileged ring).

In a traditional, single-host, layered operating system, the relationship between components is clear, straightforward, and obvious. The components in lower layers or rings are below those in higher layers or rings. In a distributed operating system, the relationships between components are much more complex. Different components can be physically located in equally-privileged, mutually-suspicious protection domains (e.g., separate computers). Further, the logical relationship between components can be the reverse of their physical relationship. For example, in a layered communications protocol, a higher layer may do encryption and then pass the encrypted data to a lower layer that need not be trusted since it has no security function. Of course, the lower, insecure layer must not be able to access the higher, secure layer's databases.

Each security feature specified in the Criteria has a minimum required assurance level. For example, it would be pointless and unacceptable to implement mandatory controls in a component having less than B level assurance. However, it is acceptable to implement discretionary controls in a component with C level assurance, and mandatory controls in a separate component having B or A level assurance.

IMPLEMENTATION STRATEGIES

Given the above considerations, an implementation strategy for the SDOS must be chosen that allows each of the required security features to be implemented in a component having the appropriate level of assurance.

In addition to security features and assurance, the time and cost of implementation of the SDOS, preservation of investment in existing application and operating system software, and any specialized requirements of the application to be run on the SDOS must all be considered. Therefore, the concept exploration contractor started with the model of an existing, object-oriented DOS, Cronus, and considered alternative SDOS implementation strategies that could provide the same user functionality as Cronus, in a secure environment.

Cronus runs on heterogeneous hosts, in a layer above existing operating systems such as UNIX (Tm) and VMS. The existing operating system is called the constituent operating system (COS). The hosts communicate over insecure networks such as Ethernet and ARPAnet.

Strategy I consists of adding security features to the existing DOS. This has the obvious disadvantage that, since the underlying operating systems of Cronus have only C2 assurance, the resulting SDOS could have no more than C2 assurance. This is unacceptable because MLS implies at least B2 assurance. This alternative is mentioned only for completeness.

Strategy II consists of porting the DOS to a MLS COS (having assurance of at least B2 and preferably B3 or better), and then adding the appropriate security features. This would involve partitioning the DOS into security-relevant and non-security-relevant parts, and making the former part of the Trusted Computing Base (TCB). This strategy has two variants: Strategy II-A involves writing a MLS COS that provides exactly those features needed to support SDOS. Strategy II-B involves using an existing MLS COS, and building SDOS on top of it.

Strategy II has the advantage of being MLS, but it has the disadvantage of losing heterogeneity and losing investment in existing application software. Strategy II-A has the additional disadvantage of long development time and high cost for the COS, while Strategy II-B avoids these, with zero COS development time, and COS development cost shared by all the customers who purchase rights to the existing MLS COS. In addition, the existing COS would have at least some software development tools that run on it, and possibly also some useful application software. Strategy II-B is clearly preferable on cost grounds; Strategy II-A would only be chosen if no suitable existing MLS COS could be found.

Strategy III involves the use of secure front-ends or access machines to connect existing C2-rated DOS hosts to an insecure network. Mandatory controls would be implemented in the MLS access machines, while discretionary controls would be implemented in the C2 hosts. This strategy has the advantages of preserving heterogeneity and preserving the investments in the existing DOS, the applications that run on it, and all the usable software development tools and application software that runs on the existing COSs. It has the additional advantage of lower SDOS development cost than either alternative in Strategy II. However, it has the disadvantage of not providing for any MLS hosts. This makes some classes of SDOS application systems very awkward, if not impossible, to implement.

Strategy IV is a hybrid of Strategies II and III. It involves use of a MLS COS, on which SDOS software would be built, which could serve either as a secure front-end to connect existing single-level systems, or as an MLS SDOS host, running SDOS application software. This combines the advantages of Strategies II and III: preservation of heterogeneity and investment in existing application software, and provision for MLS SDOS hosts. Implementation cost would be somewhat higher than that of Strategy II alone, but not significantly so, if the hybrid nature of this strategy were in the design from the beginning. This strategy has one additional advantage over either of Strategies II or III: existing, non-distributed applications, that need to be interfaced to a distributed system, but which cannot be changed for reasons such as extreme age, would require not only security in the interface machine, but also some non-security-related processing, to convert old data formats and communication protocols into those used by modern networks and hosts. The hybrid MLS SDOS machine provides both capabilities in one machine.

Strategy IV has two variants, corresponding to II-A and II-B. The hybrid software could be built either on an MLS COS written as part of SDOS development (IV-A), or on an existing MLS COS (IV-B). As with Strategy II-B, IV-B is preferable, provided that a suitable existing MLS COS can be found.

Strategies III and IV share an important advantage: the ability to connect insecure DOS hosts to the SDOS. There will always be some specialized hosts, such as very-high-speed CPUs, or LISP machines, whose use is essential to the success of the application, but which are unlikely to ever have a MLS operating system. The access machine approach allows for connection of such hosts to the SDOS.

Strategy II has the serious disadvantage of loss of heterogeneity. This means that the entire SDOS must be composed of one type of CPU, running one MLS COS. All investment in existing application is lost, unless they can be ported to the new COS. No provision is made for connecting specialized, insecure hosts to the SDOS. (Porting SDOS to several MLS COSs, and possibly writing MLS COSs for several types of CPU, would be prohibitively expensive.)

Strategy IV provides for heterogeneity without incurring the cost of SDOS (and possibly COS) development for several different CPUs.

DEMONSTRATION/VALIDATION

Given the above analysis produced during the Concept Exploration Phase, the next phase of the SDOS development is to take these results and attempt to Demonstrate and Validate them by developing a B3 prototype.

B3 means that the TCB containing the security mechanisms must be isolated from the rest of the operating system and must be small enough to be subjected to analysis and tests. It must enforce mandatory and discretionary access controls and assure object reuse. It must support a trusted communication path between itself and users for use when a positive TCB-to-user connection is required, such as at login. It must record and maintain audit data. A covert channel anaysis must be conducted and a determination made of the maximum bandwidth of each identified channel.

The security policy, formal model, and detailed top-level specification from the first effort will be refined, and an informal proof wil' be produced to show that the implemented TCB corresponds to the DTLS.

The TCB must also provide network security services in the areas of communications integrity, assured service, and compromise protection.

At the same time, the SDOS must provide the distributed operating system functional services described in the previous presentation.

Proposals for this effort are currently (June 1988) being evaluated. We hope to have contract award by the end of September 1988 and the demonstration by January 1991.

SPECIFICATION/VERIFICATION TECHNOLOGY

The second major thrust in our secure distributed systems research and development program is in the area of specification and verification technology. This area includes the development of tools to handle distributed system properties, integration of security engineering into the software development life cycle, and the accreditation of systems.

TOOLS

Current specification and verification methodologies do not adequately handle characteristics inherent in distributed systems such as temporal properties, fault tolerance, and adaptation. A process can fail to make progress because it is continually bypassed, or starved, by the scheduler.

A group of processes can fail to make progress because each is waiting for one or more of the others to take some action and thus they become deadlocked. A process can fail to make progress when it continually chooses to do internal actions and stops attempting to communicate with its environment. This is called divergence.

A system of one or more processes is non-deterministic if the trace, or history, of its outputs cannot be predicted with certainty from the trace of its inputs. This can cause problems in synchronizing processes and with race conditions, where the timing of scheduler requests produces a situation similar to the starvation condition.

In battle management, the timing of events will be a critical factor. It may be desirable to specify that some process will take priority over others, or that some event must occur, or complete, within a certain length of time.

Under one RADC effort, the contractor has been investigating the feasibility of using the state-transition model augmented with temporal logic for specifying the temporal properties requirements of distributed systems [8].

Fault-tolerance is a term readily applied to hardware and is often expressed as mean-time-to-failure, which is greatly influenced by environmental factors. A measure of fault-tolerance which depends only on the system design would be preferred for a secure system. We contend that there are two basic approaches to generating the fault-tolerance specification for a system.

The first subsumes the fault-tolerance properties into other correctness concerns. One then verifies that the system produces correct behavior even in the presence of certain kinds of faults. Thus the specifications for fault-tolerance take whatever form the other correctness requirements take.

The second approach to fault-tolerance specification attempts to isolate the fault-tolerance property from other correctness concerns. A system is fault-tolerant if its behavior in the presence of faults is acceptably equivalent to its behavior without them. A comparison of possible behavior is needed, rather than a focus on the correctness of any one particular behavior. Using this second approach for formal verification will not necessarily require that other correctness properties be additionally verified. This is an important advantage of this approach since the costs involved in producing formally verified software increase with the number of independent properties to be proved, and these costs can be significant.

The contractor working this problem has produced a formal correspondence between fault-tolerance specifications produced under the second approach and specifications for multilevel security that can feed into tools development efforts [9].

The third distributed system characteristic we are investigating is adaptation. Adaptation is desirable to guarantee survivability of mission critical functions. The loss or addition of system modes or processing components will result in system reconfiguration or reconstitution. During battle, timely response may become more important than secrecy. These types of changes are impeded in conventional security policies where any change requires a new specification and reverification.

One study effort that is currently in progress is investigating how such changes in policy are allowable [10]. The goal is to be able to call a system secure for all the different policies under which the system may operate, and if the actual change between policies is made securely.

Existing automated tools for the formal specification and verification of secure systems are designed for single-site systems. As mentioned before, they do not adequately handle characteristics inherent in distributed systems. Nor do they provide features found in most software engineering environments or database design tools.

Based on a study of existing tools and the requirements of distributed systems, we would like to bring together the best of these three disciplines in a single, integrated Secure Distributed System Designer's Workbench [11]. Without such a workbench, verified distributed system design is not possible, or at best, can be done for only relatively simple systems.

Some candidates for specification/-verification tools include TED, a universal Theorem Prover Editor developed by the University of Illinois, the Enhanced Hierarchical Development Methodology being developed by SRI International, and the Gypsy Verification Environment developed at the University of Texas at Austin.

Possible candidates for software engineering tools include SR (Synchronizing Resources), a concurrent programming environment developed at the University of Arizona, and PolyAnna, a high-order specification language based on Anna, a language for annotating Ada programs.

Possible candidates for database design tools include DDEW/DBD, the Database Design and Evaluation Workbench/Database Designer developed at Computer Corporation of America, ATV, the Abstract Database Transaction Programming Language Transaction Verifier developed at the University of Massachusetts at Amherst, and Excelerator developed by Index Technology, Inc.

SOFTWARE DEVELOPMENT

Software is rapidly becoming the larger part of the cost of major command and control systems. As a result, software development standards such as 2167A [12] have been developed to ensure that software meets its requirements. Such standards, however, do not include provisions for providing assurance that the software meets the security requirements as per the TCSEC. Nor is there any guidance as to how the two standards are to be used in concert.

One RADC contract, which is investigating software life cycle issues with respect to security requirements, has identified eight leverage points in the software development process that appear to have critical impact on the security of large command and control systems [13]. These are visibility, protocols, downloading, configuration control, monitoring and measurement, environment, communication, and verification.

Visibility is especially critical during the specification and design phases. Security requirements are usually in competition with other requirements such as functionality and performance. Security requirements are often not visible at critical decision times, since there is no established tracking process during the software development process. Visibility is further complicated by the fact that there may be different constituencies for different security requirements. That is, one agency may be primarily interested in assuring confidentiality with little concern about integrity or assured service, while another agency may be more interested in assured service. One way to ensure that security requirements have appropriate visibility is to establish a security review board early in the process, which is composed of representatives from all the user agencies. It is also important to make an early determination of whether the security requirements are within the state-of-the-art or not, which could have unexpected impact on cost, schedule, or performance.

Because C2 systems of the future will be highly distributed, communication will be at the heart of the system. Consequently, protocols will be pervasive. Some standards already exist, such as TCP/IP and ISO/OSI, but it is likely that new protocols will need to be developed, which is a difficult process. Protocols provide bridges across system and organizational boundaries, which makes them vulnerable to security attacks. To facilitate security analysis at the protocol level, it is important to identify protocols early in the design, put protocol development under configuration control, subject protocols to security design review, and subject protocols to security analysis, testing, and verification.

One of the most vulnerable times in the system development process is at downloading, i.e., changing over to a new version of the system. The new version may be faulty resulting in loss of communication or frequent restarting. Since it is virtually impossible to distribute new versions of the system to all nodes at exactly the same time, different nodes will be running different versions. If those versions are incompatible, the result will be a partition of the system. The downloading channel itself is subject to penetration. Thus, special attention must be given to ensuring that only authorized sources are permitted to download software. Conversely, overly suspicious nodes will refuse to be controlled, and thus downloading cannot be accomplished. Consequently, special attention needs to be given to the development of downloading mechanism and proceures to ensure that they do not contain security flaws and to ensure that certification and accreditation are maintained.

Two problems exist with respect to configuration management which impact the specification, implementation, certification/accreditation, and maintenance phases of the system development. One has to do with the system complexity and the other with its sensitivity. A C2 system is a collection of subsystems. As such, no configuration control system currently exists to handle compound systems. Once software has been entered into the baseline configuration, it is not easily removed. This is true for tools, such as compilers, as well as pieces of code. Few verified tools exist. Consequently, it is practically impossible to guarantee the correctness of all tools and components used in the software development process. The result is a false sense of integrity from careful control of what may be an initially flawed module. To minimize these vulnerabilities, it is important to define an overall configuration control policy, review the procurement of tools for vulnerabilities, and consider using improved versions of tools as they become available during the software development process.

An important feature of a C2 system to ensure health of the system is the monitoring and measurement functions, such as message acknowledgements. These feedback channels, however, may violate the security policy. To avoid this conflict of interests, it is vital that the monitoring and measurement functions be designed into the system from the beginning and be subjected to the same kind of security analysis as the other communications paths.

From a software security point of view, it is desireable to use the most qualified people and the strongest set of tools. From a personnel security point of view, it may be necessary to do software development in a closed environment with a limited set of (cleared) people. Consequently, there is a delicate balance between security derived from control over personnel and the quality of the environment in which the software development takes place. There is a similar tension between the mandated use of Ada (tm) and the lack of Ada support tools. To maintain a proper balance, it is essential that the selection of computing equipment and the associated software development environment be reviewed prior to inclusion in the software development plan and the software test plan, and any limitation adequately justified with respect to the integrity and confidentiality of the code.

Development of large, complex C2 systems will necessarily be distributed across distance and organizations. Shared access to documents permits better version control and peer review. Communication between developers, there, is crucial to avoid divergence, but is often inhibited. Formal and informal reviews, workshops, and networking can facilitate this communication. The networking can take many forms including electronic mail and teleconferencing via asynchronous forums and synchronous "telemeetings".

Security anaylsis of large software modules is too tedious to do manually. Consequently, verification of the complete correctness of a C2 system is well beyond the state-of-the-art. However, automated formal verification is potentially within the state-of-the-art for certain aspects of the code, such as information flow properties and starvation states. These are important to ensure that there are no unintended signalling channels and that the system will not cease to function for lack of included in the software development plan for critical sections, or functions, and for selected properties, such as fault-tolerance.

ACCREDITATION

Systems that process classified information must go through a security accreditation process to determine whether sufficient procedures and mechanisms are in place to protect the information. The accreditation decision is made by a Designated Approving Authority (DAA) who is, or acts on behalf of, the operational entity responsible for the system.

The DAA is most often the "owner" of the system, including the data it processes, and must possess the ability to control the resources necessary to maintain the validity of the accreditation. The DAA begins with inputs from the system certifiers, who have evaluated the system against some standard criteria such as DOD 5200.28-STD, "DOD Trusted Computer System Evaluation Criteria", and gives it a rating. The certifying agency is usually NSA.

The DAA must then compare and balance the mission functional requirements against the security assurance provided by the system. Are the assurances necessary? Are they adequate? Do they impede other mission critical functions? How much risk is the DAA willing to take to ensure other functional capabilities? In the end, it is a management decision weighing acceptable risk and acceptable functionality.

The complexity of a large, complex command and control system and the amount of interaction with systems outside of its operational control make it difficult to identify a single overall DAA. The accreditation of subsystems and components within a large command and control system itself may involve multiple accreditors, representing the different organizations involved in specific subsystems. The determination of relationships among the various accrediting authorities and the subsystem DAAs is vital to the success of the accreditation program and must be resolved early in the system development process.

The fact that most command and control systems are developed in an evolutionary, incremental fashion further complicates the accreditation process. Current procedures dictate that every change requires recertification and, thus, reaccreditation.

Accreditation is also complicated by the fact that the resources may be distributed among fixed ground, mobile, and space-based facilities. In particular, some form of remote upgrade and maintenance is needed for space-based and other unmanned components. Using an electronic remote upgrade/maintenance mechanism creates both great technical and managerial concerns with respect to accreditation. Management of changes authorized for human-accessible systems is complicated but can usually be handled procedurally. Management of changes accomplished via an electronic mechanism becomes quite complex. The key to the problem is knowing that the change made to the system was indeed the authorized change. It cannot be tracked procedurally.

Another accreditation issue is that of adaptive policies, which were discussed earlier and present accreditation problems similar to evolutionary development and distribution of resources.

Achievements to date on the accreditation study effort includ identification of issues [14] and the development of a framework for developing an accreditation plan based on an abstract system architecture and an application-specific interpretation of DOD 5200.28-STD [15].

DATABASES

Database systems are an integral component of current and future C2 information and decision support systems. It is predicted that today's military commander or decision-maker will rely upon data and information stored in electronic databases for battle management functions. In the truest sense of the word, database technology has advanced to the point where it is being infused into operationally-oriented system environments. As more and more data is stored in databases the problem of protecting data and information arises. The recent West German hacker scandal is proof that our nation's computer and database systems are inadequately protected against hostile and malicious attacks.

Within the Air Force, as in the Department of Defense as a whole, the number of computerized databases in use is increasing [16]. A number of missions which use database management systems (DBMS) technology are involved with processing at least some classified data. The natural mixture of data within these systems is such that the various data may be classified at different security levels.

The problem of handling multilevel data within the DBMS environment is compounded by the fact that typical commercial-off-the-shelf (COTS) DBMS's are not built with adequate controls to enforce DOD security doctrine. The current method by which most organizations control access to information is known as "system-high" processing mode. In this mode, all personnel who have access to any database containing some restricted data must be cleared to see all of the information in the system. It is a well-known fact that "system-high" is both vulnerable and expensive.

In response to the aforementioned problems, the Air Force Study Board established a Summer Study on Multilevel Data Management Security. The study was conducted 12-30 July 1982 at Woods Hole, Massachusetts. Group 1 identified three architectures with the potential to meet security and performance requirements for the near-term. The different architectures, Integrity-Lock, Kernelized, and Distributed, employ the technologies of cryptographic sealing, secure operating systems, and back-end DBMS's, respectively. Group 2 of the Summer Study dealt with the problem of constructing a secure military message system. Group 3 of the Summer Study had the charter to investigate long-term solutions to the DBMS security problem. Group 3 recommendations included developing a security model whereby data is protected at the view level. The group investigated potential strategies for dealing with inference and aggregation and also began studying the utility of knowledge engineering techniques to assist in maintaining a secure DBMS processing environment.

Since 1974, the Rome Air Development Center (RADC) has been very active in sponsoring research programs aimed at designing and prototyping a family of trusted DBMS's. The research program sponsored by RADC can be logically partitioned into four major areas of emphasis: architectures/mechanisms, and tools, automated/expert database design tools, and data integrity.

ARCHITECTURES/MECHANISMS

Early RADC work in the architectures area include the Hinke-Schaefer model [17], the System Development Corporation "Secure DBMS Techniques" effort [18], and the Integrity-Lock prototype. The Hinke-Schaefer work was a landmark study in which a secure relational DBMS was designed to operate on the secure MULTICS operating system kernel. DBMS relations were classified at the column (attribute) level with each attribute decomposed into MULTICS protection segments.

Concurrent with the initiation of the 1982 Air Force Summer Study, RADC signed a three-year contract with System Development Corporation to develop a secure DBMS design methodology. The effort, entitled "Secure DBMS Techniques", culminated in a manual, top-down design methodology providing a system designer with a tool for making high-level architectural choices independent of any security policy or COTS DBMS product. The final report of the effort identifies eight generic secure DBMS architectures and eight key decision attributes with their associated metrics. The decision attributes are: ability to counter security vulnerabilites, relative performance, simplicity of trusted software, simplicity of untrusted software, minimizing number of machines/disks per security level, security policy support, ability to support user authentication within the DBMS, and minimization of R&D. The overall design methodology allows a designer to specify key decision and design criteria with the output being a family of candidate architectures which meet certain critical design criteria.

The Integrity-Lock prototype effort [19] was a 20-month contract with the MITRE Corporation. In the Integrity-Lock concept, a database entity, either a relation, tuple, attribute, or field, is tagged with an unforgeable, encrypted checksum or signature as the entity is input into the database. When a database entity is retrieved, it is taken from the disk and passed to a trusted computational component which recalculates the entity's checksum to ensure that the data has not been changed or tampered with. MITRE developed a proof-of-concept prototype by retrofiting the Integrity-Lock mechanism onto a COTS DBMS. Based upon the initial proof-of-concept, the Integrity-Lock is being implemented at an operational Air Force site.

An outgrowth of the early RADC-sponsored research in architectures as well as the positive results of the 1982 Air Force Summer Study led to the research in four other architectural approaches: secure distributed data views (SeaView), LOCK data views, secure entity-relationship DBMS, and secure back-end DBMS. In August 1985, RADC awarded a contract, entitled "Secure Distributed Data Views" (SeaView), to SRI International and their subcontractor, Gemini Computers, to design a Class A1 secure DBMS where data is accessed through user views. Thus far in the effort the contractors have produced a security policy, formal model, and formal top level specification (FTLS) for an A1 secure relational DBMS [20, 21, 22]. Work is progressing on the development of an implementation specification and proof-of-concept demonstration.

The SeaView architecture is basically a two-level architecture. At the top-level is the extended trusted computer base (TCB) which is responsible for enforcing discretionary access control and consistency policies for multilevel relations. The underlying layer of the SeaView architecture is the general-purpose TCB which is responsible for enforcing access control policies on storage objects. Simplistically, the abstraction of multilevel user views and database relations is decomposed into single-level storage objects and stored under the auspices and protection of the tamperproof, general purpose TCB. The demonstration system plans to use Oracle as the database engine and GEMSOS as the underlying TCB.

SeaView's major accomplishments include: (1) a security policy and policy interpretation that define what security means for a database system, (2) a multilevel relational data model that can accommodate element-level granularity of security labels, (3) a formal model that defines security properties that define a secure state, and transition properties that further restric allowable transitions, (4) initial definition of multilevel SQL database language, (5) rules for determining the appropriate label for derived data, (6) rules for labeling incoming data, and (7) the formulation of the concept of polyinstantiation which refers to the simultaneous existence of multiple data objects with the same name, where the multiple instantiations are distinguished by access class, thereby preventing low-level users from inferring the existence of high-level objects.

In December 1985, RADC awarded a contract, entitled "LOCK Data Views", to Honeywell to design a Class A1 secure DBMS for the LOgical Coprocessing Kernel (LOCK). The LOCK provides a platform upon which the secure DBMS will be built. Simply, the LOCK can be described as a general purpose A1 architecture which provides for security enforcement in a hardware kernel. Since LOCK is providing the foundation for the secure DBMS application, it is believed that the amount of verified DBMS code can be kept to a minmum.

The secure DBMS architecure for the LOCK is structured such that all access to the database is through user views. The heart of the LOCK Data Views architecure is the pipeline. The three system pipelines are the query/response, data input/update, and database definition/metadata pipelines. These pipelines combine with discretionary and mandatory access control mechanisms to ensure the trustworthiness of database processing. The LOCK Data Views effort has produced a security policy and implementation specification with work continuing on the formal model and FTLS.

The secure entity-relationship (E-R) data model effort completed in May 1988. The contractors, AOG Systems Corporation and Gemini Computers, developed a security policy, formal model, descriptive top level specification, and functional description for a Class B3 secure DBMS [23]. The core of this effort was the E-R data model. This model is a semantic data model and describes the real world in terms of entities, attributes, and relationships.

An entity is a data structure which describes some thing, concept, or phenomenon which can be named. An attribute or property is a characteristic of an entity. For example, the entity EMPLOYEE could have the following properties: ID#, NAME, SALARY, and RANK. Finally, a relationship is a data structure which describes an association between entities. For example, a simple E-R database could have a relationship WORKS-ON which associates employees and the projects they work on.

The effort culminated with a demonstration of a simple E-R DBMS on top of the GEMSOS TCB. The demonstration portrayed the functionality of the data model and how a user could add, delete, and modify various E-R data structures.

Unisys Corporation is under contract with RADC to develop a security policy, formal model, DTLS, and implementation specification for a Class B3 secure DBMS architecture. This effort is based on the multiple back-end DBMS approach. In this approach, a multilevel database is physically segregated on different back-end database machines. The backbone of this architecture is a secure local area network with all access to the database passing through users workstations interfaced to the network.

For example, a user at a workstation submits a multilevel query or request to the workstation. The workstation passes the query to a trusted front-end computer, such as a Gemini processor, which then examines the query and decomposes it into its single-level constituent parts. The singe-level subqueries are then passed to the appropriate back-end DBMS's where partial results are computed. Partial results are brought back into the front-end trusted processor where final query processing and packaging occurs.

This effort is based upon COTS components and it is believed that this architecture represents a feasible, viable approach [24, 25]. A major issue with this approach includes how to handle the interaction of DBMS, network, and processor TCB's. It is also important to examine how data replication issues impact multilevel query processing and update capabilities.

EMERGING TECHNOLOGIES

In addition to the architectural research programs descibed above, two emerging technologies come to mind which have direct applicability to the military database environment. Each area warrants consideration in the context of multilevel security.

First, is the rapidly advancing technology of object-oriented databases. Object-oriented databases are touted as being capable of meeting the military's future database processing needs, especially in terms of handling multimedia data types such as text, images, time, knowledge, and conventional data.

The second area of importance is that of knowledge-base management systems (KBMS). Numerous research has been conducted by DOD in the area of KBMS's and some systems, such as the Navy's Force Requirements Expert System, are beginning to see operational use. If KBMS technology is to be infused into DOD operational environments it is important that security issues be examined and that security be designed into systems from the beginning.

AUDITING

In the area of auditing for secure DBMS processing environments, very little work has been done to date. Auditing is important in terms of providing the capability to conduct independent review and examination of system records and activities [26]. Auditing is a true challenge in the database environment with its numerous transactions and overall dynamism. For a database system to be considered secure it must contain an audit capability that not only captures all security-relevant events and transactions but also provides a security officer with the facilities for reducing the otherwise voluminous audit trail into a coherent and understandable system history.

RADC believes that the time is ripe for developing prototype audit systems. Technology advances in the areas of workstations, graphics, windows, and expert systems provide the technology baseline for designing and evaluating audit software. An important issue in the development of a reliable and efficient audit system is how to build a generic audit shell with extensible features to capture application-specific audit requirements.

DATABASE DESIGN

Another area of extreme importance is that of providing Computer-Aided Software Engineering (CASE) support to the design of multilevel databases [27]. The exercise of designing databases is a very complex process. A database designer typically performs a requirements analysis, describes transactions and functional dependencies, and finally develops conceptual and logical schemas. The total process is extremely labor intensive and error-prone. When one factors in the implications of multilevel security into large-scale military databases it is easy to see that the potential exists for making mistakes in the database design which could lead to aggravated inference and aggregation problems.

Over the past few years, RADC has sponsored research in the area of automated database design tools, specifically the Database Design and Evaluation Workbench (DDEW) developed by Computer Corporation of America. Recently RADC has been involved with efforts to examine the use of expert systems and knowledge-engineering techniques to analyze database design structures for inference and aggregation problems [28].

The goal of the RADC-sponsored research is to design a set of tools which enable a database designer to logically design a multilevel database and based upon both the security requirements of the environment and the semantics of usage to analyze the database for potential security vulnerabilities.

DATA INTEGRITY

The final area of research is that of data integrity. Data integrity means many things to many people. In a C3I environment, data integrity is critical since polluted or maliciously modified data can have catastrophic effects on the outcome of a mission. As part of RADC's research program in the architectures and mechanisms area, data integrity issues have been initially investigated. However, more research and experimentation is required to better understand how data integrity fits into the secure DBMS environment.

RADC is currently examining, in-house, how to test various data integrity mechanisms, including polyinstantiation, referential, entity, semantic, and Biba integrity. The results of RADC's internal investigation will be factored into upcoming architecture prototype contracts. In terms of data integrity concerns, the concept of polyinstantiation stands out.

As stated earlier, polyinstantiation is the simultaneous existence of multiple objects with the same name at different access classes. The key and security label of the tuple actually constitute the multilevel key. It can be seen that the concept of polyinstantiation, if implemented in a secure relation DBMS, can result in a number of similar tuples visible to high-level database users.

This situation can result in a very confusing view of the database for a high-level user. Therefore, user interfaces and multilevel query language tools must be developed to allow a high-level user to browse a set of polyinstantiated tuples and retrieve the correct result. It is intended that all of the integrity mechanisms developed by RADC architecture contractors will be extensively tested and evaluated before being included into operational secure DBMS environments.

CONCLUSION

Command and control systems can be considered the nerve system of our nation's armed forces. C2 systems are necessary in all environmental settings, namely peacetime, crisis, anti-terrorism, and wartime modes of operation. As computer, communication, and surveillance technology continue to improve, more data than ever is being collected and stored in electronic information systems. With an increase in our ability to automate military decision-support systems comes an increase in the vulnerability of the critical data stored and processed within that system. The recent West German hacker incident is visible proof that our nation's military computer systems are vulnerable to hostile attack. Also, as evidenced by the Walker spy scandal and other similar cases, the insider threat is real.

The realization that a threat does indeed exist has led RADC to define a comprehensive security R&D program in the technology area of distributed systems. This program deals with distributed operating systems, trusted system engineering, and database management. Since the information stored in automated information systems is used to make critical national security decisions, it is important that the data be available, uncorrupted, uncompromised, and correct.

The noted military strategist Carl von Clausewitz developed, in the 1800's, the concept of "friction" in war. In his classic book, On War [29], Clausewitz states, "Everything is very simple in war, but the simplest thing is difficult. These difficulties accumulate and produce a friction of which no one can form a correct idea who has not seen war." Friction can be viewed as the uncertainty of war based upon unforeseen occurrences, situations, or phenomena. One of the major reasons for automating our C2 systems is to help alleviate this friction.

If our nation's C2 systems are to meet the goal of providing correct and reliable information to decision-makers and thus lessen the effects of friction, it can be seen that the characteristics of confidentiality, integrity, and availability become just as important as other system architecture concepts and drivers. In conclusion, with accurate and uncompromised information in-hand, our nations key strategic and tactical decision-makers have an important asset by which to lessen the effects of friction on the complex process of military planning and decision making.

REFERENCES

[1] Dept of Defense, Trusted Computer System Evaluation Criteria, DoD 5200.28-STD, Dec 85.

[2] Casey, T. A., S. T. Vinter, D. G. Weber, R. Varadarajan, and D. Rosenthal, "A Secure Distributed Operating System," Proc 1988 IEEE Symposium on Security and Privacy, Oakland CA, Apr 88.

[3] Vinter, Stephen T., "Extended Discretionary Access Controls," Proc 1988 IEEE Symposium on Security and Privacy, Oakland CA, Apr 88.

[4] Vinter, S. T., et al, The Secure Distributed Operating System Design Project, RADC-TR-88-xxx, in preparation.

[5] Lawrence, Thomas F. and Richard A. Metzger, "Cronus - A Distributed Operating System (DOS) for Future Command and Control Systems," Proc 1988 JDL Command and Control Research Symposium, Monterey CA, Jun 88.

[6] McCullough, Daryl, "Non-Interference and the Composability of Security Properties," Proc 1988 IEEE Symposium on Security and Privacy, Oakland CA, Apr 88.

[7] Weber, D. G. and Bob Lubarsky, "The SDOS Project - Verifying Hook-Up Security," Proc 3rd Aerospace Computer Security Conference, Orlando FL, Dec 87.

[8] Odyssey Research Associates, Inc. and RCA/Aerospace and Defense, Temporal Properties of

Distributed Systems, Interim Report Contract F30602-86-C-0263, Nov 87.

[9] Weber, D. G., Specifications for Fault-Tolerance, ORA TR 19-3 (Interim Report Contract F30602-86-C-0263), Jan 88.

[10] Schneider, Edward A. and Leena-Maija Reissell, Adaptive Policies, Interim Report Contract F30602-86-C-0263, in preparation.

[11] Halpern, J. Daniel and Sam Owre, "Specification and Verification Tools for Secure Distributed Systems," Proc 10th National Computer Security Conference, Baltimore MD, Sep 87.

[12] Dept of Defense, Defense System Software Development, DOD-STD-2167A, 29 Feb 88.

[13] Crocker, Stephen D. and Emilie J. Siarkiewicz, Software Methodology for Development of a Trusted BMS: Identification of Critical Problems, Unisys TM-8361/003/00 (Interim Report Contract F30602-86-C-0048), Apr 88.

[14] Unisys Corp and Trusted Information Systems, Inc., SDI Battle Management System Accreditation Issues, Unisys TM-8361/001/00 (Interim Report Contract F30602-86-C-0048), Jun 87.

[15] Unisys Corp and Trusted Information Systems, Inc., SDI BMS Security Accreditation Phase I Report, Unisys TM-8361/001/00 (Interim Report Contract F30602-86-C-0048), Jun 88.

[16] Air Force Studies Board, Multilevel Data Management Security, National Research Council, National Academy Press, Washington DC, 1983.

[17] Hinke, Thomas H. and Marvin Schaefer, Secure Data Management Systems, RADC-TR-75-266, Nov 75.

[18] Hinke, Thomas H., Design Considerations for Secure Database Management Systems, RADC-TR-86-75, Aug 86.

[19] Duffy, Kevin and Richard Graubert, Integrity Lock Design Documents, RADC-TR-86-108, Aug 86.

[20] Denning, D. E., T. F. Lunt, R. R. Schell, et al, Security Policy and Interpretation for a Class A1 Multilevel Secure Relational Database System, SRI, Interim Report Contract F30602-85-C-0243, Nov 86.

[21] Denning, D. E., et al, Secure Distributed Dataviews, The SeaView Formal Security Policy Model, SRI, Interim Report Contract F30602-85-C-0243, Jul 87.

[22] Lunt, T. F. and R. Alan Whitehurst, The SeaView Formal Top Level Specifications, SRI, Interim Report Contract F30602-85-C-0243, Feb 88.

[23] Lefkovits, Henry C., George Gajnak, Roger Schell, William Shockley, et al, Multilevel Secure Entity Relationship DBMS, RADC-TR-88-xxx, in preparation.

[24] Swanson, John E., Secure Distributed Database Management System - State-of-the-Art Survey, Unisys Corp, TM-WD(L)-8905/017/00, Nov 87.

[25] Westby-Gibson, Doug T. and James Gray, Secure Distributed Database Management System - Component Trade-off Analysis, Unisys Corp, TM-WD(L)-8905/018/00, Apr 88.

[26] National Computer Security Center, A Guide to Understanding Audit in Trusted Systems, NCSC-TG-001, Version 1, Jul 87.

[27] Conversation with Ms Hillary Hosmer (MITRE Corp) and Capt Mike Merriman (Electronic Systems Division), 13 May 88.

[28] Morgenstern, Matthew, "Controlling Logical Inference in Multilevel Database Systems", Proc 1988 IEEE Symposium on Security and Privacy, Oakland CA, Apr 88.

[29] Clausewitz, Carl von, On War, Ed. and trans. Michael Howard and Peter Paret, Princeton University Press, 1976.

THE AT&T BT-100 PROCESSOR: A SYSTOLIC PROCESSING ENSEMBLE

Richard R. Shively and Allen L. Gorin

AT&T Bell Laboratories

ABSTRACT

The ASPEN project is a DARPA-sponsored multiprocessor architecture investigation, initially focused at evaluating the effectiveness of the binary-tree form of multiprocessor topology at a broad class of structural pattern recognition applications.

Several example applications have been demonstrated on the initial processor designed and built as part of this project including: a) several real-time, speech recognition applications (with continuous speech as input), b) K-means clustering, and c) several kernels of linear algebra as well as linear programming. Other applications studied and simulated on the architecture include: a) bipartite graph match problems, which occur in sensor fusion and target tracking, and b) synthetic-aperture-radar processing.

Physical realization of the architecture has resulted in the highest processor density reported in the literature, viz: 2 gigaflops/cubic foot. By application of newer technology, including wafer-scale packaging, a 40 gigaflops/cubic foot capability is planned. This figure of merit is clearly significant for embedded Command-and-Control processors.

INTRODUCTION

Many Command and Control problems require the availability of very fast computing technology, and often the circumstances mandate tight restrictions on power, volume, and weight. Avionics and space-based applications are perhaps the most stressing in physical constraints, but these parameters are important in any deployed mobile environment. Indeed, if the premium paid to achieve compactness is not significant, even applications in a benign environment clearly benefit from an easily portable, compact processor implementation. The processor technology developed as part of the DARPA-sponsored ASPEN project provides an approach to meeting these objectives while providing a very powerful approach to implementing many of the kernels of processing required for CC decision aids.

The kernels of processing of many application areas requiring fast embedded processors are amenable to solution by highly parallel processor architectures. The key to understanding a processor architecture is understanding the class of applications at which it is targeted. A brief list of application drivers for the ASPEN project is worthwhile:

a. Linear programming is the core of many allocation problems, and Gaussian elimination is normally the approach for solving the set of simultaneous linear equations embedded in this process. A technique for achieving an N-fold acceleration of this process with N processors is outlined in the Applications section.

b. Several distinct Command and Control applications can be abstracted as bipartite graph match problems. Finding the optimal mapping of the elements of one set (or graph) onto points in another set according to some cost criteria is the objective. This is

* This work was supported in part by DARPA under contract N00039-86-C0173.

exemplified by relating two or more target files derived from two different surveillance sensors i.e.sensor fusion, or may involve relating objects detected by the same sensor in successive looks. (Objects detected in successive frames of an IR detector, or in successive stages of a mobile tactical passive sonar survey are examples of the latter.) For these and many other examples, a highly parallel processor structure that can support an associative memory mode of operation is the key to relating large sets of data. This match process may be implemented by distributing the elements of one set among several distinct parallel processors, and then broadcasting elements of the second set to all processors to enable performing simultaneous match operations, including detection of a "no match" circumstance. Unlike the classical associative memory in which some form of exact match is sought, any relationship definable by a stored program can be used.

c. In what may be viewed as a dual to broadcasting data to an ensemble of processors for fast parallel matches outlined above, the fast determination of maximum, minimum, or more generally the "k best" among a data set is a key kernel in many CC processing steps. Assuming the elements of the data set are distributed among the parallel processors, the max/min search can be achieved in logarithmic time with a binary tree topology, and the search for successively ranked elements can proceed with a new result each instruction cycle if proper hardware and software capabilities are included. Rank order of threat evaluation in fast-evolving anti-aircraft engagements, and determination of the k best template matches in support of pursuing multiple hypotheses in structural pattern recognition are two representative examples of how this operation plays an important role.

d. Clustering is an essential step in many applications where a large amount of noisy data items must be related. EW and large-area passive sonar surveillance are two examples of where the number of signal detections will in general be much greater than the number of platforms generating the signals. Therefore, linking items in the data set (signal detections) is essential to understanding the information, both for reasons of grouping signals from non-threat platforms and deriving further classification information as a result of knowing the combination of signals from a given platform. In its simplest form, clustering is the match process described earlier. More generally however, clustering involves starting with data items represented by a scatter-plot in an N-dimensional space, and algorithmically determining the most probable number of entities (platforms in the examples cited) that the measurements represent. "K-means" clustering refers to assuming successive values of K as the number of entities represented, and finding the K centroids that yield minimum total variance for each value of K.

Signal Understanding refers to a broad class of problems in the automatic pattern recognition and interpretation of sensor data that encompasses many of these processing kernels, and was therefore chosen as an area in which to implement demonstrations to investigate the effectiveness of a highly parallel processing capability. As described in [1], the pattern recognition process involves measurement of a set of relevant features from the signal in question (speech, sonar, imagery, or radar), and then a comparison of the feature set with a library of stored reference templates also expressed in the same features. The unknown signal is then classified as that reference to which it best matches. This is illustrated in Fig. 1. The comparison process may in general be hierarchical as a pattern may be expressed as a composite of many primitive elements, and may involve pursuit of multiple hypotheses as a multi-stage, sequential decision process keyed by contextual information is implemented.

Speech recognition is a specific instantiation of many of these concepts. For example, in speech recognition one typically measures features that characterize the short-term acoustic spectra of the spoken words [2]; these features are then compared to a reference library of statistical models that enable matching to occur despite significant distortion in the rate of speech (specifically Hidden Markov models), and classified as that reference that has maximum likelihood.

In *syntax-directed level building*, the word models are augmented by syntax rules, such that pattern classifications are not performed in isolation but rather guided by the K-best hypotheses regarding what is said before and after the word in question together with constraints imposed by syntax rules regarding what sequences are allowable. This speech recognition demonstration, implemented on the ASPEN system to run in real time, was chosen to illustrate the concepts of contextual pattern recognition and an approach to efficient implementation of this on a parallel processor structure as outlined above.

ARCHITECTURE OVERVIEW

The AT&T BT-100 processor is a network of processing elements (PE) interconnected in a binary tree. Each PE comprises a programmable digital signal processor, memory, and a communications processor, as shown in Figure 2. Each board comprises a 7 PE subtree plus an expansion PE. The expansion scheme, as described by Leiserson in [3] and illustrated in Figure 3, allows one to build larger trees from identical submodules. The interconnection scheme is *recursive* since interconnecting two modules of the type shown in Fig. 3 yields a resultant module that again comprises a subtree and one expansion PE with a total of four backplane buses, just as the initial modules had. Thus, a tree-machine has attractive scalability characteristics. In particular, the PEs remain the same as the tree-size increases, as do the board-level modules, and larger configurations can be realized in the field by simple insertion of modules and moving backplane jumper blocks. Furthermore, the interconnection overhead is small with respect to both hardware and software, leading to compact, low-cost machines.

An important choice in the BT-100 development was implementing the PE with a digital signal processor (DSP), which is a microprocessor that is particularly effective at fast floating-point arithmetic. DSP chips are often perceived as serving a more specialized functional market, but when capable of both integer and floating-point arithmetic and accompanied by a full higher-order-language compiler/optimizer, they perform well in a variety of applications not restricted to signal processing. The AT&T DSP32 [4] which is being used as the PE for the BT-100 provides 32-bit floating-point arithmetic, 8 or 12.5 megaflops in currently available devices, and all of the software support items listed above. No support devices (co-processors, I/O chips, etc.) other than memory are required for the DSP32 operation. The DSP32 has been benchmarked on the Whetstone benchmark at 6.5 megawhetstones, using the C compiler/optimizer described in [5].

SOFTWARE ARCHITECTURE

The binary tree hardware architecture is scalable in two specific senses. First, the PEs do not change as the size of the tree increases, remaining 3-port devices communicating with only a parent and two children. Second, the 4-port Leiserson module is also unchanged as the tree size increases, regardless of the number of PEs in the module. Moreover, the BT-100 software architecture leads to an operating system and application programs that are also scalable, in the sense that the code is independent of the size of the tree.

There have been many proposed software paradigms for tree machines, the most general and flexible of which program each PE individually in a MIMD mode and achieve interprocessor communications via tree neighbor protocols. One of the major results of our work has been the demonstration that a restricted programming paradigm, described below, suffices for many applications of interest. This result is important since it provides a scalable software environment, in which processing and global communications are orchestrated via function calls from a serial host program. For a large class of applications, this approach provides a much simpler software environment in which the processors are controlled as a set, rather than individually, and thus the program complexity remains the same as the tree size increases.

For fast, economical, and reliable implementation of Command and Control decision aid systems, such an emphasis on simplicity in software structure is essential.

The major parallel construct is a *sliced procedure*, in which identical programs are executed simultaneously in each PE, but on different data sets. The multiple executions of this single program can follow different instruction streams, (though within the same program), depending upon the data. These potentially different instruction streams are forced to converge and synchronize at the completion of the sliced procedure, (also called a barrier synchronization). This concept can be described as Single Program Multiple Data (SPMD) processing. Within the classical SIMD and MIMD concepts defined by Flynn [6], the SPMD can be viewed as data-driven MIMD, or alternatively as coarse-grain SIMD.

In support of the SPMD processing mode, we use several global burst communications functions that extend the concepts developed by Bentley and Kung [7]. These communications are invoked by function calls from the host and are examples of traffic-specific interprocessor communications. The interconnection hardware and software has been designed to optimize these global operations, applying the concept of "reduced instruction set" to interprocessor communications. *Broadcast* transmits data from the host to enabled PEs. *MinResolve* identifies the PE in the tree with the minimum value of some variable. The *Report* command, typically following a *Resolve*, transmits data from the identified PE to the host. There are several variants of each, covering integer and real data types, and single and block data sets.

This software design provides another type of scalability to tree machines, namely program scalability. The data flow within the tree is controlled by the host independent of the size of the tree, and each PE runs the same code. Therefore, the source code for an application (e.g. pattern matching) is essentially unchanged when additional patterns and additional PEs are added. This software scalability makes a tree processor easy to program. For pattern recognition in particular, so long as the number of PEs is scaled proportional to the problem size, it will be shown that the execution time remains essentially the same. The communications complexity grows, in the abstract, only as $\log_2 N$. In many practical situations, this term appears only as a pipeline fillup time. Examples include block broadcasts and reports, in initiation of sliced procedures, and in k-best resolve operations for large k. This leads to the concept of scalable algorithms, i.e., ones in which the source code remains the same as problem size grows, and in which execution remains the same if one scales the hardware proportionally.

APPLICATIONS

A brief description of a number of example applications is useful to explain some of the concepts outlined above.

Before expositing specific examples, a comment regarding the use of an binary tree structure as an associative memory is in order. In one-dimensional match problems (e.g. correlating radar returns with targets in track, with range as the only dimension), the processing load in a conventional uniprocessor implementation can clearly be reduced by sorting the lists to be matched prior to comparison, and in the radar example, range sorting is a *de facto* consequence of the receipt of returns. In many modern Command and Control problems, however, data items are characterized by multidimensional descriptors (e.g. in signal detection: frequency, bandwidth, stability, amplitude, azimuth, etc.) such that one-dimensional lists do not suffice.

Solving a set of N linear simultaneous equations, as in the core of the simplex or Karmarkar approaches to linear programming, can be accelerated by the number of PEs using a paradigm similar to that used for pattern recognition. Assume a distinct equation is stored in each PE. (This assumption is generalized by using the notion of *virtual PEs* where multiple virtual PEs are mapped onto each physical PE.) Then the inner loop of Gaussian elimination comprises the following steps:

a. *Resolve* to find the largest coefficient in the next column of the matrix to be processed (i.e. identify the pivot element);

b. *Report* the equation in which that coefficient appears to the root PE, and *Broadcast* that equation to all PEs.

c. Then all PEs perform simultaneous pairwise eliminations of the variable in question (i.e. corresponding the column selected) using the resident equation and that just broadcast.

The result after N iterations of the above loop is the complete solution, rather than an upper triangular matrix.

If communication cycles to move a number one layer up or down the tree structure as part of the broadcast and resolve operations occur at the same rate as the arithmetic operations (add, multiply), then this approach yields an acceleration of .65 N times as fast a single PE not required to perform these communication steps.

K-means clustering is one of several algorithms for starting with a scatter-plot of data (e.g. templates representing different pronunciations of the same word in speech recognition, detected signals in EW or ASW) and linking the data items into probable groups based on some distance metric. The algorithm involves an inner loop as follows. Assume the data vectors to be clustered are distributed across the PEs, and that K tentatively selected centroids are held in the Host. (Again, for description purposes, the data vectors are held one each in virtual PEs, with as many virtual PEs per actual PE as the data set requires.) The Host *Broadcasts* the centroids one at a time, and the distances to all data vectors are computed in parallel in the respective PEs. The resultant list of K such distances in each virtual PE at the end of this part of the inner loop are scanned to find the minimum, i.e. find the centroid and hence cluster with which that data vector is associated. Centroids are then recomputed via summation of all vectors just associated with each cluster. Convergence of this process leads to a solution for the K clusters. K may be stepped through multiple values, and these values are evaluated by computing the ratio of intra-cluster to inter-cluster variance, a process also amenable to parallelization. Assuming the number of PEs, N, exceeds the number of data items, again an approximate N-fold acceleration occurs.

Parallel dynamic programming is applicable to a wide range of structural pattern recognition operations. In [8], the application of parallel dynamic programming to speech recognition is described. Dynamic programming applied to structural pattern recognition involves a graph search with following steps: evaluate alternative paths, select the best, then make the transition. Evaluation of alternatives is performed in parallel on BT-100, and selection of the best path(s) is performed in log time by the tree.

References

1. Duda, R. O., and Hart, P. E. *Pattern Classification and Scene Analysis*, Wiley, 1973

2. Rabiner, L. R., and Levinson, S. E. "Isolated and Connected Word Recognition--Theory and Selected Applications", *IEEE Trans. on Communications*, vol. 29, no. 5, May 1981

3. Leiserson, C. E. ,"Area-Efficient VLSI Computation", Ph.D. Thesis, Carniegie Mellon University, 1981

4. Boddie, J. R., et. al., "A Floating-Point DSP with Optimizing C-Compiler", *ICASSP88*, (Session 31.V2), April, 1988

5. Hartung, J., et al, "A Practical C-Language Compiler/Optimizer for a Family of Floating-point DSP's", *ICASSP88*, (Session 42.d8), April, 1988

6. Flynn, M. J. "Very High-Speed Computing Systems", *Proc. of the IEEE*, Vol. 54, no.12, Dec. 1966, pp1901-1909

7. Bentley, J. L. and Kung, H. T., "A Tree Machine for Searching Problems," Technical Report, Dept. of Computer Science, Carnegie-Mellon University, Sept. 1979

8. Gorin, A. L. and Shively, R. R., "The ASPEN Parallel Computer, Speech Recognition, and Parallel Dynamic Programming," *ICASSP 1987*, pp. 976-979

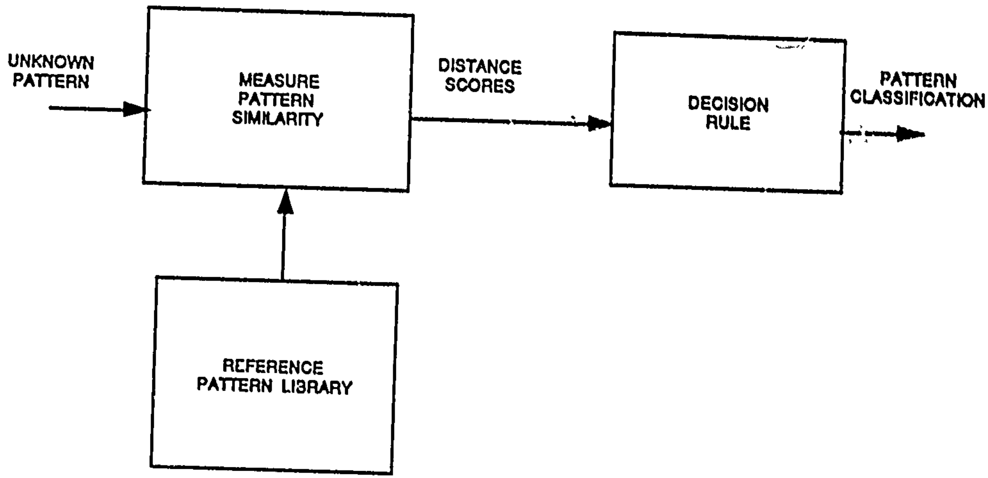


Fig. 1: Pattern Recognition Information Flow

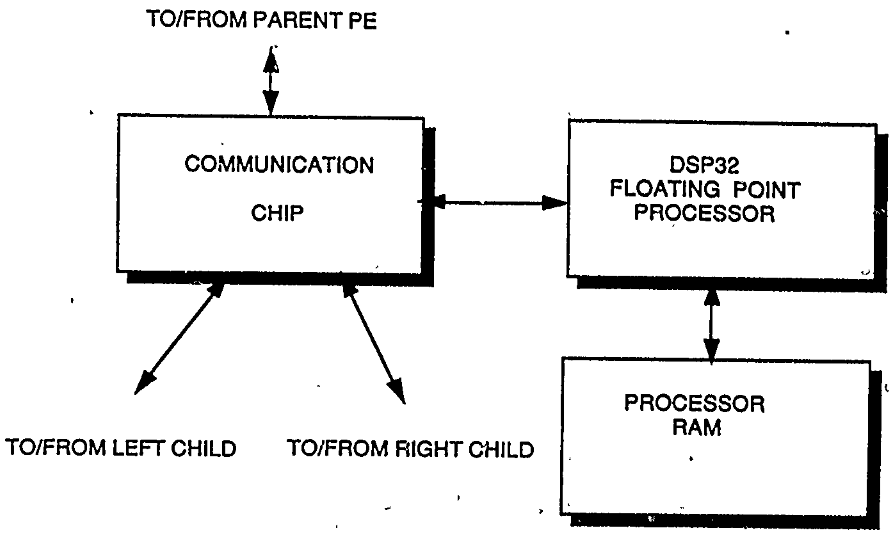


Fig. 2: Processing Element Structure

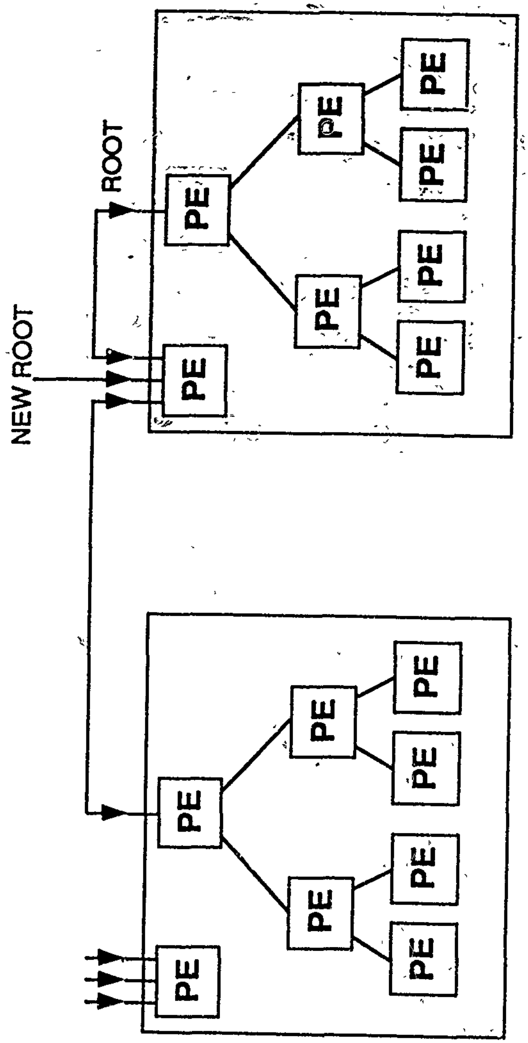


Fig. 3: Tree Expansion Scheme

THE POTENTIAL APPLICATIONS OF NEURAL NETWORKS AND NEUROCOMPUTERS IN C3-I

RONALD E. WRIGHT

CHIEF ENGINEER, NEW VENTURES DEPARTMENT,
FERRANTI COMPUTER SYSTEMS LIMITED,
BRACKNELL, BERKSHIRE, ENGLAND

ABSTRACT

One A.I. technology, the Expert System, is finding practical application in C3-I systems. However, it can be argued that Expert Systems do not exhibit intelligence and that any rule-based system is limited and even vulnerable. In the last few years the "New A.I." or "Sixth Generation Computing" has emerged, based on the brain-like properties of highly connected processing elements. This may represent the basic technology of intelligence, and has the potential to have a dramatic impact on C3-I systems, allowing autonomy and adaptability. However, little is really known about how to design and structure such systems and the physical technology has yet to be evolved. This paper suggests areas of application of connectionist computing architectures, but concludes that at this time the requirements of C3-I are ahead of the available technology. It suggests areas where research should be focussed, partly from the background of the European ALVEY and ESPRIT Information Technology research programmes. It now appears certain that the technology will evolve, if gradually, and C3-I systems will change dramatically, and with it the roles of the associated human operators.

DEVELOPMENT OF NAVAL COMMAND AND CONTROL SYSTEMS

Since the 1950's we have seen the increasing use of digital processing techniques to present a tactical situation picture to the ship's combat teams, but these systems give little support to the crew to help them assess the situation and make approriate command decisions. The human operator is still an essential component in most current Command and Control systems. However, we are beginning to see applications where the situation or performance requirements preclude the use of human operators-in-the-loop, for example in the control of ship's close-in-weapon systems (CIWS) and space Battle Management station for Ballistic Missile defence.

The technology of the Fifth Generation of computer systems, which addressed the representation and processing of knowledge, appeared to offer a means of providing intelligent support in Command and Control processes, particularly by the use of Expert Systems [Ref 4]. Papers presented at the earlier MIT/ONR workshops [Refs. 1 to 3] have described the use of Expert Systems in command tasks such as task force disposition at sea, flying programmes and EW management. These systems have a rule-based approach, and enable procedures and experience to be built into a system. Expert Systems are thus really a design discipline; they do not give the system a creative intelligence. In fact, unless they are carefully designed they may give the system an unwanted rigidity and predictability. Over the last five years such systems have been trialled, with both the British and US navies; but we are still some two to three years away from having such systems incoporated in operational equipment.

Naval officers and their children now look to digital computers, usually in the form of a pocket calculator, for their numeric calculations. For our symbolic expressions the mighty pen is being replaced by the more powerful word processor. Soon we can expect the naval officer to look to his Expert System as his planning and diagnostic aid. What we now seek is help for the operator in his other creative functions, such as situation assessment and decision making. Such help appears to be at hand in the form of neural networks or neurocomputers.

What properties has the human brain got that makes it so essential in Command and Control?

It has awareness of what is going on, partly by reference to its own built-in world model.

An ability to focus on the essentials of a situation, direct its attention selectively, work with incomplete or uncertain information.

It has short and long term memory, with an ability to associate relevant information. It is particularly good at recalling temporal and special patterns and sequences, such as music and road routes.

It can exercise judgement and make decisions, responding to real-time dead-lines.

It has wishes and works towards goals - it cares!

It is adaptive and creative.

On the negative side:

It can be distracted and get tired and inattentive.

It can become confused and get the wrong end of the stick (sometimes because of limitations of its world model).

It can be emotional and illogical under stress.

It is fallible.

It can become ill, be injured and die (for example as a result of NBC warfare).

It needs a finite time to respond.

It comes complete with a body that has to be fed, watered and housed.

So, in some military systems we need to augment the human brain or replace it. Here can I introduce my First Law of Neural Networks:

"Human Intelligence will only increase by improved human teaching techniques and the development of real creative artificial intelligence."

Figure 1 shows the the human brain in partly exploded form. There is nothing else like it in the known universe. As this illustration shows, it has a highly complex architecture. It has been studied and probed by specialists from many disciplines and a vast quantity of research papers and books written about it.

But if we ask the question: "Do we really know how it works?", the honest answer is that we only have a few clues. [Refs. 5 and 6].

One level at which some progress has been made is at the level of the parts, particularly the nerve cell or <u>neuron</u>. This little fellow (Fig. 2), which is just about resolvable by the human eye, has inputs, the dendrites, by which it receives information and a cell body which is the processing element, which processes the information to give an output (in the form of an electrical pulse) which it sends to other neurons and cells along its single axon. The axon separates into a number of small fibres that have terminals or buttons Each of these terminals forms a functional connection to another cell, the synapse.

As far as we know a neuron communicates with other neurons (or muscle or gland cells) only by way of these tiny synaptic junctions. A given neuron in the brain may have several or many hundreds or connections to other neurons.

If the human brain has 10^{11} neurons

It might then have 10^{14} synapses

And factorial 10^{14} as the number of possible connection arrangements. It appears that it is in this massive potential interconnectivity that the brain has its ability to store and process information

One approach to understanding how the brain functions is by computer modelling at the neuron level. Figure 3 shows a simple model of a single neuron having four inputs and one output.

The inputs are connected to summing units through a device that applies a set of synaptic weights. The output depends on the magnitude of the inputs modified by the synaptic weights and the transfer characteristic of the "summer", which represents our neuron cell body or processing element. The neuron fires with its output going high when the summed inputs exceed a threshold This neuron could be adjusted or "trained" to fire, for example, when any two or more of its inputs are in the high state. It is a simple matched filter.

An artificial network can be formed by interconnecting a number of neurons (Fig. 4) and current research work is directed to defining types of artificial network and network elements and investigating how such networks can be adjusted or trained to do specific functions such as pattern recognition. For example, the network illustrated can be adjusted to classify a four bit input pattern.

Artificial networks are not constrained to just mimic the networks and components of the human brain. A number of types of network have been postulated and evaluated, mainly by modelling on some form of special processor or supercomputer which at some level uses serial Von Neumann type operations.

Such networks (connectionist networks) are typified by their:

network topology
activation rule
learning (programming) rule
dynamics

There is now a considerable but confusing amount of literature describing over thirty different classes of networks. [Refs. 7 and 8].

The above discussion has used the terms neural network and neurocomputer without defining them. It appears to be generally accepted that a neural network is the interconnected set of physical components such as has been illustrated in Figure 4, and it could be put in a black box with a set of input and output terminals. A neurocomputer will contain one or more neural networks, but also provides its user with means of specifying, controlling, running and monitoring the neural network. A virtual neurocomputer is similar, but the neural network is modelled or emulated rather than being a direct physical representation, and the modelling may not be in real-time

<u>RESEARCH OBJECTIVES</u>

It appears certain that Neural Network technology will have an impact on future defence systems, but where they will be best applied and how they will be realised technically is a matter of some speculation [Ref. 9].

The pragmatic designer of Command and Control systems will want to know:

Where neural networks can advantageously be applied.

Which type of network to use in a particular application.

How to design networks into a system and predict their performance (architectures, tools and design methodology).

How to implement the networks in hardware and programming terms.

How to test and qualify the resulting system.

One approach would be a top-down functional decomposition of the system requirements, provided that some means of identifying the functions which lend themselves to neural network implementation could be established. The difficulties of dividing the C2 process into component parts were addressed by Levis and Athans last year [Ref. 10]. However, the simple conceptual model of Fig.5 will serve our present purposes We can anticipate that in the immediate future progress can be made by contacts between people conversant with C-2 design and people who are beginning to get an appreciation of the applicability of different types of Neural Network from hands-on modelling. What is needed is an explicit design methodology. As a contribution I offer my Second Law of Neural Networks:

"Any process which is better performed by a human or animal brain than by a computer is a potential application for Neural Networks".

I leave you to apply this law to the Command and Control processes.

The rest of the system designers' requirements are still in a process of evolution, and indeed some aspects and problems have not yet been defined or addressed. They thus form an area for research and development. This is not to say that practical implementations of Neural Networks are not already possible. In fact, it has been the success of some experimental applications of neural networks in areas where conventional A.I. and symbolic and numeric computing has been found wanting that has led to the current resurgence of interest in Neural Network techniques.

One way in which progress can be made is to focus expertise in Neural Networks onto one or more potentially realisable applications. Fortunately the second phase of the European Strategic Programme for Research and Development in Information Technology (ESPRIT) is being launched in 1988 and a number of consortia of industrial and research organisations have proposed projects involving the study and demonstration of the applications of Neural Networks. Among the applications proposed are data fusion, situation assessment, decision support, planning and optimisation, all directly relevant to Command and Control processes. The UK Government is also initiating a programme of collaborative research, as a successor to their earlier ALVEY programme, which includes an investigation of neural computing. In 1987 DARPA initiated a study at MIT's Lincoln Laboratories into the potential defence applications of Neural Networks, with a view to defining a major research programme.

All these initiatives indicate that most of the needs of the system designer are likely to be researched and developed. One downstream problem, which may not yet be receiving the attention it deserves, is how you test and validate a system incorporating Neural Networks. Conventional computer systems are vulnerable to failure of a single computer component or software bug, leading us to resort to redundant dissimilar software. Neural Networks, by their massively parallel organisation, are inherently tolerant to single or even multiple component failures. However, they are usually programmed by training procedures, so that different neural computers trained for the same task, may not have identical structure and parameter values. Even if the design is performed on a virtual neurocomputer, and the design is manufactured by replicating identical neural networks and parameter values, it is often by no means apparent how the network is organised and what algorithms it is using. If we give our quality controller a listing of, say, ten million interconnections, what is he going to make of it?

We ourselves have some insight into this problem, as people are a form of neurocomputer and we have been training them and assessing their capabilities for years. So our neurocomputer can be trained, tested, and gradually promoted to having increased responsibility in a similar way to human operators.

So, I introduce my Third Law of Neural Networks, namely:

"A Neural Network/Neurocomputer is entitled to the same consideration and promotion as its human counterpart."

CONCLUSIONS

Neural Network technology appears to be complementary to numerical and symbolic computing and offers the potential of helping or even replacing the more creative activities of human operators in C3-I systems. Similar expectations were held for knowledge based systems, but these have not been fully realised. Artificial Neural Networks, because of their analogy to the neurons of the human brain, must have the potential for ultimately producing artificial creative intelligence. The immediate future offers the possibility of mechanising applications which can be based either on a low level of neural network implementation or networks of such a size as can be represented on available virtual neurocomputers. However, our current dominant computer technology is based on two-dimensional planar semi-conductor technology, which does not lend itself to massive interconnectivity. Thus large fully-implemented neural networks await major developments in component technology.

The brain is highly structured at various levels, and these higher level architectures are not fully understood. Most models of neural networks assume a uniform homogeneous structure, and the significance of heterogeneous networks and higher level architectures has yet to be determined. Such considerations indicate that the development of Neural Networks to extend the areas of mechanisation of C3-I will be a protracted and difficult task. However, Neural Networks may eventually represent the high ground of Command and Control technology.

The Japanese, with their study of automated brain functions in their ambitious "Human Frontier Science Programme", have quietly introduced the Sixth Generation of computing.

So I introduce my Fourth Law:

"There will be no further computer generations after the Sixth, just extensions of it."

We have reached a final frontier in man's computing endeavours. In a few years time your battle commander will reach for his neurocomputer, or more likely speak to it, whenever he has a tactical decision to make.

Moreover, as our understanding of the mechanisms of the mind increases, it will become apparent that we ourselves are just robots; complex and capable as we are. Hopefully by then the idea of resolving conflicts by engaging in battles of robots against robots will have become a thing of the past, and perhaps conflicts will be resolved by some omnipotent authority, possibly in the form of a multi-national neurocomputer called Solomon.

REFERENCES

1. Wright, R E. "An Overview of How Expert Systems May Be Applied in Naval Command and Control Systems" Proceedings of the 8th MIT/ONR Workshop on C-3 Systems, 1985. MIT Reference LIDS-R-1519, pp. 7-13..

2. Gadsden, J.A. and Lakin, W.L. "FLYPAST : An Intelligent System for Naval Resource Allocation". Proceedings of the 9th MIT/ONR Workshop on C3 Systems, 1986. MIT Reference LIDS-R-1624, pp. 139-144.

3. Lakin, W.L. and Miles, J.A. "Intelligent Data Fusion and Situation Assessment". Proceedings of the 9th MIT/ONR Workshop on C3 Systems, 1986. MIT Reference LIDS-R-1624, pp. 133-138.

4. Taylor, Edward C. "AI in Command and Control : What and When". Proceedings of JDL Command and Control Research Symposium, 1987. Reference SAIC-87/1119, pp. 379-384.

5. Ornstein, Robert and Thompson Richard F. "The Amazing Brain" published by Chatto & Windus, London 1985. ISBN 0-7011-3989-8. Paperback.

6. Searle, John "Mind, Brains and Science" published by the British Broadcasting Corporation, London 1984. ISBN 0-563-20222-1. Paperback.

7. "IEE First International Conference on Neural Networks" published by SOS Printing, San Diego, June 1987 Four Volumes. IEE Cat. No 87THO191-7.

8. Eckmiller, Rolf and Malsburg, Christoph v.d. , ed. "Neural Computers", published by Springer-Verlag, New York in co-operation with the NATO Scientific Affairs Division, 1988. ISBN 0-387-18724-3. NATO ASI Series, Series F, Vol. 41.

9. North, Robert L. "Neurocomputing : Its Impact on the Future of Defence Systems", Defense Computing, Jan-Feb 1988, pp. 18-23 and 66

10. Levis, Alexander H. and Athans, Michael. "The Quest for a C3 Theory : Dreams and Realities". Proceedings of JDL Command and Control Research Symposium, 1987. pp. 7-11. Reference SAIC-87/1119.

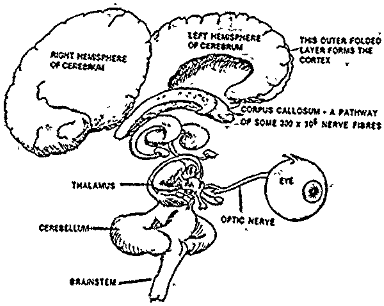


FIGURE 1 - THE HUMAN BRAIN

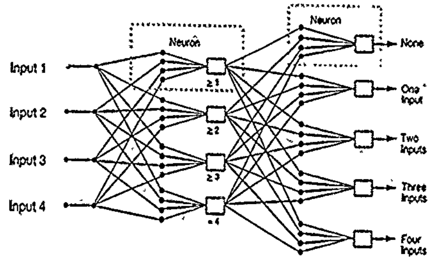


FIGURE 4 - TWO LAYER ARTIFICIAL NEURAL NETWORK

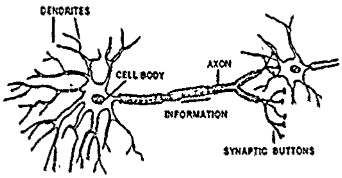


FIGURE 2 - A TYPICAL NEURON

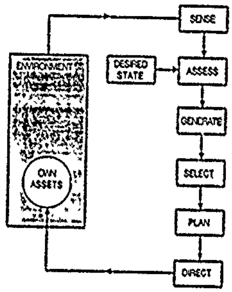


FIGURE 5 - A SIMPLIFIED CONCEPTUAL MODEL OF THE C^2 PROCESS

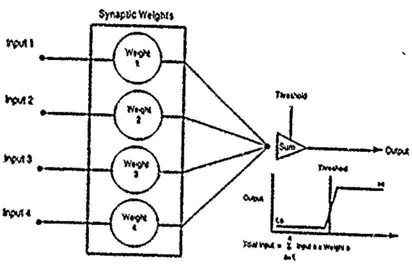


FIGURE 3 - SINGLE LAYER ARTIFICIAL NEURON

ROUTING IN RECONFIGURABLE RADIO NETWORKS

A. Ephremides

University of Maryland
Electrical Engineering Department
College Park, MD 20742

ABSTRACT

In radio networks that are subject to unpredictable topological change, it is necessary to develop algorithms for effective routing of broadcast and/or point-to-point messages.

In the past we have addressed the problem of link activation and shown that it is a NP-complete problem with a good heuristic that can be implemented in a distributed fashion.

In this paper we consider the question of routing. First we show that the problem of routing is related to that of link scheduling. Also we consider separately the cases of broadcast versus point-to-point traffic.

In both cases the problem acquires new dimensions when the volatility of the topological connectivity map is introduced. When the rate of changes is extremely fast the only available solution is flooding. When the rate of changes is too slow, there are several shortest-path algorithms that can be used for extensive periods between changes. The interesting case is when the rate of changes is moderate. We develop an algorithm that is adaptive and deadlock-free that relies on a query-response exchange process.

INTRODUCTION

The problem of routing in radio networks differs from routing in point-to-point, hard-wired networks in two important respects: 1) the topology and connectivity structure of the network is more volatile and can change frequently owing to a variety of factors, such as link/node jamming, antenna orientation, node motion, etc., and thus the routing tables need to be updated more frequently and, more importantly, through the use of the variable connectivities themselves. 2) The implementation of a routing strategy must take into account the imposition of interference-free schedules for link activation and transmission; thus, a particular path may cease to be preferable because of increased delays that the schedule of transmissions imposes, in other words, the choice of a path and the choice of a transmission schedule depend on each other. We examine the problem of routing with respect to these two new perspectives.

1. VARIABLE CONNECTIVITY

If the rate of changes in the topology of a network is extremely high, little can be done in terms of choosing good routing paths. The only alternative is flooding. Thus each source node broadcasts its message to all of its neighbors and each of the nodes receiving this message rebroadcast it once, and so on. Eventually, if the intended receiver has not been disconnected from the source as a result of these changes, the message will reach its destination.

On the other hand, if the rate of changes is extremely low, there exist several good algorithms [1] for point-to-point routing such as the distributed version of the Bellman-Ford algorithm, which can be invoked and utilized in the radio network. These algorithms, in addition to permitting alternate routing in case of occasional failures, have good overall performance properties, since they converge to minimum distance (or delay) paths. The way in which they achieve automatic and distributed recovery (i.e., routing) from failures is based on a carefully ordered of control message exchanges that was first described in [2].

The interesting and unsolved case occurs when the rate of changes is neither too high nor too low. In this intermediate situation flooding is neither necessary (because there is sufficient time between connectivity changes to permit systematic searches for better strategies) nor desirable (because it creates unnecessarily numerous multiple transmissions that overload the network, increase the delay, and reduce the throughput).

On the other hand, the invocation of the Bellman-Ford or Merlin-Segall types of "good" algorithms is not sufficient because the rate of convergence of these algorithms might be too slow compared to the rate of topological changes. Also, the rather substantial overhead in terms of control messages that the implementation of these algorithms requires would also tend to overload the network.

Thus, there is a need for creative new schemes to address the routing problem in these intermediate situations of connectivity variations. Some algorithms that are based on hierarchies and clusters have been discussed recently in the literature [3], but none have been sufficiently validated to enable an assessment of their suitability for the environment of interest described here.

A new algorithm has been proposed that is based on a simple idea that was described first in [4] and that shows promise for the case of interest. To describe this algorithm it is useful to distinguish two cases. First, consider the case in which the source node (say i) has zero connectivity knowledge about the entire network when it wishes to transmit a message to a particular destination node (say j). Second, consider the case in which node i discovers that the connectivity information on which it was basing its routing decisions has undergone some unknown change.

In the first case the idea of the algorithm is to flood a query message that seeks to locate node j. The first node that possess information about j's whereabouts responds by flooding a reply message that eventually reaches i (and, more importantly, all other intermediate nodes which although at the moment are not interested to locate j, do so inadvertently, and therefore possess the required knowledge later on if and when the need should arise. Subsequently, node i utilizes the "discovered" path to set up a virtual circuit for routing messages to node j (incidentally, there may very well be numerous alternative paths that are so discovered as a result of the query-reply process) until it is determined (in a manner to be specified) that this path is no longer valid (or, even, desirable).

This brings us to the second case, in which node i possesses a wealth of information about overall network connectivity (as a result of the ongoing query-reply phases in which it has participated), but knows that the path to j is not valid anymore owing to one or more topological changes that have occurred. In this case, node i floods a new query, again seeking to locate node j. the tempting thing to propose is that in this case any node that possesses what it believes to be valid information about j generates a reply as in the original phase and thus node i acquires a new path based on this reply. In fact one could argue that if the topological change was localized, the alternative path will be discovered very promptly and with very little overhead in terms of control message exchanges. However, more care is required because this procedure can lead very rapidly to loops, unless special provisions and safeguards are introduced.

There are several possible provisions that cannot be comparatively evaluated a priori. One possibility is to treat the new request message from node i as in the original zero-knowledge phase. The advantage of this simple "fix" is that the danger of loops disappears, because the original procedure can be, indeed, shown to generate loop-free paths. The disadvantage is that it negates the very intent of the first phase which was to "build" an information base that will be useful and relied upon when minor perturbations of the network connectivity map take place.

Al alternative method is to use the structure of the Merlin-Segall protocol [2] "in reverse," in the sense that the source, rather than the destination, generates a "distress" message that is propagated across the network with a similar careful ordering of transmissions in order to achieve the discovery of a loop-free path that by-passes the location of the fault.

Both alternatives have been considered and are being currently incorporated in the code that is under development in the Communications and Signal Processing Laboratory of the University of Maryland. In both cases there are minor variations depending on what information is included in the query and reply message format. For example, in the query message it is possible to incorporate only the destination ID and the sequence number of the search or include also the source ID. Similarly, in the reply message it is possible to include an estimate of the distance (or, more simply, the number of hops) from the destination in order to permit optimization in the selection of routing paths. In both cases, it is essential that time stamps be included in order to implement time-outs and decisions about when to retransmit in the absence of a reply.

To evaluate and validate a distributed algorithm such as the one just outlined it is necessary to simulate it and measure its performance, but also to compare it to the alternatives of flooding and fixed-topology routing. This task is not simple. What is needed is a simulation testbed that is well-suited to the nature of the algorithm and, also, a variety of representative topological variation scenario in the presence of which the algorithm and its rival alternatives must be evaluated.

Fortunately, a software tool which is designed to assist distributed simulations, called JADE, has been developed at the University of Calgary, has been tested on similar tasks by the Information Technology Division at the Naval Research Laboratory, and has been acquired by the University of Maryland and is now available to us. This tool is essentially a distributed interprocess communication protocol that has powerful monitoring and synchronization capabilities. It enables us to perform preliminary simulations of these algorithms in order to assess some of their basic properties. Subsequently, a full-scale performance evaluation will be needed which will take place at NOSC and which will be performed on the NASTEE testbed developed there.

2. CONNECTION TO LINK ACTIVATION

In last year's Symposium we presented a study of the problem of distributed link activation and scheduling subject to non interference constraints and provided a distributed heuristic for good schedules, given that the optimal scheduling problem is NP complete. Any schedule of link activation in a multi-hop radio network involves the reuse of channel resources by nodes that are sufficiently apart and forces TDMA-like transmissions among nodes that are only one or two hops apart. These TDMA transmissions inject delays between periodic uses of the channel by a given node, especially if that node has a high degree of connectivity (i.e., a large number of neighbors). Quite obviously then, the inclusion of a particular link on a routing path must take into account the additional delay induced by the schedule. Conversely, the formulation of the scheduling problem must take into account the end-to-end traffic requirements and the link flows that these requirements and an associated routing regime induce. In [5], the link scheduling problem was formulated based on simplified broadcasting traffic requirements that were uniform among all nodes and took into account only one-hop transmissions among neighboring nodes.

Here we attempt to provide the ultimate formulation of the joint routing-scheduling problem. As a first-step, we start from the case in which a given number of messages reside at the source nodes in the network and we wish to develop a schedule of transmissions and a set of routes that will permit the delivery of all of these messages to their destinations in minimum time.

The reason that this formulation is interesting is that the original scheduling problem considered in [5] proved to be amenable to an efficient solution by means of Hopfield neural nets [6]. In fact, the formulation of the augmented-routing-scheduling problem seems not only to be amenable to a similar solution implementation (which, it could be argued, has only limited interest because it is a centralized rather than a distributed one) but also to permit a distributed implementation by parallel use of multiple neural nets, each one performing its part of the computation at one associated node.

Suppose that a network described by a graph $G = (N, A)$, where N is the set of nodes and A is the set of links (ordered pairs of nodes), finds itself in an initial condition s_0 described as a matrix with enties $x^0_{ij}, k, \ell = 1, \cdots, N$, where x^0_{ij} is the number of messages (packets) which at time 0 reside in node i with ultimate destination j.

We are interested in obtaining a schedule of transmissions that will lead the network through a sequence of states $s_0, s_1, s_2, \cdots, s_k$ to a final state s_k in which $x^k_{ij} = 0, \forall i, k$, for minimum k. A schedule consists of a sequence $C_n n = 1, 2, \cdots$ of triplets (A_n, D_n, d_n), where A_n represents a broadcast mode in time slot n, that is the set of nodes that are allowed to transmit in that slot, D_n represents the set of destination nodes that corresponds to the set of transmitting nodes A_n in slot n, that is it identifies by destination, the type of message that each node in A_n must transmit in slot n, and d_n represents the set of immediate neighboring nodes to which the corresponding transmissions should be directed, that is it represents the set of next-step destinations for the single-hop transmissions that will occur in slot n.

Clearly, we are interested in obtaining a sequence $S_0, S_1, S_2, \cdots, S_k$ that does not have repetitions and achieves minimum k through the schedule sequence $C_1, C_2, \cdots, C_k$. What is important to note is that the dynamic programming principle applies. Namely, if the sequence $C_1, C_2, \cdots, C_k$ is optimal and corresponds to the sequence of transitions $S_0, S_1, S_2, \cdots, S_k$, then the sequence $C_2, \cdots, C_k$ must be optimal if the network starts from initial state S_1. Of course, an optimal sequence must exist since the problem is finite but there need not be a unique such optimal solution.

This observation leads to a classification of the possible states of the network in terms of layers L_k. A state S belongs to layer L_k if it can reach the "empty" all-zero state in a minimum of k transitions. Thus, the problem of determining the minimum time for emptying the network from an initial state corresponds to finding the layer to which the initial state belongs. Furthermore, the passage from one layer to the next is achieved by one element of the schedule sequence we are seeking.

The problem is again NP-complete. The fact that a neural net can be constructed to achieve the minimization for the much simpler problem of determining only a maximal mode A_n suggests that it may be possible to construct such a net for solving the combined routing-scheduling problem. So it is of interest to examine how the Hopfield net can be constructed and what its properties are.

Of course it should be noted that solving the "empty-the-network" problem does not imply that the general routing problem is solved. In the general routing problem there is a continuous flow of external inputs (or of new packet generations) at each node. The objective then is to minimize the expected total time of message delivery to the destination. Of course, the two problems are related and it would be of great interest to explore their relationship.

A Hopfield neural net is a dynamical system described by the following system of equations:

$$\frac{du_i}{dt} = -\frac{u_i}{\tau} + \sum_{j=1}^{N} T_{ij} g(u_j) + I_i, \qquad i = 1, \cdots, N$$

where the I_i's, τ, and the T_{ij}'s (which satisfy $T_{ij} = T_{ji}$) are constants and the function $g(\cdot)$ is a nonlinear function whose shape resembles that of a soft limiter, i.e., it starts at zero at the origin and saturates to 1 at some value u_0 of its argument. The remarkable property of this system of equations is that for any initial condition $[u_1, (0, u_2(0), \quad , u_N(0)]$ the limit of $g(u_i(t))$ as $t \to \infty$ is either 1 or 0, for any $i = 1, \cdots, N$. Another important property is that among all 0-1 combinations in the sequence $g(u_1), g(u_2), \quad , g(u_N)$ the ones favored by the system are the ones that produce local minima of the function

$$f(v_1, v_2, \cdots, v_N) = -\frac{1}{2} \sum_{i=1}^{N} \sum_{j=1}^{N} T_{ij} v_i v_j - \sum_{i=1}^{N} v_i I_i$$

for

$$0 \leq v_i \leq 1, \qquad i = 1, \cdots, N$$

To make the problem of link activation correspond to the minimization of a function of the form we just examined (and, consequently, to hope for a neural net implementation of the solution) we need to identify that function. In the Hopfield model the index i corresponds to one of N interconnected neurons, and T_{ij} takes a value that corresponds to the "strength" of the connection between neurons i and j. In our problem we identify each node with a "neuron" and let

$$T_{ij} = \begin{cases} 1, & \text{if } i \text{ and } j \text{ are 1 or 2 hops away from each other} \\ 0, & \text{otherwise} \end{cases}$$

where $a < 0$, and we let $I_i = I > 0$.

Also, we let $g(u_i)$ be one or zero depending on whether node i transmits in a given slot or not. It is easy to see then that if $|a| > |I|$ the stable states of the neural net are those that correspond to maximal independent sets of nodes in the radio network connectivity graph. But finding maximal independent sets is equivalent to finding "good" link activation schedules (see [5]).

The interesting thing in this correspondence is that, although the neural net function minimization takes place over a continuum and the link activation problem requires a combinatorial maximization over a finite set, the property of the neural net that forces its equilibria points on the discrete set of binary sequences makes it possible for the solutions to the two problems to coincide.

Also, since in a radio network a given node is not affected by those nodes that are not connected to it or to its neighbors, the equations that need to be solved (usually by relaxation that simulates the net, which one could also build in hardware) by each node are "local" since they only involve the u_i's that correspond to neighboring nodes. This observation permits the important conclusion that the neural net solutions can be implemented distributedly.

CONCLUSIONS

In this paper we brought forth these aspects of the routing problem that distinguish radio nets from non-radio networks. We outlined an algorithm for routing under topological variations and we formulated the joint link activation-routing problem in terms of a combinatorial optimization problem that can be efficiently, and distributedly, solved via neural nets.

REFERENCES

1. D Bertsekas, R. Gallager, Data Networks, Prentice-Hall, 1987.
2. A Segall, "Advances in Verifiable Fail-Safe Routing Procedures," IEEE Trans. Comm., Vol. 29, pp. 491-497, 1981.
3. N. Shacham et al, "Study on Distributed Processing System Topology for Battle Management/C^3 System," SRI report-project 1041, September 1987.
4. M. Weber (Bates), A. Ephremides, "A Simulated Performance Study of Some Distributed Routing Algorithms for Mobile Radio Networks," Proc. Johns Hopkins Conference, Baltimore, MD, March 1983.
5. A Ephremides, "The Problem of Distributed Scheduling of Radio Links," Proc. C^2 Research Symposium, Washington, DC, June 1987, pp. 191-197.
6. J. Hopfield and D. Tank, "Neural Computation of Decisions in Optimization Problems," Biological Cybernetics, Vol. 52, pp. 141-152, 1985.

INTERCONNECTION ALGORITHMS IN MOBILE C^3 TOPOLOGIES

P. Papantoni-Kazakos, Michael Paterakis and Liu Ming

Department of Electrical Engineering
Thornton Hall
University of Virginia
Charlottesville, Virginia 22901

Abstract

We consider a two-cluster multi-hop packet radio network. Each cluster employs a limited sensing random access algorithm, and contains local users who transmit their packets only via the algorithm in their own cluster. The system also contains marginal users, who may transmit their packets via either one of the algorithms in the two clusters.

For the above system, we adopt a limited sensing random access algorithm per cluster that has been previously studied. This algorithm utilizes binary, collision versus noncollision, feedback per slot, and under the Poisson user model and in the absence of marginal users its throughput is 0.43. We consider a dynamic interconnection policy for the marginal users, and we then study the system performance. Specifically, we determine the stability region of the system and the packet expected delay. The proposed interconnection policy accelerates the marginal users, presenting them with a significant delay advantage over the local users. This is desirable when the marginal users transmit high priority data.

I. INTRODUCTION

We consider mobile packet radio C^3 topologies, where dynamic user clustering evolves. The users in each cluster utilize a common single channel for transmission, and they communicate with each other via a limited sensing random access algorithm (LSRAA). As the topology evolves dynamically, and for better system connectivity, neighboring clusters may overlap. Users located in the overlapping regions are then exposed to transmissions and feedbacks from more than one clusters, a generally time-varying phenomenon due to mobility of the users which can be exploited for the improvement in performance of the overall system. Let us, for example, consider two overlapping clusters, and let us call the users in their overlapping region marginal users; let us call the users in cluster i, i=1,2, which are not located in the overlapping region, local users for cluster i. The local users in cluster i, i=1,2, comprise a Poisson traffic with intensity γ_i, i=1,2, and they communicate via the limited sensing random access algorithm LSRAAi, i=1,2. The users who are marginal to both clusters comprise a third category, which we will also model by a Poisson traffic with intensity γ_3. Three users classes thus evolve. The users in each one of the classes 1 and 2 respectively communicate via only the LSRAA1 versus LSRAA2 algorithms, while due to double exposure, the users in class 3 have a choice. For communication with each other and system connectivity, they can join either one of the LSRAA1 and LSRAA2 algorithmic systems. Such a choice may be depending on the two cluster feedbacks that the users in class 3 are simultaneously exposed to. The issue here is the design of dynamic algorithms that implement such a choice, for better delay performance for the users in all three classes.

We consider interconnection policies for the two-cluster model described above. One possibility could be the following: Upon generation of a new packet, each user in class 3 joins the LSRAA1 with probability p and remains there until his packet is successfully transmitted; with probability, 1-p he joins the LSRAA2 instead. The probability p is then chosen so that the delays across the three classes (classes 1, 2, and 3) are equalized. The above policy is relatively simple, and compatible with all the existing LSRAAS. The disadvantage is that the marginal users must know the a priori assigned probabilities at all times. In the presence of the dynamically changing topologies considered here, those probabilities should change dynamically as well, and their values should be constantly known to the users. But this implies knowledge of the system topological dynamics at all times, which is either very hard to obtain or requires a large increase in system feedback information, and thus in bandwidth of the system feedback channels.

For the mobile environments we consider here, dynamic schemes are more appropriate for the marginal users in the system. Such schemes only require that the marginal users know the operations and characteristics of the LSRAAs they can join, no increase in feedback information is thus necessary then. In this paper, we develop such a scheme for the two-cluster model, when the local LSRAAs in each cluster are as those in [1]. We then determine the system stability region and we simulate the delays for each of the three user classes.

II. SYSTEM MODEL

We consider a two-cluster packet-radio system. We assume that in each of the two clusters, some synchronous

This work was supported jointly by the ONR contract N00014 86-K-0742 and the AFOSR contract F49620-87-C-0107

LSRAA is deployed. In particular: (1) Time is divided in slots of length equal to the duration of a packet, and the starting instants of the slots are identical in both clusters. (2) In each cluster, the clusterhead broadcasts a feedback per slot, which corresponds to the outcomes induced by the local LSRAA. This feedback is binary, Collision (C) versus Non-Collision (NC). (3) In each cluster, each local user is required to monitor the feedback from the local clusterhead continuously, from the time he generates a new packet to the time that this packet is successfully transmitted. We assume zero propagation delay and, that both the forward and the feedback channels are errorfree.

Each marginal user receives the feedbacks from both the local clusters correctly and without propagation delays. At the time when a marginal user generates a new packet, he starts monitoring the feedbacks from both clusters continuously, until he decides to join the operations of one of the two LSRAAs, for the transmission of his packet. Upon this decision, he maintains the continuous monitoring of only those feedbacks that correspond to the LSRAA he chose, until his packet is successfully transmitted.

It is assumed that the mobility of the users in the system is low enough, so that each user remains within the same geographical region (local or marginal) from the time he generates a packet to the time that this packet is successfully transmitted. If the local LSRAAs in each cluster have good delay characteristics, then with high probability this time period is relatively small.

It is assumed that the local traffic generated in cluster i, i=1, 2, is a Poisson process with intensity λ_i, i=1,2, and that the traffic generated by the marginal users is another Poisson process with intensity λ_3. As found in [2], for a large class of RAAs, the stability region of an algorithm as the population size increases is determined by its throughput under the Poisson user model.

III. THE ALGORITHMS

We assume that the two LSRAAs in the system are identical. Each LSRAA is the window algorithm in [1], which achieves throughput 0.43 and operates with binary C versus NC feedback. This algorithm has simple operational characteristics and is very insensitive to feedback channel errors.

Upon generation of a new packet, a marginal user imagines himself belonging to the systems of both the LSRAAs and follows their algorithmic steps, until the first time that he enters a collision resolution event in one of them. Then, he remains with the latter LSRAA system, until his packet is successfully transmitted.

Let time be measured in slot units, where slot t occupies the time interval [t, t+1). Let $x_t(j)$ denote the feedback that corresponds to slot t, for cluster j; j=1,2, where $x_t(j)$=C and $x_t(j)$=NC represent collision and noncollision slot t in cluster j, respectively. The local LSRAA in cluster j is implemented independently by each user in the system, and utilizes a window of length Δ. Let some local user in cluster j generate a new packet within the time interval $[t_1, t_1+1)$. Then, he immediately starts observing the feedback sequence $\{x_t(j)\}_{t\ge t_1}$, beginning with the feedback $x_{t_1}(j)$. Let us define the sequence $\{t_i(j)\}_{i\ge2}$, as follows: $t_2(j)$ is the first time after t_1, such that $x_{t_2(j)}(j)=x_{t_2(j)-1}=NC$. Then, as will be explained below, $t_2(j)$ corresponds to the ending slot of a Collision Resolution Interval (CRI) in cluster j, and from $t_2(j)+1$ on, the user can identify the ending slots of CRIs induced by the algorithm in cluster j. Each $t_i(j)$ corresponds to the ending slot of some CRI in cluster j, and $t_{i+1}(j)$ is the first after $t_i(j)$ such slot. At $t_i(j)$, the user updates his arrival instant, as follows: $t_1^{(i)}=t_1+(i-2)\Delta$; we call the sequence $\{t_1^{(i)}\}_{i\ge2}$, updates. Let $t_k(j)$ be such that $t_k(j)\in\{t_i(j)\}_{i\ge2}$, $t_1^{(i)}<t_i(j)-1-\Delta$; $\forall i\le k-1$, and $t_1^{(k)}>t_k(j)-1-\Delta$. Then, in slot $t_k(j)+1$, the user enters a CRI within the LSRAA of cluster j, and transmits his packet successfully during its process. He stops observing the feedback sequence $\{x_t(j)\}$ at the point when his packet is successfully transmitted. If the user is instead marginal, then he observes both feedback sequences $\{x_t(1)\}_{t\ge t_1}$ and $\{x_t(2)\}_{t\ge t_1}$, and follows the evolution of both the time sequences $\{t_i(1)\}_{i\ge2}$ and $\{t_i(2)\}_{i\ge2}$. If $t_k(1)<t_k(2)$, then in slot $t_k(1)+1$ he enters a CRI within the LSRAA of cluster 1, and transmits his packet successfully during its process. If $t_k(1)>t_k(2)$, instead, then he joins a CRI in cluster 2, in slot $t_k(2)+1$. If $t_k(1)=t_k(2)$, then he selects one of the local LSRAAs with probability 0.5. The above, describe the first entry rules, for the local and the marginal users, that is, how and when each newly generated packet first starts participating in some CRI. From the description of the first entry rule, it is clear that the marginal users have an advantage over the local users. In particular, their waiting time until they first enter some CRI is generally smaller than that of the local users; therefore, their total delays are generally smaller than those of the local users.

Consider the algorithm in cluster j, and let it start operating at time zero. Then, slot 1 is empty. In slot 2, the arrivals in [0,1) are transmitted, and a CRI begins. If the number of arrivals in [0,1) is less than two, then $x_2(j)$=NC, the CRI lasts one slot, and a new CRI begins with slot 3. If the number of arrivals in [0,1) is at least two, then $x_2(j)$=C, instead, and the CRI lasts as long as it takes to resolve the collision in slot 2. The end of the CRI can be identified by all the users in the system, (as will be seen below) Let T be a slot that corresponds to the end of some CRI. Then, in slot T+1, all the users with current updates in $(T-\Delta-1, T-1]$ transmit. If $x_{T+1}(j)$=NC, then the CRI which started with slot T+1 lasts one slot, and a new CRI starts with slot T+2. If $x_{T+1}(j)$=C, instead, then a collision occurs, whose resolution starts with slot T+2. No arrivals that did not participate in the collision at T+1 are transmitted, until the latter is resolved. During the collision resolution, each involved user acts independently, utilizing a counter whose value at time t is denoted r_t. The counter values can be either 1 or 2, and they are updated according to the rules below.

1. The user transmits in slot t, if and only if r_t=1. A packet is successfully transmitted in t, if and only if r_t=1 and x_t=NC.
2. The counter values transition in time as follows

(a) If x_{t-1}=NC and r_{t-1}=2, then r_t=1

(b) If x_{t-1}=C and r_{t-1}=2, then r_t=2

(c) If x_{t-1}=C and $r_{t-1}=1$, then

$$r_t=\begin{cases}1\text{, with probability 0.5}\\2\text{, with probability 0.5}\end{cases}$$

A CRI which starts with a collision, ends when it becomes known to all users that the initially collided packets have been successfully transmitted. From the operations exhibited above, it is not hard to see that such a CRI ends the first time (after its beginning) that two consecutive NC slots occur.

IV. ALGORITHMIC ANALYSIS

For convenience in notation, we will refer to the local users in cluster 1, the local users in cluster 2, and the marginal users, as subsystem 1, subsystem 2, and subsystem 3, respectively. In the algorithmic analysis, we adopt the Poisson user model (infinitely many independent identical users) for each of the three subsystems. In particular, we will assume that the three subsystem traffics are mutually independent, and that the user traffic in subsystem-j, j=1,2,3, is Poisson distributed with intensity λ_j.

We consider the evolution of the algorithms in the two-cluster system, and we assume that the system starts operating at time zero. Let us consider the sequence in time of the CRIs induced by the two LSRAAs in the system. Let the sequence $\{T_n\}_{n\geq 0}$ be such that. (1) For each n, T_n corresponds to the starting point of a slot which is the beginning of some CRI. We note that at T_n, two CRIs may simultaneously begin, one for each of the two LSRAAs in the system. (2) T_n is the first after T_{n-1} time instant which corresponds to the beginning of some CRI. (3) $T_0=2$, and at T_0 two CRIs begin; one for each of the two LSRAAs in the system.

Let $\{T_n^{(s)}\}_{n\geq 0}$ be the subsequence of sequence $\{T_n\}_{n\geq 0}$, which consists of those time instants when two CRIs begin simultaneously; one for each of the two LSRAAs in the system. Clearly, $T_0^{(s)}=T_0=2$. Let $D_{n,s}^{(j)}+1$, j=1,2,3, denote the total length of the unresolved arrival intervals in subsystem j, at the time instant $T_n^{(s)}$. $D_{n,s}^{(j)}$ is then called "the lag of subsystem j at time $T_n^{(s)}$." From the algorithmic operations in the system, we conclude, (1) $D_{n,s}^{(j)}\geq 1$ and the sets that consist of $D_{n,s}^{(j)}$ for j = 1, 2 are denumerable. (2) $D_{0,s}^{(j)}=1$, j=1,2,3. (3) At time $T_n^{(s)}$, the LSRAA in cluster k, k=1,2, examines two arrival intervals: one from subsystem k which has length $\min(D_{n,s}^{(k)}, \Delta)$ and contains arrivals generated by a Poisson process with intensity λ_k, and one from subsystem 3 which has length $\min(D_{n,s}^{(3)}, \Delta)$ and contains arrivals generated by a Poisson process with intensity 0.5 λ_3. (4) The triple ($D_{n,s}^{(j)}$, j=1,2,3) describes the state of the system at time $T_n^{(s)}$, and the sequence $\{S_n\}_{n\geq 0} \stackrel{\Delta}{=} \{D_{n,s}^{(j)}, j=1,2,3\}_{n\geq 0}$ is a three-dimensional irreducible and aperiodic Markov Chain.

We studied the ergodicity conditions of the Markov chain $\{S_n\}_{n\geq 0}$. The latter determine the stability region of the system. This region is plotted in Figure 1. To obtain results on the expected packet delay we simulated the system. In Figures 2 and 3, we plot expected packet delay, as a function of the Poisson intensity λ_3. From those figures, we observe the advantage of the marginal users, in terms of expected delays. Even when the expected delays of the local users increase without bound (saturation), the expected delays of the marginal users remain low, never exceeding ten slots, for all the examined cases. The delay advantage of the marginal users, as compared to the local users, increases monotonically, as the rate of their traffic increases.

V. COMMENTS AND CONCLUSIONS

In this paper, we studied a two-cluster interconnected system. Each cluster deploys the limited sensing random access algorithm in [1], and the interconnection is due to marginal users, who dynamically select one of the two algorithms for their transmissions. The interconnection policy adopted is dynamic, and requires no a priori knowledge of the traffic populations and characteristics, and of the states of the involved subsystems. It only requires knowledge of the algorithmic rules, and monitoring of feedbacks from the time a packet is generated to the time that it is successfully transmitted. In addition, the adopted interconnection policy presents a significant delay advantage to the marginal users. In all cases, it maintains the value of the expected delay for the marginal packet below ten slots. This delay advantage to the marginal users may be of high importance, when they transmit high priority data, and when dynamic cluster reconfigurations may result in temporary isolation of the marginal users if the transmission of their data is delayed.

REFERENCES

[1] M. Paterakis and P. Papantoni-Kazakos, "A Simple Window Random Access Algorithm with Advantageous Properties," Proceedings of INFOCOM'88 Also, submitted for journal publication.

[2] M. Paterakis, L. Georgiadis, and P. Papantoni-Kazakos, "On the Relation between the Finite and the Infinite Population Models for a Class of RAAs," IEEE Trans. Comm., vol. COM-35, pp. 1239-1240, Nov. 1987.

For the Figures see next page

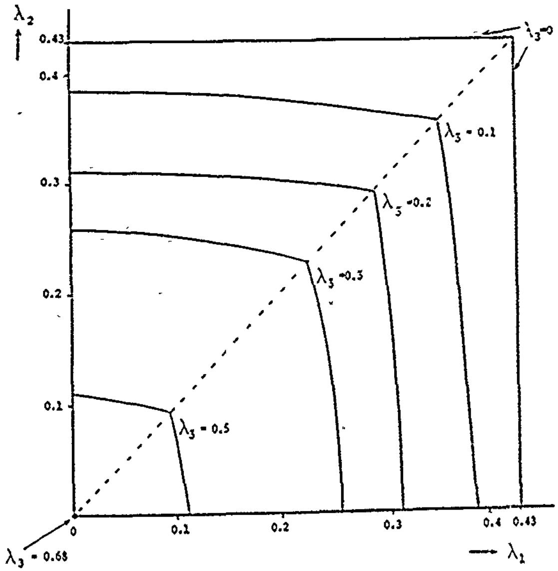


Figure 1
Boundaries of the (λ_1, λ_2) stable regions parametrized by λ_3

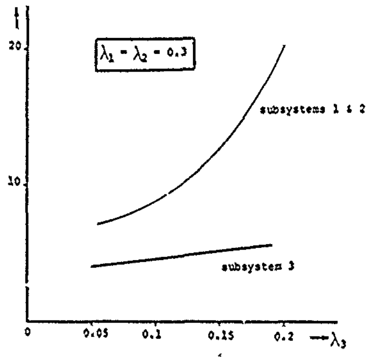


Figure 2
Expected Packet Delays

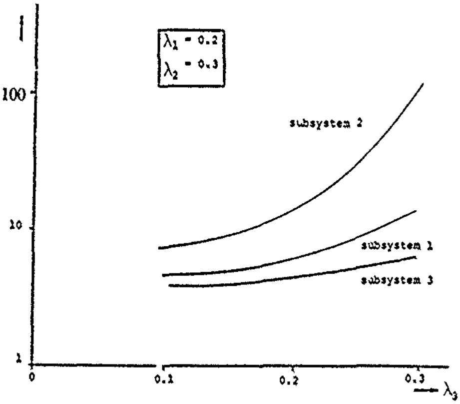


Figure 3
Expected Packet Delays

GATEWAY ROUTING PROTOCOLS FOR ENHANCED INTERNET SURVIVABILITY

BRADLEY J. HARNISH
ROME AIR DEVELOPMENT CENTER
RADC/DCLD
GRIFFISS AFB, NY 13441-5700

ABSTRACT

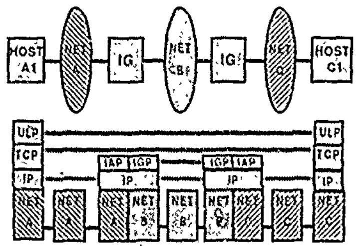
The probability that user datagrams will be delivered to specified destinations in a potentially hostile environment increases with the number of viable delivery paths between the source and destination hosts. Therefore, the number of usable paths in a common user communications system represents a measure of the survivability of user data in that system.

This paper presents a discussion of the enhancements to Internet survivability provided by gateways. The Internet architecture is described from a gateway perspective, gateway routing is described, and the primary features of two gateway routing protocols developed under Rome Air Development Center contracts are presented.

INTRODUCTION

The Department of Defense internet architecture is based on a reliable end-to-end protocol, a best-effort delivery protocol, and host level internet gateways to provide interoperability across diverse packet switched networks (see Figure 1). The Internet Protocol (IP), [MIL-STD-1777], provides a connectionless packet delivery service for the Transmission Control Protocol (TCP), [MIL-STD-1778], that operates between users on packet switched networks. IP defines the internet addressing scheme, header information, and routing mechanisms that allow packets to be treated as separate entities as they are transported across network boundaries from source to destination hosts. With this approach, internet resources are dedicated on a packet-by-packet basis instead of on a connection basis, and the term datagram is used, instead of packet, to emphasize the connectionless nature of the service. TCP manages the end-to-end connections between hosts with sequencing, acknowledgement, and re-transmission capabilities to ensure reliable data transactions.

Internet gateways (IGs) provide the actual interface between networks by implementing the physical, data link, and network layer protocols of each network they interconnect. The IGs peer with the attached network packet switches at the physical, data link, and network layers and with hosts and other IGs at the IP layer. A description of the protocol encapsulation between internet components can be found in [MIL-STD-1777].



ULP = Upper Level Protocols
TCP = Transmission Control Protocol
IP = Internet Protocol
NET = Network, Link, & Physical Protocols
IGP = Interior Gateway Protocol
IAP = Inter-AS Routing Protocol

FIGURE 1 -- PROTOCOL REFERENCE MODEL

Survivability of user data in a packet switching environment refers to the increased probability that data will reach its destination when communications resources are subjected to failure. The IP datagram approach supports internet survivability by dedicating internet resources on a packet-by-packet basis, so a datagram can be routed around failed components that it might encounter en-route to a particular destination.

Interoperability between diverse internet resources also supports internet survivability by increasing the number of potential paths that can be used to reach destination hosts. The IGs' ability to effectively discover and utilize this aggregate connectivity directly affects internet survivability. This ability is generally enhanced when the gateways collaborate their efforts to collectively determine the global connectivity information used to calculate internet routes.

Without specifying the exact physical configuration, it is difficult to precisely quantify the level of survivability that can be expected in an internet system. Many assumptions must be made about the topology of the internet resources, their administration in the operational environment, and the scenarios that would stress them. There are also many political, administrative, and legal issues involved in a strategy for internet survivability, since the resources are owned by a multitude of governments and their agencies, universities, and private industry. A discussion of some of these topics is given in [DAY].

This paper presents a discussion of the enhancements to internet survivability provided by IGs. In Section One, descriptions of an interoperability model that presents the internet from an IG perspective and a gateway protocol that enhances survivability within that framework are given. In Section Two, a description of the network partition problem is given, and a gateway protocol that solves the problem of delivering data into a partitioned strategic network is described.

I. INTEROPERABILITY AND ROUTING

INTERNET GATEWAY ROUTING

IP defines internet addresses for all hosts and IGs in the internet. Each IP address has two parts: a network number and a local address field. The network numbers are assigned such that each one uniquely specifies the network to which the host or IG is connected. The local address field is used to further specify the host or IG on that network. Therefore, from the IP perspective, IGs appear as hosts on each network to which they are connected. Each one is assigned a separate IP address for each network connection, so every IG has at least two IP addresses.

The routing mechanism specified by IP is a table look-up procedure based on the network number portion of the destination address listed in the IP header of each datagram. When an IG receives a datagram, it compares the network number to entries in the IP routing table. If the destination host is located on a network attached to the IG, then the matching IP routing table entry will point to the network interface module that can reach the destination host directly. Otherwise, it will point to the network interface module that can reach the next IG on the path to the destination host. If no match is found, a "destination unreachable" message will be sent to the source host or IG

The IP module then passes the datagram to the appropriate network interface module. There, the local address field of the IP address is mapped to the network layer protocol address and the datagram is forwarded into the network. The network takes over from there to deliver the datagram to the destination host or next IG.

An important IP option is the source route option. Hosts or IGs can specify the IGs that will be used for delivering datagrams to a destination. Those IGs are listed in the IP header in the order that they are to be traversed. When an IG receives a datagram that includes a source route, the IP address of the next IG listed in the source route option field is used instead of the destination address field.

The IP specification states a requirement for the IGs to support the dynamic adaptation of the IP routing table to reflect the current status of the internet topology. Generally speaking, the support processes consist of three main components [MCQUILLAN]

a) a measurement process for determining pertinent internet characteristics,

b) a protocol for distributing the information about these characteristics, and

c) a calculation to determine internet routes.

The first component enables each IG to determine the status of the internet topology in its local vicinity. Messages are exchanged between neighboring IGs to determine their current operational status and to measure the delay across the link (network) between them. The second component permits each IG to construct a global topology database by assimilating the information it periodically receives from other IGs. The third component operates on the global topology database to determine the best route, based on some metric (distance, delay, etc), to all other networks and IGs. The result of this process is the routing table used by the IP mechanism.

Collectively these processes make up the gateway routing protocol that is layered on top of the IP layer in the IGs. It generates IP datagrams to communicate with other IGs. Peer relationships at the gateway routing protocol layer exist only between IGs. See IGP in Figure 1.

AUTONOMOUS SYSTEMS

As an introduction to the concept of Autonomous Systems, it is useful to draw an analogy between a packet switch network and the internet. With the IGs being analogous to the packet switches and the networks serving as the backbone and subscriber access links, the analogy shows how the internet can be viewed as a gateway network. It illustrates the packet switch functionality required in the IGs for network control, datagram routing and the gateway routing protocol.

Due to the sheer size of the internet, which has grown considerably over the past few years, the single gateway network model is no longer valid. Although all of the IGs in the internet implement the specified IP routing mechanism and their gateway routing protocols are functionally similar, the implementations that have evolved over the past two decades are quite different. In addition, the diverse organizations that own and maintain the IGs have different service requirements, and the coordination required to manage the internet as a single gateway network amidst this diversity is no longer practical.

These factors led to the architectural creation of separate and independent gateway networks called Autonomous Systems (ASs). With this model, presented in [ROSEN], the internet is viewed as a collection of several ASs, or separate gateway networks, that are overlaid on the aggregate set of backbone and subscriber access links (networks) as shown in Figure 2. Provisions for the autonomus operation and management of each AS were proposed, the concept of Inter-AS routing was

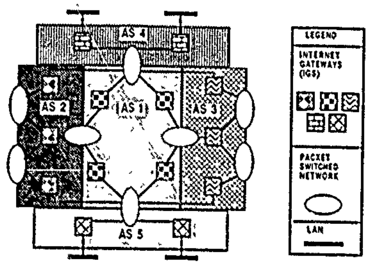


FIGURE 2 -- AUTONOMOUS SYSTEM CONCEPT

introduced, and a protocol design was presented. Each of these will be discussed briefly below. This approach relieved the problems mentioned above while maintaining the unified view of the internet for IP hosts and IGs by confining compatibility and management concerns to a minimal set of IGs.

An AS boundary is defined in terms of the administrative control of its nodes (IGs) and the common gateway routing protocol that binds them together. The IGs that belong to a particular AS subscribe to a common gateway routing protocol that runs only between member IGs.

The name Interior Gateway Protocol (IGP) (see Figure 1) was coined to refer to any gateway routing protocol that is implemented in an AS for internal use only. In terms of the three components discussed above, an IGP defines the measurement, distribution, and calculation algorithms used by all IGs in its AS. Therefore, IP routing decisions based on this data will involve only the member IGs and the links (networks) that interconnect them.

IGP boundaries have the effect of protecting ASs from each other while giving each the autonomy to optimize their IGP to meet their particular service requirements. They have the negative effect of limiting interoperability, and hence survivability, since not all hosts are reachable from a single AS. To counter this effect, and complete the AS model, Rosen introduced an Inter-AS routing protocol to route between ASs and allow hosts to maintain their flat IP view of the internet. The added complexity introduced by this model represents a necessary evil required for the continued operation of the rapidly expanding internet in the absence of a single omnipotent governing body.

INTER-AUTONOMOUS SYETEM ROUTING

Inter-AS Routing Protocol (IAP) is the generic name for gateway routing protocols designed for routing between ASs. It is the subject of several research efforts currently underway The Exterior Gateway Protocol (EGP) was introduced as part of the AS model and it is the current operational IAP. Those under development include EGP II and the Dissimilar Gateway Protocol which is described below.

IAP (see Figure 1) is layered on top of IP in the IG protocol suite next to IGP, since both use IP to deliver protocol messages to other IGs. However, interactions between IAP and IGP vary between the different designs currently under development. Only a subset (usually one or two) of the IGs in an AS participate in IAP They serve as the IAP spokesmen for the entire AS and are called IAP gateways to distinguish them from the other IGs.

All IAP designs involve some form of measurement and distribution processes, and some have a common route calculation as well In addition, IAP requires processes for peer acquisition and the negotiation and management of dialog parameters such as polling intervals and timeouts that will be used between IAP gateways. They are indicative of the flexibility required to maximize compatibility between the diverse IG implementations. A brief description of EGP is given below.

After its initial debut, the EGP design was refined in [SEAMONSON] and specified as a Defense Advanced Research Projects Agency (DARPA) standard in [MILLS]. The operational implementation relies heavily on a privileged AS, called the core AS, which is controlled by DARPA. The core AS is used as a transport medium for all other ASs and serves as the central distributor of global reachability information. The EGP gateway in each AS builds and maintains reports about the networks that are reachable by the IGs in its AS. These reports are sent to an EGP gateway in the core AS. The core IGP is used to combine these reports and build a global reachability list that is then distributed to the non-core ASs via EGP updates. The interpretation and distribution of these lists within the non-core ASs is left to the discretion of each AS administrator.

The EGP measurement and distribution mechanisms are such that the reachability lists contain loop-free paths through core IGs to all networks attached to the internet. However, no metrics are assigned to these paths to allow optimization of routing decisions. In addition, the number of networks in the internet has grown to the point of causing a severe burden on the core AS resources.

DISSIMILAR GATEWAY PROTOCOL

The Dissimilar Gateway Protocol (DGP) is currently being developed under a Rome Air Development Center contract with Ford Aerospace Corporation, Colorado Springs Division and their sub-contractor, M/A-COM Government Systems Division. The program will produce a formal DGP specification and a DGP implementation for the Multinet Gateway Advanced Development Model.

The DGP designers had the luxury of learning from the operational experience of EGP The fundamental difference between the two is that DGP includes capabilities for route calculations based on type-of-service specifications, route restrictions, and multiple paths to a destination network. Another major difference between the two is the flexibility designed into DGP that de-couples administratively defined hierarchical organization and AS peering restrictions from the protocol itself.

At least one IG in each AS will participate in DGP and it will serve as the DGP spokesman, or one of the DGP spokesmen, for the entire AS. These IGs are called DGP gateways to distinguish them from the other IGs. An additional designation for some of the DGP gateways is the name "border" gateways. Consider the example shown in Figure 3 where the DGP gateways are circled. The border gateways of AS1 are 2.2 and 3.0. Note that they are not members of AS1 but are directly connected to a DGP gateway in AS1 by networks Y and X respectively. Border gateways for AS2 are 1.1 and 4.0. IGs 1.0 and 2.2 are border gateways for AS3 and AS4 respectively.

DGP relies on the IGP of each AS to make internal measurements and to maintain a consistent IGP database at each of its IGs. Although it does not dictate the measurement or distribution mechanisms to be used, DGP does require that changes to the IGP database are reported when they occur and it specifies the format of the reports.

DGP is designed to operate on an arbitrary hierarchy of ASs. The relative level of each AS is left for an administrative authority to assign. This provides the flexibility needed to accommodate future growth while maintaining compatibility with existing implementations. DGP is also designed to operate on administratively assigned peering restrictions that specify which ASs may interact.

Together, the assigned hierarchical level and peering restrictions dictate the hierarchical organization of the DGP database in each DGP gateway. The content of a DGP gateway's database equals its IGP database plus the pertinent information contained in updates it receives from DGP peers. Updates include a list of IGs in the peer's AS, the networks attached to those IGs, and a metric associated with each network interface (provided by the IGP of that AS). For a given DGP dialog, the lower level peer reports its entire database to the upper peer. The upper level peer, in turn, reports to the lower peer the border gateways of the AS(s) reported by the lower level peer and the networks attached to them. Therefore, DGP gateways at the highest level in the hierarchy have complete routing information, and hence a greater resource burden, than those at lower levels.

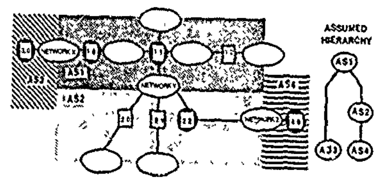


FIGURE 3 -- DGP TOPOLOGY & HIERARCHY

Using the example DGP topology and the assumed hierarchy shown in Figure 3, the following peer relationships exist. Between AS4 and AS2, 2.2 is the upper and 4.0 is the lower. Between AS2 and AS1, 2.2 is the lower and 1.1 is the upper. Between AS3 and AS1, 3.0 is the lower and 1.0 is the upper peer. (Within AS1, 1.0 and 1.1 are equal peers and report their entire database to each other.) The database in 2.2 includes a list of all AS2 IGs and the networks attached to them, the database of 4.0, and the networks attached to 1.1. The database in 1.1 includes the entire database of 2.2 plus a list of all AS1 IGs and the networks attached to them, and the database of 3.0.

The DGP database and route restriction information in each DGP gateway is used to calculate inter-AS routes. The route calculation is a shortest path computation (optimized for maximum throughput, minimum delay, lowest number of hops between IGs, or minimum cost) that is performed on request, as described below.

When an IG receives a datagram with a destination address that is outside of its AS, it requests a route from DGP. If the IG is not a DGP gateway, it can use its IGP to send this request to a DGP gateway in its AS. The request specifies the source (requesting) IG, the destination address, type-of-service indication (max thruput, min delay, min hop, min cost), route restriction indication(s) (packet size, reliable delivery), and the number of alternate routes to be calculated.

DGP calculates the route based on its database contents. If the specified destination is known to the DGP gateway, the route consists of an ordered list of IGs in the path to be traversed from source to destination. Otherwise, the route supplied is an ordered list of the IGs in the path to the nearest upper level border gateway The requesting IG stores the route for use with the IP source route option for subsequent datagram forwarding to that destination.

II. PARTITION REPAIR BY GATEWAYS

Military packet switching applications have strong requirements for continued computer communications during crisis periods. These requirements led to the development of the connectionless datagram service and internet technologies described above, and they continue to drive survivable communications research today. In addition to the IAP efforts described above, many research efforts have focused on IGP designs to meet the service requirements of specific ASs. This section will address the network partition problem and an advanced IGP called the Survivable Internet Routing protocol.

THE NETWORK PARTITION PROBLEM

A network can become partitioned into two or more segments, where hosts in the same segment can still communicate but hosts in different segments can not. Partitions can be caused by the failure of nodes and/or certain links in the network or by severe congestion. In the case of mobile radio networks, link failures can result from nodes that move out of range of each other.

As discussed in [CERF], the IGs can play an important role in restoring communications between the network segments. Given the ability to discover and exploit the aggregate internet connectivity, the IGs can find alternate paths through other IGs and networks that can be used to transport datagrams to the segments of a partitioned network. Since an IG must be attached to a segment to provide an external path, the physical configuration of internet resources relative to the network partition(s) is obviously an important factor.

Figure 4 shows a sample internet configuration with partitions in networks B and C. Because of the location of the partition in network C, there is nothing that the IGs can do to restore service to host C1. From the gateway network perspective, the partition in network B represents a broken backbone link between IG1 and IG3 and a questionable access link to network B hosts.

In Figure 4, IG1 and IG3 would detect the network B partition when their IGP polling messages to each other were not answered. IG2 and IG4 could detect the partition in network B from IGP updates by noticing that IG1 and IG3 are reporting connectivity to network B but not to each other. These updates will indicate that network B is no longer a viable path between IG1 and IG3, and the backbone routes that are re-calculated as a result of these updates will reflect this topology change.

A more difficult problem arises when an IG receives traffic destined for a host on a partitioned network. As discussed above, the IGs normally base routing decisions on the network portion of IP addresses. The local portion of the IP address is used by the network interface module to determine the network layer protocol address of the destination host. The physical location of hosts is generally not conveyed by these addresses. Therefore, additional information and/or processing is required to determine which IG can reach the specified host. Once this is known, traffic can be forwarded to that IG for subsequent delivery to the host. This approach is consistent with the routing mechnaism specified by IP.

To restore communications to hosts on different segments of a partitioned network (e.g hosts B1 and B2 in Figure 4), the hosts could either play an active or passive role with the IGs in finding external paths through the

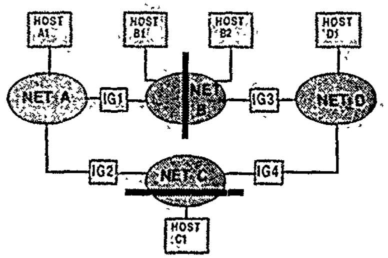


FIGURE 4 -- PARTITION REPAIR EXAMPLE

internet. In the active case, the hosts play an integral role in the route determination activities In the passive case, the hosts would simpl. re-direct undeliverable traffic to an IG, and the IG would find the external route.

In addition to issues regarding the division of responsibilities between hosts and IGs, the targeted AS environment is also an important consideration in IG based partition repair solutions. In a tactical environment, mobile radio networks can be both host access links and backbone links in a particular AS. Partitions can come and go as the nodes move about, and the association of cooperating resources, which may be both stationary and mobile, can be difficult to predict and control. Other problems such as link quality measurement, update duplication and aging, and contention resolution arise from the broadcast nature of packet radio transmissions.

In contrast, partition repair solutions targeted for a strategic environment have a different set of constraints. Here, communications are primarily transmitted over point-to-point links between stationary resources. Changes in topology during peacetime will occur primarily as a result of resource failure, scheduled maintenance, and occasionally by careless backhoe operators. After an attack, the surviving topology is likely to settle down and become relatively stable. The duration of the transient period will obvioulsy depend on the severity of damage and the IGs' ability to adapt to the changes.

Since strategic networks are generally more richly connected and diverse than tactical networks, compatibility issues associated with active host involvement are difficult to resolve. However, the relative stability of the strategic environment reduces the complexity of the problem and enables the delegation of partition repair responsibilities to the IGs.

RELATED WORK

In response to the discussions in [CERF], Radia Perlman presented potential solutions in [PERLMAN1] The proposed approach is based on the dynamic association of gateways with named segments of partitioned network(s). Source hosts actively participate with the IGs to locate, by trial and error, the segment where the destination host is attached. Once located, the source hosts insert a special IP address for the destination host, based on the derived network segment name, in the destination address field of the IP header. The IGs route the traffic based on this derived address as described in Section One.

To avoid the problems involved with active host involvement, Perlman presented a revision to the first proposal in [PERLMAN2]. The modification provided for source gateway determination of the network segment names and for optional host participation. If a source host does not participate actively however, the source gateway will send duplicate datagrams to each network segment.

Another approach was proposed in [SU] which eliminates the need to name the segments of partitioned networks. In this approach, the IP host addresses are expanded to include a name which associates the host to a specific IG. Thus, the IP addresses are changed from network centric names to gateway centric names and IG routing decisions are based on the IG portion of the IP address instead of the network portion. The approach involves active host involvement where the hosts are responsibile for associating themselves with an IG by using a "gateway affiliation request."

Only the IGs attached to a partitioned network (i.e. destination IGs) need to be concerned with host reachability information. Other IGs continue to operate as usual without regard to network partitions. When traffic for a host on the partitioned network is received by a destination IG, it will know if it can reach the host or not. If it can, it will forward the datagram to the host directly. If it can't, it will encapsulate the datagram in a new IP header and forward it to a destination IG that can reach the host. That IG will strip off the extra header and use the original header information to deliver the datagram into the network segment to the destination host.

SURVIVABLE INTERNET ROUTING

The Survivable Internet Routing (SIR) protocol is an IGP that was designed by Ford Aerospace Corp, Colorado Springs Division under an RADC contract that was completed in November 1986. A detailed description of the issues and rationale which led to the chosen SIR design can be found in [MCDOWELL]. A follow-on RADC contract was awarded to Ford in September 1987 to implement the SIR algorithms for the Multinet Gateway Advanced Development Model (MNG ADM). A description of the primary features of the SIR protocol design is given below.

From its inception, the SIR design was targeted for the strategic environment. Initial contract studies focused on ways to incorporate the desirable features of previous work into an IG based solution with a minimum impact on existing protocol standards and with passive host involvement. Considerable attention was centered on the trade-offs between the distribution and maintenance of host reachability information and the amount of backtracking required when traffic is re-routed to appropriate destination IGs.

The SIR development was primarily dedicated to the problem of delivering traffic to hosts on a partitioned network. The end products of the initial SIR contract include a generic partition repair solution and a complete IGP specification for the MNG ADM. The normal IGP processes, as described in Section One, were developed under the Multinet Gateway contract (awarded to Ford in 1981) and incorporated into the SIR configuration.

The SIR IGP measurements and distribution protocol provide a complete view of the AS topology at each MNG node. Updates are distributed to all other MNG nodes using a reliable flooding mechanism on a periodic basis and whenever a change is detected. The route calculation is based on the minimum distance to the MNGs attached to the destination network and the next MNG in the path. Unique identifiers are assigned to each MNG by the SIR IGP to avoid the confusion caused by each having multiple IP addresses. This gateway centric naming scheme, used only for gateway-to-gateway traffic in an MNG AS, simplifies partition detection, update management, and source routing mechanisms.

Partitions are detected in all MNGs by local IGP measurements and from IGP updates as described above. Once detected, the partitioned network is noted in the IP routing table to signify that traffic destined for that network must invoke partition repair processes. With this approach, this special processing is executed only as needed without interfering with normal IGP or IP datagram forwarding processes.

The SIR IGP processing that provides a generic partition repair solution is described below without further reference to the MNG. It is assumed that each IG knows of all network partitions and all IGs attached to the partitioned network(s). The term "source IG" will be used to refer to the first IG to receive an IP datagram destined for a partitioned network, and the term "destination IG" will refer to an IG attached to a partitioned network.

Upon receipt of a datagram destined for a host on a network which is known to be partitioned but a route to it is unknown, the source IG will initiate special processes to determine which destination IG can reach the specified host. These processes will build an IGP Probe Request message for each destination IG on the network in question The destination hosts and the source IG are specified in the IGP Probe Request.

Upon receipt of an IGP Probe Request, each destination IG sends a special IP message, called an Echo Request, to the host listed in the IGP Probe Request. Obviously, the host will only receive the ones sent from IGs attached to its segment of the partitioned network. The proper response to an IP Echo Request is an IP Echo Response. Echo Request/Response messages are short IP datagrams that hosts are required to process/generate when they connect to the internet. Upon receipt of an IP Echo Response from the specified host, the destination IG(s) send(s) an IGP Probe Response to the specified source IG to indicate that it can reach the specified host.

Upon receipt of an IGP Probe Response, the source IG extracts the destination IG address or name and enters it into the host reachability table it has created for the partitioned network. If multiple IGP Probe Responses are received in response to a given IGP Probe Request, only the first one to arrive is processed. This reflects the assumption that the first one to arrive was sent on the best path between the source IG and the destination host.

The destination IGs listed in the host reachability table are used in conjunction with the IP source route option when subsequent traffic destined for the listed hosts is received. When an entry does not exist, the probe sequence just described must be repeated. To avoid excessive host reachability tables in each IG, the entries can be limited to a certain number and aged accordingly. Care must be taken when implementing this scheme to achieve a good balance between the number of hosts for which paths are known and the frequency of which the probing sequence must be repeated. Some guidlines are given in [MCDOWELL].

Partition repair processing remains in effect until the IGP updates indicate that the partition no longer exists. When that happens, the partitioned network indication is removed from the IP routing table for the network in question so that the special processing will not be invoked.

Hosts and/or switches that can recognize that their network has become partitioned can re-direct intra-network traffic to an IG for external delivery via the internet. The IGs perform the processing described above to find a path to the destination host.

SUMMARY

Inter-AS Routing Protocols (IAPs) restore the interoperability that was lost with the creation of Autonomous Systems (ASs) and internet survivability is restored as a result. The Dissimilar Gateway Protocol enhances the survivability offered by the Exterior Gateway Protocol by providing near optimal routing capabilities and distributing the IAP burden according to operational need and Internet Gateway capabilities. These features will lead to a more dynamic and distributed control of internet traffic that should improve IAP robustness and operational efficiency.

Internet Gateways can restore service to hosts on a partitioned network by finding internet paths between surviving segments. The Survivable Internet Routing protocol is a source gateway approach to the partition repair problem for the relatively stable strategic environment. Internet Gateways collectively determine internet routes to segments of partitioned networks. Host involvement in this process is limited to echo request/response processing required by the Internet Protocol error service specification.

REFERENCES:

NOTE. The Internet Engineering Notes (IENs) and Request For Comments (RFCs) cited below are available from the DDN Network Information Center, SRI International, Room EJ291, 333 Ravenswood Avenue, Menlo Park, CA 94025 and also on-line from the ARPANET host SRI NIC ARPA.

[CERF] Cerf, Vinton, "Internet Naming and Addressing in a Tactical Environment," DARPA Internet Working Group Report IEN-110, Information Processing Techniques Office, Defense Advanced Projects Agency, August 1979.

[DAY] Day, M. H., Lockhart, C. M. "Survivable Network Planning at AT&T Bell Laboratories," IEEE MILCOM 86, AT&T Bell Laboratories, Oct 1986.

[MCDOWELL] McDowell, William E., Clark, Patricia A., Heckard, Max A., Rose, Kevin H., "Survivable Internet Routing Protocol Program - Final Technical Report," RADC TR 86-288, Rome Air Development Center, February 1987

[MCQUILLAN] McQuillan, L. M., Richer, Ira, and Rosen, Eric C., "The New Routing Algorithm for the ARPANET," IEEE Transactions on Communications, Vol. Com. 28, No. 5, May 1980.

[MILLS] Mills, D. L., "Exterior Gateway Protocol Format Specification," DARPA Internet Working Group Report RFC 904, M/A COM Linkabit, April 1984.

[PERLMAN1] Perlman, Radia, "Internet Routing and the Network Partition Problem," DARPA Internet Working Group Report IEN-120, Bolt, Beranek, and Newman Inc, October 1979.

[PERLMAN2] Perlman, Radia, "Flying Packet Radios and the Network Partition Problem," DARPA Internet Working Group Report IEN-146, Bolt Beranek, and Newman Inc, October 1982.

[ROSEN] Rosen, Eric C., "Exterior Gateway Protocol (EGP)." DARPA Internet Working Group Report RFC 827, Bolt, Beranek, and Newman Inc, October 1982.

[SEAMONSON] Seamonson, Linda J., Rosen, Eric C., "STUB" Exterior Gateway Protocol." DARPA Internet Working Group Report RFC 888, BBN Communications, January 1984.

[SU] Su, Zaw-Sing, Mathis, James E., "Internetwork Accommodation of Network Dynamics:Reconstitution Protocol," Interim Technical Report, SRI Project 5453, SRI International, February 1986.

A COMMUNICATION NETWORK WITH SOME HIGH PRIORITY MESSAGES

I. Stavrakakis, D. Kazakos
Department of Electrical Engineering
University of Virginia
Charlottesville, VA 22901

Abstract

In this paper we consider a communication system with a large number of geographically separated users. An active user (i.e. one who needs to make use of the common communications resource) can be in one of two possible states (H, L), depending on his nature and / or the type of information that he desires to communicate. Users who are in state H are given some priority over those in state L. Users which are in state H by nature can be commanders in a military environment. Users which are in state H due to the type of information that they possess can be any user who has a critical information that deserves fast transmission to a central decision maker. The traffic generated by the users in state H is assumed to be small compared to the total traffic that the system can accommodate.

For the above system we develop a binary feedback (collision / non-collision) random-access protocol which serves all the users. Throughput analysis is performed and the stability region of the system is obtained. Mean packet delay results are also analytically obtained for the cases in which the traffic generated by the users in state H is less than 30% of the total traffic that the system can accommodate. The delay results show that the protocol induces much shorter delay for the high priority messages.

I. Introduction

A lot of work has been directed towards the development of the multi user random access communication systems with a homogeneous population of users [1]-[5]. There are many practical applications, however, in which some or all users can alternate between to possible states H and L. Packets which are generated by users in state H should be given some priority over those generated by users in state L. Users who are in the same state are considered to be in the same class. As a result two classes of users are created (H, L) and the user population is generally non-homogeneous.

In a military environment, permanent members of class H can be commanders, while any other user of the system who has critical information can move from class L to class H temporarily and return to his original class after the critical information has been transmitted successfully.

In a mobile user environment where users move in and out of the range of the system, or move from region to region, fast moving users may need to experience shorter delays than the regular ones; this may be necessary to make packet transmission possible while the user is still inside the region. Also, users that are close to the boundaries of a region and are going to move outside it, should experience shorter delays. These users can be members of class H.

In a non-military static user environment members of class H can be users who pay more or users who carry control information which is critical for the operation of a system. have high priority and should reach their destination faster than the regular ones. High priority packets can be those which are generated by high priority users (e.g. important users, or users that can pay more for better service), or can be packets that are generated by any user of the system but the information that is carried is characterized as important and deserves high priority in its transmission.

In the next section we describe a communication system with two classes of messages (or users) in detail. In the same section we also describe the proposed random access protocol which determines the common channel alocation. In section III throughput and delay analysis are briefly described, while in the last section the results of the analysis are shown and conclusions are drawn.

II. The Random Access Protocol

We consider a large population of geographically separated users who use a single communication channel. Users which at certain time instant are in state H have some priority over the rest of the population and they form the high priority class H. It is assumed that the packet traffic generated by that class represents only a small percentage of the total traffic that is served by the system. In other words, it is assumed that the packets that need special service are rare and this is a realistic assumption at least for the environments which were described above.

Research supported by the Air Force Office of Scientific Research through Grant AFOSR 87-0095.

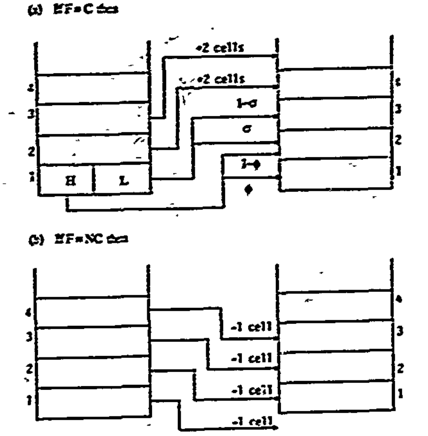


Figure 1.
Operation of the algorithm via the imaginary stack;
H (L) are the high (low) priority users.

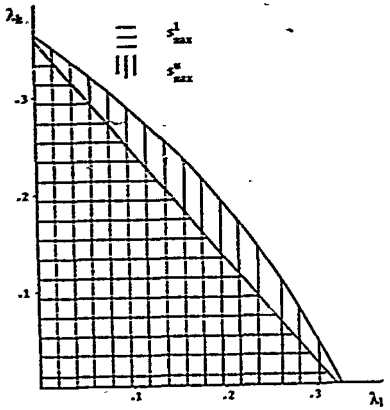


Figure 2.
Upper and lower bounds on the maximum stable throughput; λ_h and λ_l are in packets per packet length.

session length of multiplicity (mu , nu), by following procedures similar to those that can be found in [4], [7], [8]. The set of pairs (λ_h,λ_l) for which such a bound was possible to obtain, is a lower bound on the stability region of the algorithm. An upper bound can be obtained by solving a truncated version of an infinite dimensionality linear system of equations with respect to $L_{\mu,\nu}$, [14]. The latter system is obtained by considering the expectations of the recursive equations which describe the operation of the system. The stability region of the algorithm is plotted and it is shown in Fig. 2.

The mean delay of the high and low priority packets is also calculated but only for input traffic pairs (λ_f,λ_s) such that $\lambda_f \leq .065$ packets per packet length. For that region, bounds on the involved quantities was possible to obtain. This range of pairs determines the operation region of the algorithm; i.e.

$$S_{op}=\left\{(\lambda_h,\lambda_l):0\leq\lambda_h\leq0.065,\ 0\leq\lambda_l\leq\lambda_{l,max}(\lambda_l)\right\}$$

where $\lambda_{l,max}(\lambda_h)$ can be obtained from Fig. 2. The delay analysis is performed by applying the regeneration theory procedures that appear in [12], [4], [9], [10], or by using directly the strong law of large numbers, [11], [7]. Very tight upper and lower bounds on the mean delay of the high and low priority packets, D_h and D_l, respectively, were calculated for some values of the input traffic; the results appear in Table 1.

V. Results and Conclusions

The protocol that we developed and analyzed is appropriate for an environment where users can be in one out of two possible states. Two different classes of users are created to acommodate users in different states. Thus the user population is non-homogeneous and some user are given priority to transmit their packet. An algorithm for a homogeneous user population that consists of users in state H only and use binary feedback information and simple splitting after a collision, has been found to achieve a maximum stable throughput of ~ .36 [13] The algorithm that we suggest for the non-homogeneous population achieves total throughput, at least, between 320 - .357 depending on the contribution of the two classes to the total input traffic.

In Fig. 3, Fig. 4 and Fig. 5, plots of the bounds on D_h and D_l versus λ_l, for λ_h=0.01, λ_h=0.03 and λ_h=0.065 respectively, are shown. These values of λ_h correspond to an input traffic coming from the high priority class equal to ~ 3%, ~ 10% and ~ 20% of the total traffic that can be served by the system. From the plots it can be observed that the high priority packets (coming from users in state H) experience shorter delays than the packets of class L the difference is essential for $\lambda_l > .5\lambda_{l,max}$ If the nominal

The input traffic to the channel that is generated by each class of users is assumed to be Poisson distributed with intensities λ_h and λ_l respectively; the Poisson model is proved to be an appropriate model for the cumulative traffic that is generated by a large population of bursty users, which is assumed to be the case in the system under consideration. Messages are assumed to be packetized and of fixed length; it is assumed that time axis is slotted and that the beginning of a packet transmission coincides with the beginning of a slot.

All users may access the channel as long as they have a packet to transmit; the first transmission attempt takes place at the beginning of the first time slot that follows the packet generation instant. Because of the freedom that the users enjoy in accessing the channel, a transmission attempt results in either a successful packet transmission, or in a packet collision if more than one packet transmissions were attempted in the same time slot. Thus it becomes obvious that an algorithm is necessary in order for the conflicts to be resolved and the channel to remain usable.

It is assumed that all users that have a packet to transmit (and only these users need to do that) keep sensing the channel and are capable of detecting a packet collision; that is, we assume that a binary feedback information is available to all active users before the end of the current slot, revealing whether the slot was involved in a packet collision (C) or not (NC). Channel errors are not taken into consideration and packet collision is the only event that results in an unsuccessful transmission.

The first time transmission policy is kept the same for both classes of users; it is simple and implies that a packet is transmitted at the beginning of the first slot following the packet generation instant. It is apparent that if the two classes are to experience different delays, they should follow different steps in the collision resolution procedure. We develop a simple limited sensing collision resolution algorithm. The limited sensing characteristic is apparently important for a mobile user environment since the users may not be able to know the history of the channel before their packet generation instant. We assume that the state of a user is determined by the content of a counter that is assigned to each one of them; this counter is updated according to the steps of the algorithm and the feedback from the channel. Users whose counter content at the beginning of a time slot is equal to one, transmit in that slot.

Let $c_i^h(c_i^l)$ denote the counter content of a high priority (regular) user, at the beginning of the i^{th} time slot. Let also F_i, $F_i \varepsilon (C,NC)$, denote the channel feedback information just before the end of the i^{th} time slot. The steps of the collision resolution algorithm consist of the following counter updating procedures that take place at the end of each time slot.

(A) If $F_i = C$ then

$$c_i^h = 1 \begin{cases} c_{i+1}^h=1 & \text{with probability } \phi \\ c_{i+1}^h=2 & \text{with probability } 1-\phi \end{cases}$$

$$c_i^l = 1 \begin{cases} c_{i+1}^l=2 & \text{with probability } \sigma \\ c_{i+1}^l=3 & \text{with probability } 1-\sigma \end{cases}$$

$$c_i^j = r \rightarrow c_{i+1}^j = r+2\,,\ r\geq 2\,,\ j\varepsilon(l,h)$$

(B) If $F_i = NC$ then

$$c_i^j = r \rightarrow c_{i+1}^j = r-1\,,\ r\geq 1\,,\ j\varepsilon(l,h)$$

The first time transmission policy can also be described by using the concept of the counter; it simply implies that a new user sets the counter equal to one at the end of the slot in which its packet arrival took place. It did not seem to us reasonable to develop different first time transmission policies for the two classes of users. It would probably be a waste of the channel capacity to give priority to rarely appearing high priority packets, before it becomes known that a collision took place. If a conflict occurs, then the collision resolution algorithm offers some priority to the high priority packets that were involved in the conflict.

From the description of the algorithm it can be easily observed that the system is of continuous entry, i.e. new users enter the system at the beginning of the first slot that follows their packet arrival, unlike what happens in the blocked access algorithms [2]. The limited sensing characteristic of the algorithm, together with the lack of need for a central controller to coordinate the users, increase the robustness and applicability of the system.

The operation of the algorithm can also be described via the concept of the imaginary stack. Users whose counter content equals n are located in the n^{th} cell of the stack. Depending on the channel feedback and their location in the stack, the users move up and down as it is shown in Fig. 1.

IV. Performance Analysis of the Protocol

In this section we derive bounds on the stability region of the algorithm and the mean packet delay. Analysis is based on the concept of the session and the development of recursive equations to describe the operation of the system. A session is defined as a number of consecutive slots between properly selected renewal points of the system, [6]. If μ users in state H and ν users in state L attempted a packet transmission in the first slot of a session, then the pair (μ,ν) determines the multiplicity of that session. It can be easily concluded that the multiplicities of the sessions are independent identically distributed random variables.

If for an input traffic pair (λ_f, λ_s), the expected value of the session length of multiplicity (μ,ν) is finite, for μ and ν finite, then we say that the operation of the system is stable and the pair (λ_f, λ_s) belongs to the stability region of the system. The maximum overall sets of stable points (λ_h, λ_l) determines the maximum stable throughput region and is denoted by S_{max}.

Detailed stability and delay analysis of the proposed protocol can be found in [6]. For the stability analysis of the system we calculate a linear upper bound on the mean point of operation of the system is set around $\lambda_l = .9\lambda_{l,max}$, then the average high priority packet delay is less than half the one of the other class.

In table 1, the delay results of the suggested algorithm are compared with the delay, D*, that the homogeneous class equivalent algorithm (as described above), induces [13]. Again we can observe that always $D_h < D^*$ and particularly $D_h < .5D^*$ around the nominal point, the latter being defined as before.

Since privileged service is offered to some users, there has to be a price that the rest of the population must pay. The first consequence is the small reduction in the total throughput, as mentioned before. The other penalty is the increased average low priority packet delay compared with the one that the homogeneous population equivalent algorithm induces. From table 1 we can see that, indeed, $D_l > D^*$, as it was expected. The increase in D_l is far from catastrophic and it is realistic to consider that it is possible for a system to tolerate these delay increases for the low priority class, especially if strict limitations exist for the high priority users.

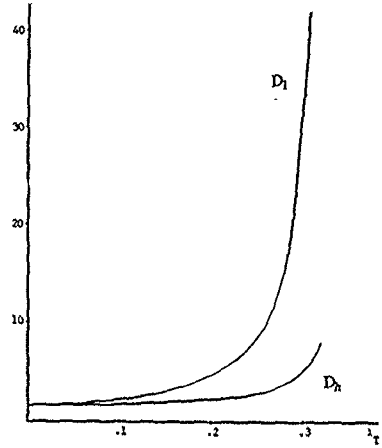


Figure 3.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for $\lambda_h = .01$ (packets/packet length).

λ_h	λ_T	λ_l	D_h	D_l	D^*
.01	.02	.01	1.555	1.590	~1.57
	.11	.10	1.829	2.369	~2.10
	.18	.17	2.186	3.815	~2.90
	.26	.25	3.095	9.922	~6.20
	.31	.30	5.793	39.793	~16 00
	.32	.31	8.718	78.748	~23.00
.03	.04	.01	1.632	1.678	~1.66
	.13	.10	1.951	2.571	~2.21
	.20	.17	2.389	4.312	~3.33
	.28	.25	3.672	12.681	~8.33
	.31	.28	5.453	28.961	~16.00
	.32	.29	7.113	45.905	~23 00
.065	.075	.01	1.800	1.878	~1.82
	.165	.10	2.234	3 054	~2.70
	.235	.17	2.900	5.595	~4.33
	.315	.25	5.801	23.101	~18 00
	.325	.26	7.200	33 080	~26.00

Table I.
Delay results.

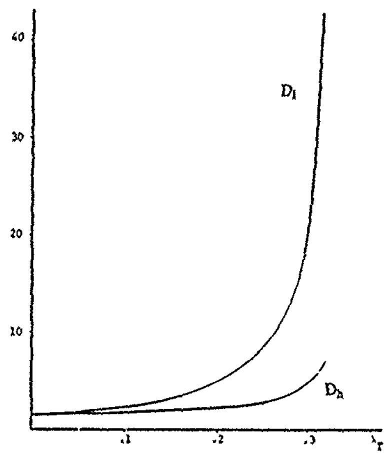


Figure 4.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for $\lambda_h = .03$ (packets/packet length).

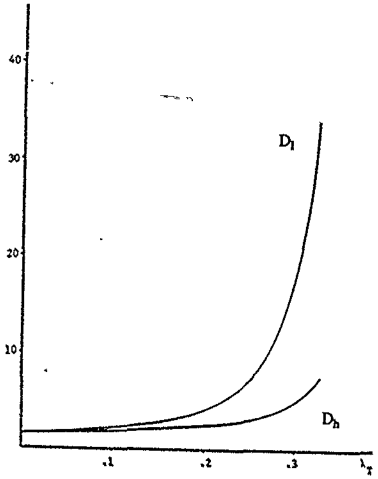


Figure 5.
Mean packet delay of the high, D_h and low D_l priority classes (in packet lengths) versus the total input traffic rate λ_T (packets/packet length), for λ_h=.065 (packets/packet length).

References

[1] B.S. Tsybakov, N.D. Vvedenskaya, "Random Multiple Access Stack Algorithm", translated from Problemy Peredachi Informatsii, Vol. 16, No. 3, pp. 80-94, July-September 1980.

[2] J.I. Capetanakis, "Tree Algorithm for Packet Broadcast Channel", IEEE Transactions on Information Theory, Vol. IT-25, No. 5, pp. 505-515, Sept. 1979.

[3] R.G. Gallager, "Conflict Resolution in Random Access Broadcast Networks", Proc. AFOSR Workshop on Commun. Theory and Applications, Provincetown, MA, pp. 74-76, Sept. 1978.

[4] L. Georgiadis, P. Papantoni-Kazakos, "Limited Feedback Sensing Algorithms for the Packet Broadcast Channel", IEEE Transactions on Information Theory, Special Issue on Random Access Communications, Vol. IT-31, No. 2, pp. 280-294, March 1985.

[5] N. Abramson, "The ALOHA System - Another Alternative for Computer Communications", Proc. AFIPS Fall Joint Computer Conference, Houston, Texas, Nov. 17-19, 1970, AFIPS Press, Montvale, N.J., pp. 281-285.

[6] J. Kurose, M. Schwartz, Y. Yemini, "Multiple Access Protocols and Time-Constrained Communication", ACM Computing Surveys, Vol. 16, No. 1, March 1984.

[7] I. Stavrakakis, D. Kazakos, "A Simple Stack Algorithm for a Code Division Multiple Access Communication System", Technical Report, No. UVA/525656/EE87/101, University of Virginia, October 1986.

[8] N.D. Vvedeuskaya, B.S. Tsybakov, "Random Multiple Access of Packets to a Channel with Errors", Problemi Peredachi Informatsii, Vol. 19, No. 2, pp. 52-68, April-June 1983.

[9] J.W. Cohen, "On Regenerative Processes in Queueing Theory", New York: Springer-Verlag, 1976.

[10] S. Stidham, Jr., "Regenerative Processes in the Theory of Queues, with Applications to the Alternating-Priority Queue", Adv. Appl. Prob., Vol. 4, pp. 542-577, 1972.

[11] G.L. Chung, "A course in probability theory", Academic Press, Inc. 1974.

[12] L. Georgiadis, L. Merakos, P. Papantoni-Kazakos, "Unified Method for Delay Analysis of Random Multiple Access Algorithms", Technical Report, UCT/DEECS/TR-85-8, University of Connecticut, Aug. 1985. Also submitted for publication.

[13] P. Mathys, "Analysis of Random Access Algorithms", Ph.D. dissertation, Diss. ETH No. 7713, Swiss Federal Institute of Technology (ETH), Zurich, 1984.

[14] L.V. Kantorovich, V.I. Krylov, "Approximate methods of higher analysis", pp. 21, Interscience Publishers, 1958.

"SAFENET" SURVIVABLE ADAPTABLE FIBER OPTIC EMBEDDED NETWORK

by CDR Marc M. Poland, USN
Space and Naval Warfare
Systems Command

and

LCDR Rex Buddenberg, USCG
Commandant (G-TES-1)

Summary

SAFENET provides naval warfare system designers with new tools and capabilities to achieve reliable, interconnection and distributed processing, along with the potential for significant cost savings.

SAFENET is the Navy's Local Area Network (LAN) Standard development effort, which is being conducted under the aegis of the Next Generation Computer Resources (NGCR) program. SAFENET provides real-time, militarized LAN standards based on the international standards organization/open systems interconnect (ISO/OSI) Reference Model, the process for creating these standards being a dynamic, interactive endeavor involving the Navy and industry. Both aspects are discussed in this paper.[1]

SAFENET I

SAFENET I is the first in this family of LAN standards. [2] It is an enhancement of the IEEE 802.5 LAN Standard and will be suitable for applications requiring high survivability in military platforms, such as Navy and Coast Guard ships.

As unique matters, the standard addresses achieving survivability at the physical layer: communications issues such as data latency and the need to achieve higher throughput (through the use of "light weight" protocols at layers 3 and 4); network management and network security. Upper layers issues, specifically at the Session and Application layers (layers 5 and 7), are also of concern but will not be discussed in this paper.[3]

SAFENET I is the result of an interactive and dynamic process of jointly developing military open system standards with industry. The goal is to achieve standards which are suitable for military applications but which originate from and are compatible with commercial, non-proprietary standards which offer the economies of commercial off-the-shelf equipment, i.e. low development costs for the Navy and industry provided support. The result, of the process, will be standards which can be returned to the public domain for use by anyone with similar military or commercial requirements.

SAFENET — The Product

SAFENET has its roots in the clear understanding by the Navy and Coast Guard that local area networks hold enormous promise as an additional design tool in providing reliable computer-to-computer and computer-peripheral data communications between weapon and warfare systems and subsystems.

The existing practice of wholesale point-to-point wiring is inefficient, expensive, inhibits systems growth, consumes precious space and adds significant weight to our platforms. The LAN offers a solution for reducing the N (N-1) discrete connection problem of this point-to-point architecture (without the use of costly and non-concurrent digital switches). The networks must meet our performance and reliability needs and be affordable.

The principal deficiency in available commercial LANs, from a military perspective, is survivability — the ability to continue to function in the face of electronic failure or battle damage. Another is the inability to provide deterministic delivery — the ability to guarantee delivery of a data packet within a given time frame. The existing ISO office automation protocols and the stochastic delivery characteristics of carrier sense LANs make them unacceptable for the stringent real-time data delivery requirements for weapon and warfare systems.

LOWER LAYER ISSUES (LAYERS 1-2)

The physical layer media decision was straightforward since the advantages of using fiber optics were obvious and there was never significant argument on the point. The key considerations were the maturity of the fiber optic components and their suitability for use in a military environment.

Linear and star configurations were rejected. Those which were based on contention media access protocols would not provide deterministic delivery. These topologies also lacked the required survivability features, in that, even if redundant, a break in both paths causes isolated partitioning of nodes in the network. Finally, these configurations do not easily accommodate growth.

On the other hand, of the various LANs, linear token passing, linear contention, and token passing rings, the rings provide the greatest throughput efficiencies. The IEEE 802.5 token passing ring standard was selected as the basis for SAFENET I. See Figure (1). By their nature, token rings deliver data deterministically and the IEEE 802.5 standard also provides eight levels of access priority. Further, simulation modeling has indicated that

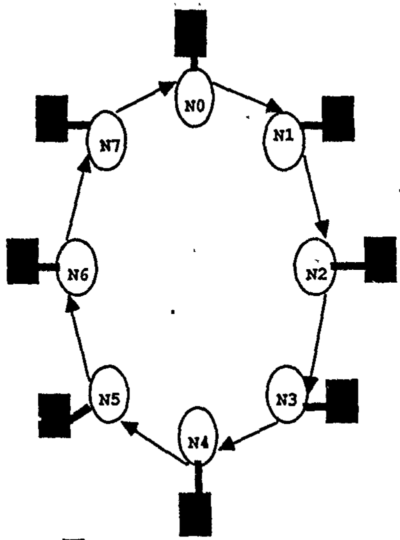


Fig. (1) Token Ring (IEEE-802.5)

token rings are indeed stable under saturated loading conditions.
What remained to be addressed was achieving a survivable architecture. The theoretical gains in survivability that result from a set of dual rings have been known for some time. Although some dual ring products were commercially available, there existed no commonly recognized standard that was multi-vendor supportable. To minimize development costs and achieve multi-vendor interoperability, SAFENET I adopted a dual

counter-rotating ring topology based on the IEEE 802.5 standard, shown in Figure (2), and assumed use of the existing Dual Counter-rotating Token Ring (IEEE-802.5 Enhanced) commercially available Texas Instruments TMS-380 adapter chipset, which implements the network Media Access Control (MAC) and Logical Link

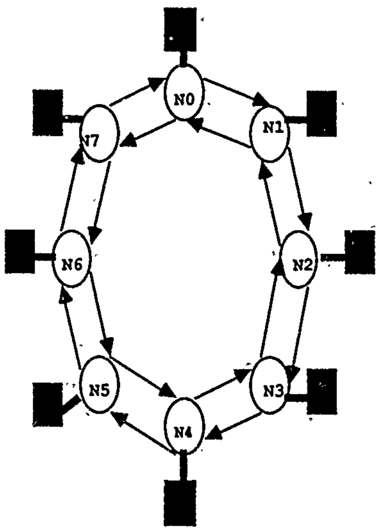
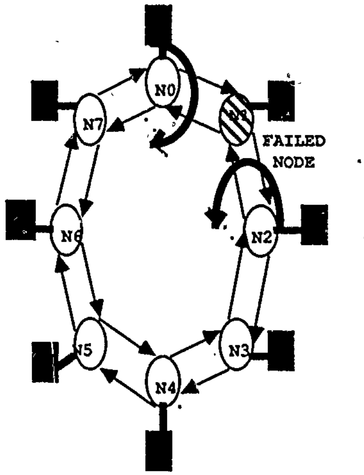


Fig. (2)
Dual Counter-rotating Token Ring
(IEEE-802.5 Enhanced)

Control (LLC) functions in silicon — no wheel re-invention allowed!

The first level of survivability consists of traffic shifting from the primary ring to the secondary in the event of damage to the primary ring. For shipboard applications, this is a feasible damage state since it is practical to physically alt-route the second ring in a ship by different cable ways so that severing one path will, with high probability, leave the second intact. The second level of survivability is obtained by the counter-rotating feature of the two rings. As indicated in Figure (3), loss of a section of the ring through destruction of node(s) or complete severing of a node — causes the last remaining node at each "end" of the network to wrap the incoming data back onto the outgoing



Failed Node Wrap
Fig. (3)
Dual Counter-rotating Token Ring
(IEEE-802.5 Enhanced)

data path,reconstituting the remaining intact portion of the network.
Additional damage to the ring will result in partitioning, but with continued operation of remaining intact sections. If additional survivability is required, several expansions on the theme are viable including a complete duplication of the dual ring network. Other alternatives include: back-up nodes or standby bridges

between segments of the ring which are considered most vulnerable, and damage control capabilities such as field splices and cable repair kits.

An additional feature incorporated into the lower layer of SAFENET I is the use of optical bypass switches at each node. If a node is inactive — without power — the switch passes the light signal straight through the node with only a small loss in signal power. Up to twelve nodes can be bypassed before signal regeneration is required. Aboard ship, that many nodes will normally be equipped with backup power to support critical applications or redundant nodes to ensure continued operation of the network.

At this writing, the physical and data link layer issues are well in hand.

Middle Layer Issues (Layers 3-4)

The SAFENET development effort has not been confined to just the lower layers of the reference model. We recognized early that to develop a complete SAFENET I standard, several other protocol issues must also be addressed.

"Lightweight" Protocols

The first of these is that of data latency, or expeditious delivery of vital packets of data in tactical situations. Although the full ISO stack will suffice for general purpose use and internetworking requirements, it was recognized that an additional stack of protocols would be required to achieve low latency/high throughput for data transfer between weapon and warfare systems. In final implementation, the additional stack would be co-resident with the standard ISO protocols.

This requires developing real-time Network and Transport layer protocols to meet our needs, since none exist. Hence, the name "lightweight" protocol.

One of the problems, predictably, was - what is real-time? That is, how rapidly must data be transferred through the SAFENET. Some applications studies indicated that tightest current needs within the Navy require a guaranteed 5 millisecond delivery time.

Armed with this assumption, we are investigating a number of alternatives. Committing middle layer functionality to silicon as opposed to software implementation, e.g., the express Transfer Protocol (XTP)/Protocol Engine, is one approach being investigated. Other approaches such as possibly splicing in portions of the General Motors Manufacturing Automation Protocol (MAP) — *into* the middle layers are also being reviewed. This portion of the SAFENET I standard is still under review at this writing.

LAN Management Issues

LAN management in the SAFENET I standard addresses three hierarchical aspects; station management, dual link management and network management. Key questions being resolved in the standard are; How many levels of management should there be? Where should they be?; What do they do and what is their effect on data transfer through the LAN? And, at what layer should the dual ring nature of SAFENET become transparent to the user?

Security

The third issue is that of a security. Military use requires that the networks be capable of carrying classified data. Complicating the problem, the same network should accommodate users with varying classification authorizations and needs. Most of the issues are identical to those of other military networks and the SAFENET efforts have been mostly to get in step with and supplement other research in this area. As those standards mature, they will be adopted. The goal is to establish security guidelines for implementing SAFENET I and subsequent SAFENETs.

The Output

In final product form, the SAFENET I standard is the first step in achieving the goals of interoperability and multi-vendor support through an open systems architecture and open systems interconnection in our computer resources. Additionally, we are proving that these networks can be realized without creating new and unique technology for the military application — the hallmark is adaptation of existing commercial standards and their support products.

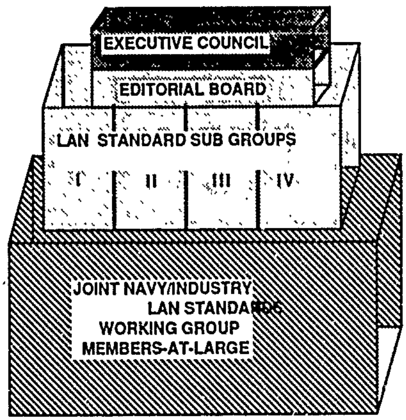
SAFENET — The Process

Describing SAFENET without discussing the process by which we are creating these standards would be telling only half the story. Through a dynamic, cooperative Navy (Coast Guard)/Industry effort, a significant leveraging of a relatively small taxpayer investment is being achieved, and a model for other similar processes has been developed. The military and industry can cooperate to the benefit of the user, taxpayer, and our national defense.

Similarly, we are finding that industry can also significantly leverage its investment by knowing exactly where we are going and fully participating in the process of getting there.

Open Forum — Open Standards

The SAFENET effort is under the direction of the Computer Systems and Engineering Division of the Navy's Warfare Systems Engineering Group at the Space and Naval Warfare Systems Command. The working group is organized into an executive council, editorial board, four subgroups that are writing the specific sections of the standard and general members-at-large.



The SAFENET Working Group (aka NGCR LAN Working Group) is only three years old, but has for about the past year been a booming operation. The group comprises members of the Navy and Coast Guard from various systems commands, laboratories and other activities along with a large membership of industry representatives.

The monthly meetings, which are completely open, are attracting well over a hundred participants, including representatives of over four dozen corporations. It is through this open relationship that we have been able to obtain the material necessary to maintain rapid progress in developing the SAFENET I standard. An interesting offspring of this effort is that through our embrace of industry standards for military use, we have seen representatives from the federal systems and commercial divisions of corporations meet each other for the first time at these meetings!

Also, by pooling the knowledge of all these experts, both government and industry, we have minimized the technical risk and maximized the potential for multi-vendor interoperability. Several industry participants have already established comprehensive research and development efforts to parallel the development of the standard. As a result, we are getting assurances that everything in the standard is buildable, because these vendors are indeed developing and demonstrating prototype products to the standard as it is being written.

Another spinoff of the open systems development is that our working group members are also active members of the IEEE, ISO, ANSI and other national and international standards organizations. They are providing a synergistic and effective mechanism to take the SAFENET standards development efforts back for recommended incorporation into the appropriate industry standards. For example, the lower layer SAFENET I reconfiguration scheme has been recommended to the IEEE 802.5 committee. Additionally, other SAFENET contributions have been made to the 802.5 Token Ring Conformance Test Book (IEEE 802.5D) Draft and the Fiber Optic Draft (IEEE 802.5J).

The goal of making the standard not just a military one, but a public one in the same spirit that TCP/IP has become public is achievable. Additionally, as we (the military) become part of the larger industry market we will gain the advantage of lower unit costs and rapid capture of new technology as it evolves. We will then be able to deliver combat capability at less cost to the taxpayer. A second advantage is that the standard is available to anyone who requires an equivalent capability and the military certainly does not have a corner on the need for highly survivable and fault tolerant networks.

Non-Development Item

One of the keystones of the SAFENET development effort is the commitment to deliver a product that can be installed on our ships soon and to minimize costs at the same time. This was the principal driver in the decision to start with the IEEE 802.5 and capture the existing TMS-380 adapter chipset in SAFENET I as the first SAFENET implementation.

Upward Compatibility

We are aware of the Fiber Distributed Data Interface (FDDI) development and that its chipset will soon be available. The decision has already been made that SAFENET II (100 MHz LAN) will be based on FDDI. In anticipation of this and also confident that the bandwidth demands will burgeon, the SAFENET I standard has carefully considered both growth path and upward compatibility requirements. Approximately 80% of

SAFENET I will directly transfer to SAFENET II. Upgrade to FDDI will require only, a change in the lower layer protocol. Also, by providing for a fiber optic cable plant that will not need to be changed, and by specifying a backplane standard for the LAN Interface Units, where needed, we have made provisions for ensuring easy upgrade from SAFENET I to SAFENET II. The upgrade is envisioned as a changeout of a single card set at each node.

A bit further in the future, we also recognize the need for fiber optic networks carrying data at gigabit rates. However, this aspect of network technology is still maturing. There are already some Navy efforts investigating high speed optical data transfer networks, analogous to a Metropolitan Area Network, which would serve as a platform-wide backbone to which SAFENETs would connect via gateway and we are watching them closely. We believe the Navy will be specifying, at least, some single mode fiber for use in the cable plant, in anticipation of networks operating at these speeds. In this aspect of SAFENET developments, we are attempting to look carefully through our optic fiber — at our crystal ball.

Platform or Ashore

The principal focus has been for platform use of SAFENET. However, there is nothing in the standard that would preclude it from satisfying ashore requirements also.

Conclusion

SAFENET I, is the first of a family of survivable, fiber optic LANs suitable for shipboard use. It represents a significant additional tool for the system designer in solving computer interconnect and data communications problems. But SAFENET and its parent, Next Generation Computer Resources program, represent something substantially more — a new way of solving military computer resource problems by adopting an open systems architecture approach and using open systems interconnection standards to achieve multi-vendor support and interoperability. This requires that we approach our network standards development by encompassing the full range of the ISO/OSI reference model. In the long run, we are working toward a complete migration of military networking to fiber optics.

Finally, and perhaps most importantly, SAFENET represents the realization that an open and cooperative effort between government, academia, and industry is achievable and welcomed by all parties. The result can be a more rapid transition of products from laboratory and commercial use into military applications to the benefit of all, especially the taxpayer.

About the authors:

LCDR Buddenberg is assigned to Commandant (G-TES-1), Shipboard Systems Division (large Cutters) and is a member of the SAFENET Working Group Executive Council.

CDR Poland is Deputy, Computer Systems and Engineering Division, Warfare Systems Engineering Group at the Space and Naval Warfare Systems Command, Washington, DC, and chairperson of the SAFENET Working Group-Executive Council.

1. The information used to prepare this paper was obtained by the authors through participation in developing the SAFENET I standard, presentations at SAFENET Working Group meetings and discussions with other Working Group members.

2. SAFENET I Draft Standard, Revision (1) of July 17, 1987.

3. DATA AND COMPUTER COMMUNICATIONS by William Stallings. The Open Systems Interconnection model for layering is:

Layer 1: Physical - concerned with transmission of unstructered bit stream over physical medium; deals with the mechanical, electrical, functional, and procedural characteristics to access the physical medium.

Layer 2: Data link - Provides for the reliable transfer of information across the physical link; sends blocks of data (frames) with the necessary synchronication, error control, and flow control.

Layer 3: Network - Provides upper layers with independence from the data transmission for establishing, maintaining, and terminating connections.

Layer 4: Transport - Provides reliable, transparent transfer of data between end points; provides end-to-end error recovery and flow control.

Layer 5: Session - Provides the control sturcture for communication between applications; establishes, manages, and terminates connections (sessions) between cooperating applications.

Layer 6: Presentation - Provide independence to the application processes from differences in data representation (syntax).

Layer 7: Appliication - Provides access to the OSI environment for users and also provides distributed information services.

INTERNETWORKING OR INTEGRATION OF DEFENSE NETWORKS ?

Philip M. Sykes

Rome Air Development Center, Griffiss AFB, Rome NY 13441-5700

1. ABSTRACT

Current Defense communications networks face a number of problems related to security, performance, resource allocation, survivability, management & control and operation & maintenance. In the majority of cases the technical solutions to these problems already exist but their implementation is impeded mainly by the large investment in existing systems, followed closely by the segregation of the long haul packet and circuit switched networks. This paper looks at internetworking and integration as two philosophies for improving Defense networks and how successful each is likely to be in addressing the needs of those networks. A number of weaknesses in internetworking are identified and a recommendation is made for an earlier than planned deployment of some key integrated network systems, particularly in the area of network access.

2. BACKGROUND

Existing military communications networks are fragmented, each requiring unique access procedures and with little or no possibility of direct communications between them. Users are dependent on a variety of networks and technologies to transfer information and the potential for sharing network resources and services is severely limited. Poor throughput for data or the inability . set up or complete a voice call are more likely to be caused by saturated network interfaces rather than congested switching or transmission systems. In this environment and particularly among data networks, internetworking has gained support as a means to provide connectivity and hopefully interoperability, between networks.

Data networking developed at a time when the only other pervasive communications resource, the switched telephone network, was almost entirely analogue and unsuited to data traffic. The telephone network was initially and is still used for access to packet switching nodes and for interswitch trunks. It is easy to understand that many people in the data networking community still consider that the switched telephone network has only a subservient role to play in data networking. As circuit switched networks are reaching a stage of providing complete end-to-end digital connectivity, the potential of such networks for handling data directly or as an access system to packet switched networks, needs to be reassessed.

With the advent of cheap computing power and high speed local area networks (LANs) much of the interactive traffic has been transferred from wide area data networks to the LANs. In contrast, file transfer traffic has increased on wide area networks causing one to question whether large file transfers could be handled more efficiently using a circuit switched connection.

Modern circuit switched networks are using out of band signaling for call setup and control and as a means of providing network management. Intelligent digital circuit switches now have the ability to communicate with each other using this signaling concept referred to as Common Channel Signaling. These circuit switched networks are acquiring significant intelligence with which a user can interact (via signaling) to enable provision and optimization of the required communication services.

Command and control functions are being decentralized to improve survivability. This implies a need for many new communications services to be present at each command and control site. The idea of having multiple independent but interconnected networks to provide these services at each site is unattractive. Integration of communication resources in this and other environments will have performance and cost advantages.

With these considerations in mind, the question must be asked: why not try to merge Defense voice and data networks into an integrated or unified network in an evolutionary manner and as quickly as possible ?

3. INTERNETWORKING

Internetworking is a term used to describe the interconnection, via gateways, of heterogeneous communication networks. The gateways terminate the internal protocols of each connected network and provide internetwork routing. The gateway function can be realized by additional software residing in a connected host or by a dedicated device.

3.1 SOLUTIONS PROVIDED BY INTERNETWORKING

a) Interoperability

Gateways provide connections between networks enabling application entities to communicate and interact. However, the interoperability of two end systems on different networks is a function of their connectivity and the compatibility of the applications running on those end systems. Internetworking does not provide interoperability but does facilitate it.

b) Survivability

Survivability is the ability of a network or network of networks to continue to function (in full or part) after a particular threat or damage scenario. Survivability is often stated as a major reason for wanting to interconnect networks. The survivability of the internetwork components (gateways and their links) and procedures, appear to receive little consideration.

Internetworking as a basis for survivability is highly questionable. Internetworking provides minimal survivability gains due to the sparseness of the interconnection and the fact that the gateway links are likely to share the same physical transmission paths as the networks they interconnect.

c) Performance

Performance across an internetwork is closely linked to the capacity of the interconnection systems (gateways and transmission links). Capacity of the internetwork connection needs to be considered in both the unstressed and stressed cases. In the unstressed case the interconnected networks are operating at their average utilization level. Gateways between these networks are serving the major function of providing connectivity to other resources (data bases, processing power, E-mail, etc.). In this environment it is unlikely that the path through the gateways is being used to avoid congestion on individual networks.

The collection of interconnected networks can experience increased levels of stress brought about by abnormally high user demands or trunk/switch failures resulting from malfunction or attacks on the networks. The resultant network partitioning will cause the internetwork "operation" to shift from a resource sharing mode to a survivability mode. In the survivability mode the remaining gateways and their links may be unable to handle the increased traffic load with throughput and delay becoming unacceptable.

Some networks may become transmission resources being used by adjacent networks to route around a network failure. The question here is whether the "transmission network" will allow the external traffic to enter through the gateways. In other words which traffic is more important, that from the adjacent partitioned network or the "transmission" network's own internal traffic? There is some hope of resolving this question if the networks use a common packet precedence/priority scheme. There is, however, a further complication when the networks are owned and operated by different entities e.g. Defense/private or US/NATO.

The instability that may result when a partitioned network effectively applies a high transient demand on another network is an area requiring further study. It is quite possible that with such transients, a network's management function may become so active that information throughput actually decreases.

Each transition through a gateway results in some delay due to packet processing: address translation, protocol conversion, security checking, etc. Gateways and their links are the choke points in a multi-network environment. When the networks come under stress, these choke points may degrade the overall internetwork effectiveness to the point where the gateways themselves appear as a partition. Packet voice with its low tolerance to transmission delay is unlikely to be supported in the near term through multiple networks and gateways.

d) Security

Networks are constructed using different security models which may enforce different security policies or operate in different modes (e.g. system high or multi-level). A security model specifies a number of network parameters including: access controls, accountability, information integrity, accreditation/certification and configuration management.

Internetworking does provide a means to logically and physically separate user communities that manipulate information having different security classifications while at the same time allowing the sharing of information of the same classification. The Multi-Level Secure (MLS) gateway concept currently being developed will allow controlled information transfers between networks having different security classifications.

e) Resource Allocation

The load profile of networks varies throughout the day so there are times when there is unused transmission bandwidth. A study [1] has shown that the load peaks of the Defense Data Network (DDN) and Defense Switched Network (DSN) do not correlate. The concept of dynamically sharing trunk capacity between the long haul data and voice networks offers the potential to ease current demand for new trunks and offers future savings. Internetworking offers no potential for dynamically sharing trunk capacity.

f) Network Management / Operation, Administration and Maintenance (NM/OA&M)

Monitoring and control functions are becoming far more complex as the numbers of connected networks grow. For example, DDN connected network growth is in the order of 50% per year. The gains that would be realized from interconnecting many independent networks will be determined by how well the collection of networks and their interconnection systems can be managed. The distributed management and control functions within the constituent networks and gateways must cooperate to form a unified management structure. In managing the information flow between networks, four issues are currently of concern. They are gateway routing, flow control, multi-level precedence and preemption (MLPP) and reconstitution and restoral. These issues have arisen because of the lack of a common network architecture.

Internetworking implies multiple, independent networks and therefore offers no potential to overcome the current duplication of OA&M effort.

4. NETWORK INTEGRATION

The long term objective of network integration is to have all networks converge toward a unified system architecture. "Unified" does not imply a single network technology but an architecture where user equipment, including traditional user devices (telephones, data equipment) and now including application specific LANs, cooperate and interact with a common wide area information network.

4.1 Conceptual Integrated Network

The wide area information network would be comprised of 4 components:

Users :- data equipment, telephones, sensors, local area networks.

Integrated access :- a common access point and interface specification for all users.

Integrated switching :- a single switch concept handling voice and data.

Integrated transmission :- a single trunking scheme handling the inter-switch traffic.

This integrated network will have to be achieved in an evolutionary manner which makes most use of existing assets while at the same time modifying current acquisition plans to more quickly achieve the end goal.

Integration will take place in three areas, namely access, switching and transmission. Integrated transmission occurs to some degree today by p--ching or multiplexing data and voice together on the same carrier system. Integrated switching can be achieved initially with co-located packet and circuit switches handling both data and voice users and hybrid switching architectures where a packet handler function is built into a digital circuit switch. A single switch fabric is the long term objective but as yet there is no clearly leading technology.

Integrated access is a reality today using the CCITT ISDN protocol architecture and offers the greatest potential for a major near term, DoD network integration initiative. ISDN offers the next evolutionary step beyond internetworking. ISDN can be viewed as a general purpose gateway between heterogeneous networks.

ISDN is characterized by:-

a) Full digital end-to-end connectivity.

b) Integrated access over a single access medium to a multitude of network services and capabilities including message, circuit and packet switching services.

c) Feature-rich out of band signaling capability between the user and the network and across the network. The signaling capability will allow users to access the intelligence residing in the network and therefore have a greater involvement in the management and control of the network.

4.2 SOLUTIONS PROVIDED BY INTEGRATION

Network integration will be driven by user demand. It is therefore essential to offer the benefits of integration to users at the earliest possible time. Integrated access systems should be implemented first, to give users the facility of integrated voice and data services even though the "network" beyond the access point may be comprised of independent circuit and packet networks. ISDN provides well defined integrated access systems that are currently being deployed in the commercial sector. The following solutions are based mainly on the benefits of integrated access through ISDN.

a) Interoperability

ISDN provides a new level of connectivity between networks. Any application specific user network or user terminal will be able to plug into any ISDN access point and receive full network services. ISDN contributes to improved interoperability through easier connectivity and through the internetwork dialogue available with Common Channel Signaling.

b) Survivability

Some of the greatest survivability gains for military communications can be achieved through interoperability with the public networks. All Regional Bell Operating Companies and the long distance carriers have embraced ISDN as their objective network architecture. Since the CCITT ISDN recommendations specify a common set of network services it will be possible to maintain these services (with a few exceptions) when a DoD user traverses public ISDNs.

The other interesting survivability consideration is that of a major communications intensive node that may need to be quickly relocated to another area. If this other area has ISDN access points or a mobile ISDN switch could be installed, it should be far quicker to provision the required services in the new location primarily due to the common access loop and connector, presented to all users.

c) Performance

Integrated access will greatly improve overall network performance. A significant advantage for data users is that an ISDN access systems provides a 64 Kbps dedicated or 16 Kbps multiplexed connection to the switching resources (circuit or packet) A 1.544Mbps access interface is also available. The need for modems and dial up connections will be eliminated. ISDN is designed to handle existing X.25 access protocols. New services based on the integrated nature of ISDN will also be available.

Portability of equipment and user control over the connection of terminal equipment will mean quicker service provisioning times.

d) Security

A unified network with a common set of interfaces and a consistent management and control structure provides a more stable, controllable environment in which to handle the issue of security.

Common channel signaling between switches and D-channel signaling between users and the network will be the means by which network access is controlled. A user access point will have a software defined security privilege. Common channel signaling offers the potential for multiple levels of security to exist within an ISDN switch and access point. The signaling network controls individual calls and has the necessary speed to enable user device and line privilege testing during the call set up process.

e) Resource Allocation

The ISDN interface allows dynamic allocation of access bandwidth. Via D-channel signaling a user may choose to configure the 64 Kbps access channel for either voice or data. Packet data access via the multiplexed 16 Kbps D-channel allows up to 8 terminals to be attached using a contention resolution access method.

Integrated switching (initially co-located circuit and packet switches) will enable users to choose the type of connection that best suits their instantaneous information transfer needs. For example, a circuit switched connection for a large file transfer or a packet switched connection for a data base query.

As a near term integrated transmission solution, an intelligent multiplexer [2] will be able to accept the output from co-located circuit and packet switches and dynamically "build" composite data/voice frames for transmission over a single bearer system.

f) Network Management / Operation, Administration and Maintenance (NM/OA&M)

The greatest duplication resulting from independent voice and data networks occurs on the access side of switches. With ISDN access lines handling both voice and data, wiring complexities and the time and cost of provisioning new services will be reduced realizing significant savings in OA&M. Network monitoring centers will be able to "look down" (using the signaling channel) at least to the ISDN access point to gather information on network status. Network management functions will be distributed in switches and perhaps to a small degree in a users' equipment. Common channel signaling and D-channel signaling will be used to exchange this information in addition to supporting network routing and reconstitution & restoral. Most importantly, users will be able to access network intelligence for the purpose of service selection and to participate in reconstitution and restoral.

5. CONCLUSION

Integration offers the next step in the evolution of the interconnection of communications networks.

Internetworking particularly in the data environment has achieved a high degree of connectively between networks. However internetworking has developed as a means to overcome physical network incompatibilities. If internetworking is pursued as an acceptable solution, then incompatible networks will continue to flourish.

This paper has attempted to assess the merits of internetworking and integration as two alternate approaches to providing reliable, survivable, efficient and well managed Defense communications systems. It appears that a turning point has now been reached. It is time for network planners and administrators to step back and assess where the current Defense communications network philosophies are leading. There appears to be two choices:-

a) That Defense networks continue to be viewed as independent networks with the Defense Data Network (DDN) trying to solve its interoperability and security problems through a complicated network of gateways while the Defense Switched Network (DSN) concentrates on voice users.

or

b) That the coordinators and users of the DDN and DSN who have philosophically adopted a long term goal of integration, modify and accelerate existing plans to achieve initial integration at an earlier stage.

5.1 Proposals

a) Equip all new DSN switches with a minimum number of ISDN ports.

As a first step all new DSN switches should be equipped with at least a limited number of ISDN Basic Rate Interface ports and the necessary software to support access to the DDN. The collection of ISDN users around each switch will form an ISDN network, however ISDN services between these networks will not be possible until the switches are linked by Common Channel Signaling.

b) Shift resources from Internetworking R&D to Integration R&D.

If ISDN hardware and software can be deployed and used effectively as a DDN entry point for data users then some of the financial and manpower resources currently planned for development of data network gateways can be shifted to ISDN access systems. This shift of resources away from internetworking toward integration could be applied in two areas. Firstly, for the procurement of more ISDN access ports on DSN switches and secondly for research and development using existing internetworking knowledge, toward more sophisticated ISDN access systems.

c) "Connect" the packet handler function of the new DSN switches into the DDN.

The new DSN switches can achieve a reasonable level of integration through the use of switch software that provides a packet handler function within the switch (AT&T 5ESS) or via an intimately linked but external packet switch (Northern Telecom DMS-100 family) These packet handler functions need to be coupled to the DDN as soon as possible.

d) Develop an intelligent transmission multiplexer

Near term integrated transmission could be achieved by intelligent T1 multiplexers that dynamically combine the output of co-located circuit and packet switches to more effectively use available trunk bandwidth. The RADC Voice/Data Integrator program [2] is providing a technology demonstration of this concept.

e) Assess the "cost" of internetworking

The idea of near term integration as discussed above should not be rejected on the basis of increased cost without an assessment being made of the current and future "cost" of internetworking in the DDN. The "cost" includes both financial resources devoted to research & development and performance. The DDN should be investigating the performance penalties of information traversing through multiple gateways. An assessment also needs to be made of the performance and survivability of the network of gateways concept when placed under stress. It may be found that this concept is only workable in an unstressed environment and that highly interconnected public networks offer greater opportunities for survivability.

The result of this assessment may be that significant resources currently applied to internetworking could be shifted to satisfy the same objectives but through integration which after all is the common goal of DoD networks - or is it?

References

1. J. J. Salerno, Increasing Circuit Switched (AUTOVON / DSN) Trunk Bandwidth RADC-TR-85-173, Rome Air Development Center, Griffiss AFB, NY.

2. Voice / Data Integrator Design, (RADC Contract # F30602-87-C-0139). Rome Air Development Center, Griffiss AFB, NY.

3. CCITT, Integrated Services Digital Network (ISDN), Recommendations of the I Series, October 1984.

4. R. Braden and J. Postel, Requirements for Internet Gateways, Internet RFC 1009, June 1987.

5. R. A. Mercer and W. L. Edwards, Issues in the Migration of Military Communications to an ISDN, 1986 IEEE Military Communications Conference.

6. HQ USAF/SCTI, Air Force Communications-Computer Systems Architecture, Vol 4, Long Haul Information Transfer, April 1987.

7. HQ USAF/SCTI, Air Force Communications-Computer Systems Architecture, Vol 5, Local Information Transfer, April 1987.

WIDEBAND PROPAGATION THROUGH A FORESTED MEDIUM

Boyd C. Fair

Information Sciences and Technology Center
SRI International
333 Ravenswood Avenue
Menlo Park, California 94025

ABSTRACT

The U.S. Army has developed a Wideband Propagation Measurement System (WPMS) capable of gathering the multipath-delay-spread and excess-path-loss data necessary to characterize, and ultimately model, urban and forested communications channels. This paper presents the results of two sets of very wide bandwidth measurements taken in a trunk-dominated forest at Fort Lewis, Washington, the second set being conducted after the forest had been selectively thinned by 27% (based on stem count).

During these experiments, propagation measurements were made at ranges up to several thousand feet, at a various antenna heights between 12 and 65 feet, and at several carrier frequencies between 400 MHz and 1750 MHz. Data were taken at both vertical and horizontal polarizations for each range/height/frequency combination. The radiated signals consisted of an RF carrier bi-phase modulated with a pseudo-random noise waveform clocked at 250 MHz, producing an instantaneous null-to-null probe-signal bandwidth of 500 MHz. The received signal was correlated with a time-delayed replica of the transmitted signal in the receiving system to allow measurement of the relative time-of-arrival of the radiated signal and its multipath components to a resolution of a few nanoseconds. The overall path loss was determined by comparing the measured transmitter power to the total received power in the receiver's IF amplifier. The difference between the overall and the free-space path loss was calculated to determine the excess path loss caused by the channel.

INTRODUCTION

The U.S. Army Communications and Electronics Command (CECOM) has a program to develop a theoretical communication model capable of predicting radio propagation characteristics of wide-bandwidth signals in a tactical (forested) environment. In support of this program, SRI International (SRI) designed and built experimental apparatus for a series of field measurement programs that we conducted to validate the new propagation models.*

We conducted two measurement campaigns in a Douglas fir forest located at Fort Lewis, Washington, known as the South Perry Woods. The forest was selectively thinned between the two sets of measurements, which allowed us to gather a unique set of data that shows the effect of tree density (stem count) on excess path loss and multipath delay spread without introducing the added complications of differing tree heights, undergrowth characteristics, tree age, and the like. In addition, we obtained information on the propagation characteristics of a trunk-dominated forest, from which the early theoretical models could be verified and enhanced. We gathered data using a probe signal that produced a 500-MHz instantaneous null-to-null bandwidth created by bi-phase modulating an RF carrier with a pseudo-random noise code clocked at 250 MHz.

* Work performed under Contracts DAAB-07-82-C-J097 and DAAB-07-85-C-K569, U S Army Communications-Electronics Command (CECOM) COM-ADP Center, Fort Monmouth, New Jersey 07703

EXCESS PATH LOSS

We define excess path loss as the difference between the path loss that occurs within the forest and the loss that would occur in free space. From scattering theory we would expect the excess path attenuation to vary exponentially with path length and the number of scatterers in the channel

Range Dependency

Figure 1 shows the vertical and horizontal excess path loss data obtained over several paths as long as 1500 ft. at frequencies of 400 MHz before thinning, and over the same paths measured at 450 MHz after the trees were thinned by 27%. All measurements were taken at antenna heights of 38 feet, where the forest is clearly trunk dominated.

Linear regression curves fitted to the vertically polarized data have a slope of 0.027 dB/ft before thinning and 0.017 dB/ft after thinning--a difference that represents a 37% reduction in the attenuation rate. A similar curve-fitting procedure applied to the horizontally polarized data yields a slope of 0.022 dB/ft before thinning and 0.018 dB/ft after thinning--or a reduced attenuation rate of only 18%. The average reduction in excess path loss of approximately 23% compares favorably to the 27% reduction in the stem count, and confirms the exponential relationship between the excess attenuation and number of scatterers in the path.

Frequency Dependency

Figure 2 shows the initial excess path loss data taken on a 700-ft path at a height of 38 ft., along with the experimental data recorded on the same path after the woods were thinned The data obtained before thinning show a strong frequency dependency of approximately 0.017 dB/MHz for the horizontal polarization, but very little (if any) dependency for the vertically polarized data. The data obtained after the forest was thinned show that the horizontal polarization frequency effect was reduced to approximately 0 008 dB/MHz--a reduction of 53% (dB/MHz) from the before-thinning data The frequency dependency in the vertically polarized data is only -0 005 dB/MHz, but there is extensive data scatter making this number suspect.

Height Dependency

The original data taken at the South Perry Woods indicated a slight height dependency at 850 MHz and 1050

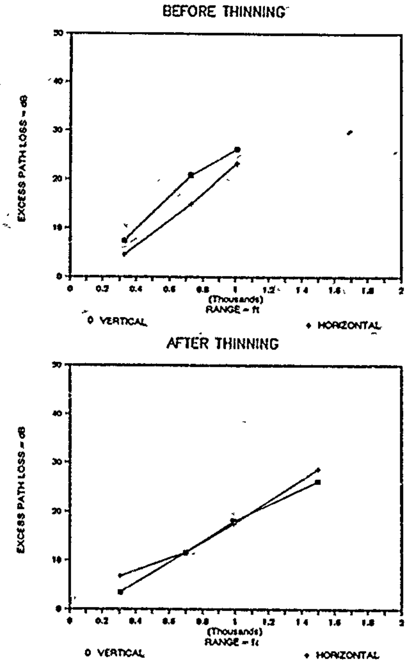


Figure 1 Excess Path Loss as a Function of Range

MHz, but little or no height effect at 400 MHz. Figure 3 shows the before- and after-thinning 400-MHz data set taken along a 1000-ft path. The after-thinning data also show no significant height dependency. A comparison of the y-axis intercepts determined by linear regression curves fitted to each data set shows the vertically polarized data are reduced by 33% (19.1 dB vs 28.7 dB), while the horizontally polarized data are reduced by 52%. The larger apparent reduction for the horizontally polarized data is significant and surprising, given the forest was only thinned by 27% If we had assumed a zero-height dependency for the trunk-dominated forests (horizontal curve fit), the percent reduction between the before- and after-thinning data sets would have been much smaller and more closely matched the 27% stem-count reduction. Additional data should be taken and analyzed to resolve this uncertainty.

Polarization Dependency

Figure 4 presents plots of three sets of data. (1) all range data taken at 400/450 MHz at a height of 38 ft; (2) all frequency data taken on a 700-ft path at a height of 38 ft, and (3) all height data taken on a 1000-ft path at a frequency of 400/450 MHz Although the major polarization dependencies were not affected by the thinning of the South Perry Woods, the magnitudes of the dependences are smaller in the less-dense forest Perhaps the most significant effect is the strong frequency dependency noted on the first set of measurements. This effect shows that the 400 MHz signals have a greater attenuation in the vertical polarization while the higher frequencies are more attenuated when transmitted horizontally. There appears to be a cross over in the polarization data between 500 to 700 MHz.

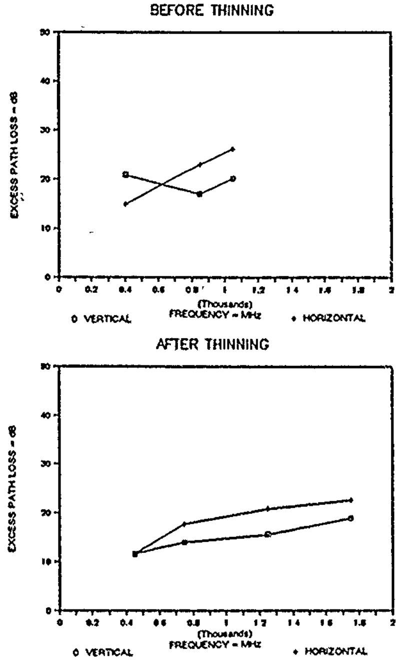


Figure 2 Excess Path Loss as a Function of Frequency

MULTIPATH DELAY SPREAD

Wide-bandwidth path-loss measurements tend to be more stable than narrowband measurements because all the individual communication paths are totally resolved and averaged to obtain the path loss measurement. The complexity of the communications channel manifests itself in the multipath delay spread characteristics. The fully resolved time-varying impulse response (TVIR) is not easily characterized by a deterministic function, however. Fully resolved TVIR measurements are relatively new; hence the wide variety of characterization schemes being used today by scientists and engineers will undoubtedly undergo refinements that will allow extraction of finer-scale details in the future.

The TVIRs measured by the Wideband Propagation Measurement System (WPMS) are complicated and highly variable in their detailed structure, as shown in Figure 5. To use these data to verify or improve theoretical scattering models requires that some simple measure of the average delay spread characteristics be employed. It is impractical to repeat measurements a large number of times with small vertical or

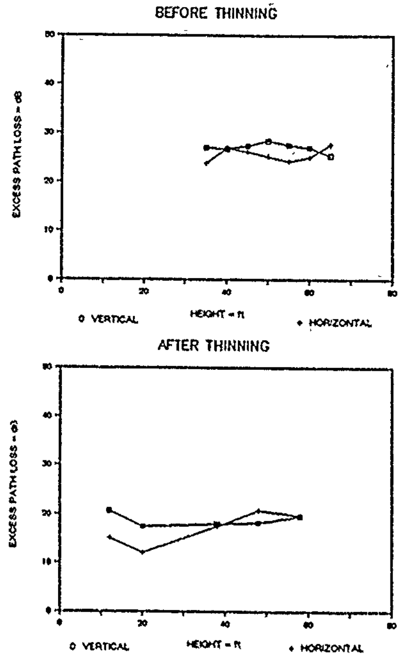


Figure 3 Excess Path Loss as a Function of Height

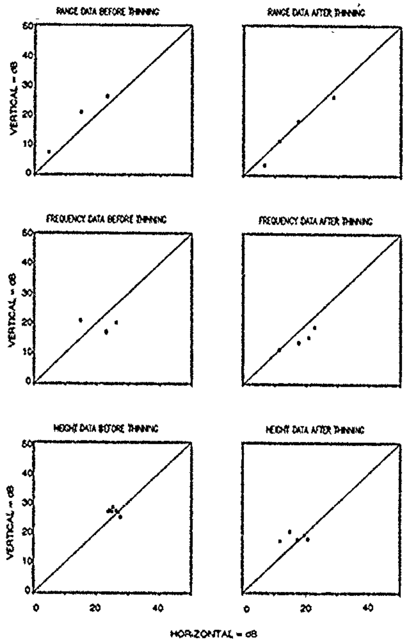


Figure 4 Excess Path Loss Polarization Dependency

horizontal displacements to determine the average multipath characteristics of a communications channel. Some other smoothing procedure must therefore be used.

Because the scheme we have used to determine the width of the measured multipath delay spreads has no inherent smoothing, it admits too much fluctuation to allow the fine details of the delay spread characteristics to be analyzed in detail. Nevertheless, it can be used for simple comparisons of the large-scale scattering characteristics of the channels being measured.

The delay spread measure used is based on the time difference between the first and last crossing of a specified received signal level. To provide a common normalization, we chose a level for characterization that was four times the average intensity of the TVIR. This level is high enough to limit the effects of noise in the system, but low enough to measure the highly delayed (attenuated) paths. This level and the resulting width determination (offset horizontally for clarity), are shown on Figure 5.

Range Dependency

Delay spreads from a few hundred nanoseconds to more than 1.5 μs were observed during the first set of measurements Except for a single data point, the initial vertically polarized delay spreads increased monotonically with increasing path length. The corresponding horizontally polarized delay spreads were smaller and showed a much weaker correlation with increasing path length.

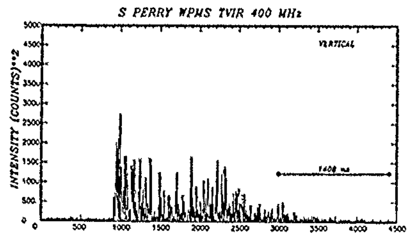


Figure 5 Sample TVIR

Figure 6 shows both the pre- and post-thinning multipath-delay-spread measurements for the data set described earlier The antenna heights used for the two measurements were identical, and the specific locations of the antenna towers were within a few feet of being the same for both measurements

As can be seen, the trend of greater delay spread with increasing range exists in both sets of measurements, with the vertically polarized delay spreads being substantially greater than the horizontally polarized data Because of the limited number of data points and the noisy nature of the measurement of delay spread itself, it would be futile to attempt to extract too much detail from the data. Nevertheless, if the slope of the straight line fit to the original vertical measurements (1.3 ns/ft) were compared with the same measurements made after the thinning (0.3 ns/ft), the (apparent) reduction in delay spread caused by the 27% thinning of the forest is found to be 77%

Comparing similar straight-line fit data to the horizontal data obtained for the before- and after-thinning measurements (0.23 ns/ft vs. 0.21 ns/ft) yields an apparent reduction in delay

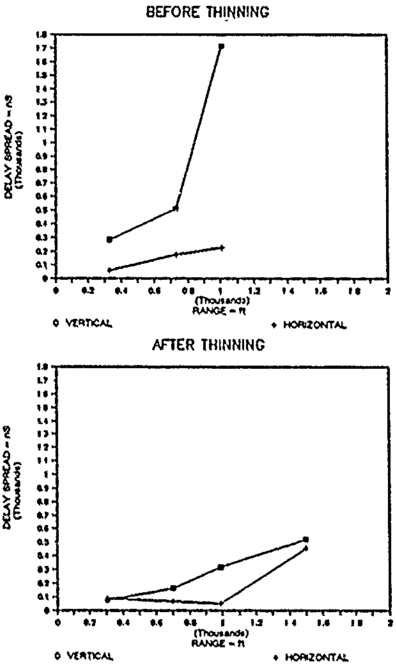


Figure 6 Delay Spread as a Function of Range

spread of only 9% This large discrepancy between the two polarization data sets may be due in part to the delay spread characterization technique. However, visual examination of the unprocessed TVIR data shows that the change in the vertical delay spread caused by the thinning is significantly greater than the horizontal, confirming the large reduction in the vertical multipath delay spread calculated and the accuracy of the delay spread determination technique.

Calculating the average of these two percent-reduction numbers, we obtain 43%. This average change shows that there is at least a first-order correlation between the delay spread as a function of range and the tree density of the forest being measured.

Frequency Dependency

Figure 7 summarizes of the before- and after-thinning frequency-dependent data In general, the scatter of both data sets appears to be greater than any trend that may be present Therefore, although the delay spread in the after-thinning data is consistently lower than in the initial measurements, determining a percentage reduction of delay spread would not be meaningful, because of the large variation in data.

Height Dependency

Figure 8 shows the original 400-MHz data taken on an approximately 1000-ft-long path and a similar set of 450-MHz data taken along the same path after the forest was thinned. The lack of height dependency is evident in both sets of data This finding is not surprising, given that all the measurements taken were below the canopy of the trees. Although the measured vertical delay spread is larger than the horizontal delay spread on both sets of data, the ratio between them is

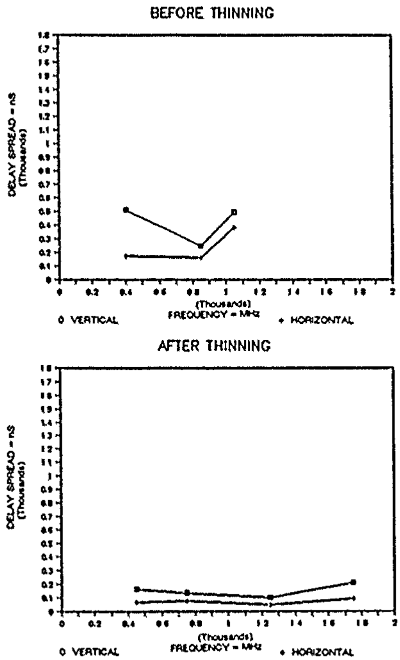


Figure 7 Delay Spread as a Function of Frequency

smaller in the after-thinning data This effect was expected given the similar properties of the forest (tree height and trunk diameter) for the two sets of measurements.

A surprising effect is the large difference in the magnitude of the two sets of delay spread measurements. The original vertical delay spread numbers, for example, were about 1500 ns, while the after-thinning measurements yielded a delay spread closer to 400 ns. These data would indicate a delay-spread reduction of nearly 70%. A similar reduction can be seen for the horizontally polarized data.

One possible explanation for this large change in delay spread is the presence of logging roads cut while the forest was being thinned. SRI experimenters noticed that one road was nearly aligned with the communications path being measured. More research on the effect of this road on delay spread characteristics would be valuable.

Polarization Dependency

Figure 9 illustrates the strong polarization effect encountered by a broadband UHF signal passing through a trunk-dominated forest. The scales are 1800 µs maximum for both axes.

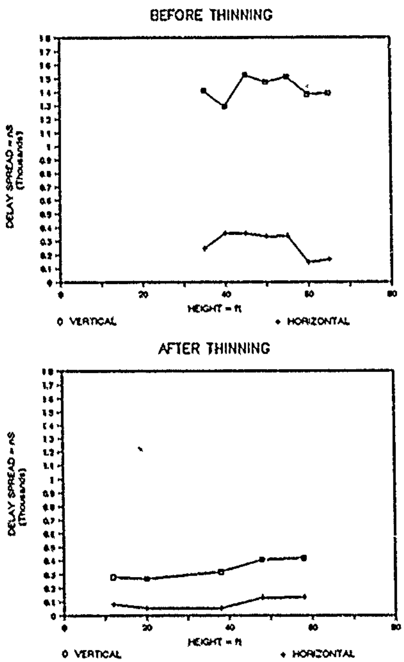


Figure 8 Delay Spread as a Function of Height

The delay spreads measured after thinning are consistently smaller than those of similar data taken before thinning. In addition, the polarization dependency is not strong, particularly in the data set taken as a function of antenna height. The smaller delay spread is partially caused by the reduced stem count and partially by the slightly different paths used for the two measurements.

Summary

The experimental data taken in the South Perry Woods show that the tree density (stem count) has a significant effect on the magnitude of the excess path loss and multipath delay spread for a very-wide-bandwidth signal. These effects result in a higher excess path loss for horizontal polarization and delay spreads sometimes greater than the free-space propagation time for vertical polarization. Little height effect was noted for either excess path loss or multipath delay spread.

These data are now being used to validate the wide bandwidth model being developed. When completed, this model will be an invaluable tool for the Army to use to predict the performance of future wide bandwidth communication systems.

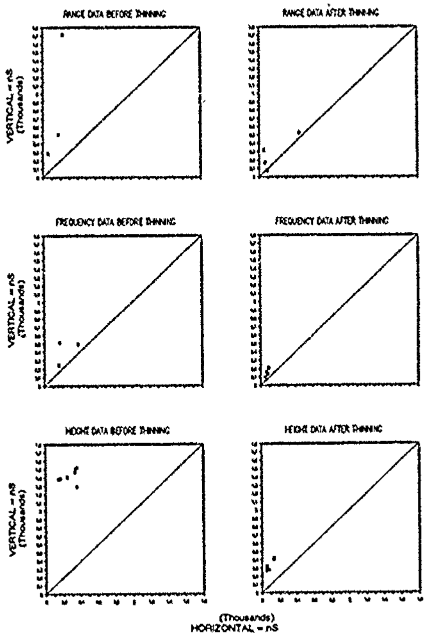


Figure 9 Delay Spread Polarization Dependency

A STATE OF THE ART ASSESSMENT IN NETWORK MANAGEMENT USING TECHNIQUES FROM ARTIFICIAL INTELLIGENCE

Michael Bereschinsky
US Army CECOM
Center for C3 Systems
Ft. Monmouth, NJ

Charles J. Graff
US Army CECOM
Center for C3 Systems
Ft. Monmouth, NJ

INTRODUCTION

This paper will present a state of the art assessment of the Automated Network Management System (ANMS) and the application of artificial intelligence (AI) to the network control problem. The paper will be presented in two parts: first, a description of the ANMS which will provide raw, filtered, and processed data to the AI based network control algorithms; and second, a discussion of the possible AI based network control algorithms, which will use the data from the ANMS to make better global, network decisions and also to provide better network performance in the uncertain environment. The ANMS will be discussed first.

ANMS OVERVIEW

The Automated Network Management System (ANMS) is an integrated set of hardware/software (currently running on a color Sun 3/160 computer in C under Unix) that is providing an automated network monitor capability.

This capability includes the near real time monitoring of internet gateways, packet radios of various flavors, packet switching nodes, and host computers running the Unix operating system. In the future the ANMS will be enhanced to include automated network control, AI assisted monitoring and analysis, and AI assisted network control. By network monitoring is meant the collection and reduction of network status data, followed by graphical representation on a CRT of the network status. By network control is meant modification of monitored network behavior that is either problematic or nonoptimal. The ANMS will be used by network operators/administrators and network analyst/researchers.

Network operator/administrative personnel will use the system to maintain continuous network operation when network problems/failures occur, and to fine tune network performance when such performance becomes nonoptimal. Network analyst/researchers will use the system to diagnose complex and chronic network problems, and to perform research on network system/component algorithms.

The ANMS collects data on the status of the various network components in the the network and presents the resultant "monitored" information in graphical representations known as "views" on a CRT. There are topological views which consist of detailed network diagrams indicating network component connectivity and associated information. There are table views consisting of network component status information. The use of logfiles of ANMS monitored data in conjunction with the ANMS view capability permits detailed analysis of network performance after the fact. This capability will be useful in the development of network algorithms.

ANMS SYSTEM DESIGN

The ANMS design consists of two major software modules, the Distributed Management Module (DMM) and the Client Process Module (CPM), and an interconnecting Network Management Protocol (NMP). The typical operation of the ANMS consists of a CPM (which is connected to a single DMM) sending queries or control commands to the DMM, and the DMM sending results back to the CPM via the NMP. Also contributing to ANMS operation are other DMMs and CPMs deployed throughout the network, various network components queried by the DMM, and human users querying the ANMS via the user interface function of the CPM as shown in figure 1. Other DMM functions include forwarding queries/control commands to DMMs when necessary, and storage of network management information in the database. The other CPM functions include network management services as defined by the ISO standards. Presently these functions include performance management and fault management. The ANMS performance management includes the retrieval and presentation of the network monitor and control data obtained from the network components. The ANMS fault management includes the detection and analysis of problematic or nonoptimal network behavior. The NMP is also used for communication between cooperating DMMs. The NMP was created to permit generation of query and control commands that were independent of the network component protocols such as HMP (Host Monitoring Protocol). The NMP is an application layer protocol that is similar to the ISO Common Managment Information Protocol (CMIP). The NMP uses DOD TCP/IP protocols as the transport protocol.

The ANMS can provide a distributed automated network management capability. The CPM's and DMM's can be redundantly located on various host computers in the network. Any CPM can provide an ANM capability by communicating with any of the DMM's located throughout the network. The redundant DMM's and CPM's in the network provide survivable automated network management.

Although it is true that many modern data communication systems are self-managing (once initialized and running), there will be times when network operation has to be modified by actions external to the core operational network itself. An example of this is the case of a local area network (LAN) connected via a gateway to a wide area network (WAN). If, for some reason, the gateway became problematical and started to flood the WAN with extraneous packets, the ANMS monitoring function would detect this condition, and the ANMS control function could remotely reboot the gateway, or down line load new software over the net, or remotely turn off/disconnect the gateway from the net.

AI ASSISTED ANMS

For such network problems, CPMs are being developed that incorporate AI techniques to assist in network analysis. One AI assisted network analysis implementation of the ANMS called the Intelligent Network Manager (INM) that detects problems in internet gateways has been developed. The INM consists of a relational cache, a set of views, and a collection of experts. The experts are small independent software modules containing a portion of the ANMS network analysis knowledge. This approach utilizing multiple experts instead of one large expert permits use of the appropriate reasoning method to be used in each expert, depending on the expert's problem domain. Monitored gateway data is stored in the cache for efficient access by the experts as shown in figure 2. The experts use their network analysis expertise in conjunction with the cache data to detect and diagnose problems. Experts communicate with the INM operator by sending descriptions of suspected network problems to the Alert View.

APPLICATION OF AI TECHNIQUES TO NETWORK CONTROL

A brief study effort was performed to address the following two questions: to determine the applicability of the emerging AI techniques to packet switched network control algorithms, and to recommend a first order implementation approach of the AI techniques into network control structure.

The study addressed the applicability of the AI techniques to the following aspects of the network control problem:

i) link quality measurement/prediction

ii) network connectivity assessment

iii) network/internetwork routing

iv) network initialization

v) link and node failure determination

vi) network reconfiguration

The study revealed that the most promising areas for future study were network/ internetwork routing and network reconfiguration. The decision to select these two areas for further study was influenced by the fact that a large body of human knowledge and expertise exists for both these areas. In addition, the timing requirement for decisions made by AI decision techniques can be met by existing technology.

The network routing problem has been studied extensively for many years. The routing function can be considered to have two parts: one for network connectivity assessment and maintenance, and a second for path calculation and packet forwarding. The connectivity assessment part of the routing algorithm must deal with uncertainty in network information in a dynamic network environment. Additionally it is well known that distributed network algorithms are prone to loop generation, and do not necessarily provide optimum load balancing. It is in these areas that the use of AI techniques seems to hold most promise.

The second major area for the application of AI is the network reconfiguration problem. For the tactical Army, the network reconfiguration problem is compounded by frequent relocation of forces, (with resultant connectivity changes), node attrition, and the poor connectivity environment due to jamming and terrain masking.

Furthermore, the network must often be reconfigured based on partial (and possibly erroneous) information by personnel under the stress of battlefield conditions. These problems point to the need to have an automated system to aid in or perform completely network reconfiguration.

INTELLIGENT ROUTING ALGORITHM

While much work in routing has been done in the past, the majority of networks use algorithms that utilize a single minimum cost path for a given source - destination pair. The cost of the path is determined by the simple addition of cost of the link for each link in the path. The link metric is predetermined but is computed based on measurements for each particular link. It is well known that the class of distributed algorithms performs best for networks that are composed of homogenous links where the connectivity changes slowly with time . In most networks, the route computation for each source destination pair is done independent of traffic type. In most practical network designs, the algorithm that updates the connectivity assessment for a given node reacts and responses to the dynamics of the network environment, but do not predict or anticipate them.

For these reasons, it was decided to pursue the development of an Intelligent Routing Algorithm (IRA). The IRA attempts to perform global network optimum utilization as well as loop prevention to provide an optimum level of performance. The IRA has two distinct components: a replicated component that acts much like a conventional distributed routing algorithm, and a centralized AI component that provides a form of topology control that yields global optimization as well as loop detection/prevention. The IRA is presented in figure 3. The distributed routing component is fully replicated at each node in the network. The routing of packets is based on routing tables that are contructed in part from data distributed by the centralized AI topology control. Each of the distributed nodes provides connectivity and network status information to the centralized AI component. The centralized AI component uses this network state information to manipulate the individual nodes view of the network topology to provide some optimum load sharing. Additionally, the centralized AI component will also insure that routing loops do not occur.

Under this scheme, it is quite possible for different nodes to have different views of the network topology. The centralized AI component continuously monitors the status of the network to determine if the network is in or will evolve to an undesirable or substantially suboptimal state. If the network is viewed to be in such a state or evolving toward such a state, then the link metrics are recomputed and forwarded to the distributed nodes to provide network loading in a more uniform manner.

NETWORK RECONFIGURATION

The second major area for the application of AI to packet network technolgy is the network reconfiguration problem. A large body of knowledge and expertise is available for the network design problem. The basic approach is to utilize existing techniques augmented with mechanisms for knowledge representation, reasoning, constraints for equipment attributes, terrain, and to integrate the design of the local assess and backbone into a single design problem.

It should be noted that the reconfiguration process will be running continuously, by taking measured data from the actual system in operation as well as new (future) input requirements. In many cases the information supplied as input will be partial (and possibly erroneous) and will include historical traffic profiles as well as future traffic estimates. Inputs will also incude routing and congestion performance histories. Potential jamming threats and possible future connectivity outages will also be an integral part of the network design process. Any future clustering of forces or mobility requirements will also be supplied as input. The output of the reconfiuration process will be a complete network topology design, including the backbone and area access, that will support current and planned tactical needs.

CONCLUSION/SUMMARY

This paper presented a brief summary of the work completed to date in both the ANMS, as well as the efforts to incorporate advanced AI techniques in network control. The ANMS is currently an ongoing program. The ANMS is currently undergoing test and evaluation at testbeds at Ft. Monmouth, NJ and Ft. Bragg, NC. The summary of the application of is by no means exhaustive and indicates that much work remains yet to be done. The utilization of AI techniques in uncertainity management, knowledge representation and reasoning, offer clear advantages over the traditional network approach.

Footnote

The results presented in this paper were developed by BBN LABS, Inc under DARPA contract # MDA 903-83-C-0131, and SRI International under US Army CECOM contract # DAAB07-86-D-A035. The authors of this paper provided the program definition and project direction/management for this effort.

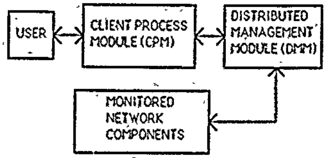


FIGURE 1. ANM DESIGN

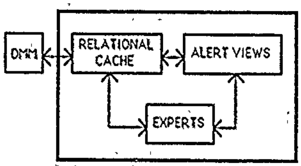


FIGURE 2. INTELLIGENT NETWORK MANAGER DESIGN

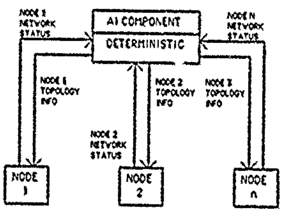


FIGURE 3 INTELLIGENT ROUTING ALGORITHM

REFERENCES

1. Feridun, Metin, et al, "ANM: Automated Network Management System", IEEE Network Magazine, pp. 13-19, March 1988

2. Ortiz, Charles J., et al, "Applicability of Artificial Intelligence Techniques to Packet-Switched Networks", Final Report for SRI Project 3000, SRI International, Menlo Park, CA. (March 1987), under CECOM Contract DAAB07-86-D-A035

A FACT-ORIENTED DATA DISTRIBUTION SYSTEM

Eric G. Heilman & Samuel C. Chamberlain

US Army Ballistic Research Laboratory,
Aberdeen Proving Grounds, MD 21005-5066

ABSTRACT

With the influx of available computer technology, vast amounts of information can be moved and manipulated. Currently, information exchange between "fighting level" forces (i.e., brigade and below) is constrained by relatively low frequency tactical radios that do not support the high bandwidths common in most modern computer networks. One potential solution to this information flow problem is to reduce the amount of data exchanged by representing military information and concepts in their most basic, primitive form. Transmission of abstracted military concepts requires less bandwidth while also providing battlefield information in a more accessible form.

The Ballistic Research Laboratory (BRL) is developing an experimental *Information Distribution System* (IDS) based on primitive data abstractions (termed "facts") of military concepts. The system consists of a RAM resident storage facility (factbase), a connectionless communications protocol (Fact Exchange Protocol), and an explicit methodology to describe a battlefield in terms of computerized data abstractions. Further, a set of *distribution rules* that describes situations in which fact exchanges are warranted between tactical nodes is being developed as part of a security control module (SCM) that resides at the heart of the IDS.

In order to present IDS battlefield information to the system operator several "application" programs were developed to manipulate facts. These computer programs include Working Map (WMAP), Organizational Chart (ORGCHART), and Fireplan. The development, implementation, and special considerations that influence these user interfaces reveal alternative methods of battlefield representation through the application of computer science techniques.

INTRODUCTION

The Ballistic Research Laboratory's (BRL) System Engineering and Concepts Analysis Division (SECAD) is developing a fire support application to demonstrate information distribution concepts within the US Army LABCOM's Smart Weapons Systems (SWS) COOP project. The tactical theme of the IDS portion of the project will revolve around the dynamic actions of five key artillery nodes (the maneuver brigade fire support element, a direct support field artillery battalion operations element, and three maneuver battalion fire support elements) that are responsible for two key fire support functions: *fire support control & coordination* (FSCC) and *field artillery tactical operations* (FA TAC OPS). Communications between these five nodes will be recorded and examined during simulations of various combat situations. Several tactical vignettes will be employed to create critical combat situations. Responses to these situations will require impromptu plan changes, realignment of forces in the command chain, the use of a secure and silent mode for battlefield information gathering, and an automated method of tracking and reporting battlefield information.

A major goal is to develop tactical computer science technology that can support fighting level commanders and soldiers who must contend with highly dynamic, unpredictable, and hostile combat environments. Current tactical command and control systems support digital data exchange via formatted "messages" (character strings) or graphics symbols that must be interpreted by a human operator. These systems fall short in two key respects: first, the information is not in a form that computers can manipulate in a sophisticated manner (i.e., "understand"), and second, the use of inefficient information exchange protocols may exceed the maximum bandwidth available on VHF-FM and HF-AM radios common to (and required by) lower echelon units. These inefficient protocols produce significant electromagnetic signatures due to the large amount of overhead required in the exchange of battlefield information and the manner in which the protocol accomplishes this exchange.

The BRL "fact oriented data distribution system" incorporates several new concepts in an effort to explore techniques that provide more flexibility and survivability to the information distribution function in command and control systems. Flexibility is enhanced via a freeform *distributed factbase* (DFB) and associated *fire support control capability profiles* (CAPs) that define the standard operating procedures of a node. Survivability is enhanced through minimizing electronic emanations by: transmitting only significant information (as determined by the commander), transmitting information in its purest form, taking advantage of "overheard" information, providing a "radio silence" (Emission Control, or EMCON) mode of operation, and using multicast transmissions when possible. A *fact exchange protocol* (FEP) is being developed that will exploit these and other features to create a streamlined, connectionless transport layer protocol. These guiding concepts suggest the construction of a *computationally intensive* protocol that allows information transfers only when internal computing fails to yield significant results. Figure 1 shows the basic structure of the Information Distribution System. The distributed factbase (DFB) is composed of four conceptual modules: the factbase for information storage, the Fact Exchange Protocol (FEP), a Package Protocol to enable connection between the DFB and application programs, and the Security Control Module (SCM) that controls access to the factbase.

The main purpose of an information distribution system is to support the exchange of ideas or concepts. At the superficial level this may simply be a unit's location, but underlying such a simple concept are the reasons for needing to know a unit's location and the frequency of this information requirement. This reasoning process can be used to define the structure of tactical information and to regulate its flow between various nodes (units) of the system. If a terse, computer efficient means of communicating between the unit nodes of the experimental system is to be developed, the incorporation of military science is essential in the definition of data abstraction primitives reflecting basic battlefield concepts. By using these primitives to exchange information between command and control nodes, a more flexible and survivable system results. However, a significant effort must be expended "up front" as combat developers and computer scientists work together in order to build a canonical list of data abstractions describing basic military concepts.

Information, provided through the media of data abstraction, is maintained in a form suitable for manipulation by sophisticated computer application programs. Using this scheme, one of the tasks of the "high-level" user application programs becomes converting data abstractions into a form suitable for user assimilation and manipulation. The main focus of this paper is a description of these application programs, however, a review of the basic DFB software is first presented to provide an explanation of the functional environment on which these applications depend.

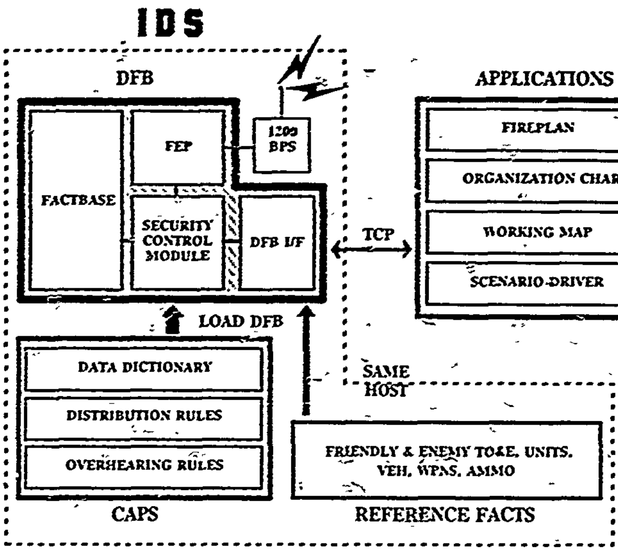


FIGURE 1. Block Diagram of IDS Software Components

FACTS and FACTBASES

Information pertaining to a combat situation is stored within a RAM resident, distributed factbase (DFB) as a collection of many interconnected *facts*. A fact is an instantiation of a pre-defined *fact type* that can be structured to describe any item, activity, or event common to a battlefield. As facts are entered into the DFB (or *stated)*, each is assigned an unique fact identification number (*fact id*) that consists of the four byte Arpanet host address of the computer on which the factbase resides and a four byte integer controlled by the resident factbase. In general, a fact consists of a header and one or more *fact items*. There are five possible data types that may be included in a fact definition. integers, floating point numbers, character strings, references (fact ids of other facts), and lists (a collection of any of the above data types).

Every fact can be categorized as either a dynamic fact, reference material, or a meta-fact. *Dynamic facts* describe changing battlefield events or activities and are stated by a user or an application program Every dynamic fact is associated with its factbase of origin (host) by its fact id. *Reference material* facts describe static "reference" information (e.g., Tables of Organization and Equipment (TO&E), vehicles, equipment, and ammunition) and have *fixed* fact ids that are common to *all* factbases. (Note. a special host address occupies the first four bytes of the fact id of a reference fact). This characteristic allows referral solely through the use of fact ids, thus significantly reducing the amount of data that must be transmitted in many cases. Reference material facts are pre-loaded into each factbase before initialization and are never created nor updated unless there is a doctrinal or equipment modification. The one exception is *unit* facts which represent specific military organizations such as the "1-51 Field Artillery" or the "USS Abc". Unit facts are never created by the user (only congress can do that!) and so they too have static fact ids and are reference material. However, they can obviously be updated (e.g., a unit's location, direction of travel, status of equipment, etc.) so they present a special class of reference material facts. *Meta-facts* are a special version of dynamic facts that are used to represent potential modifications of other facts. Since the facts in the factbase normally represent the "real world" situation, a fact type must be available to describe future modifications of current facts without modifying the representation of the current battlefield situation. A meta-fact simply contains a reference to another existing fact with a list of alternative values for items (fields) contained within the referenced fact. A typical use is to exchange future battlefield plans and options. Meta-facts

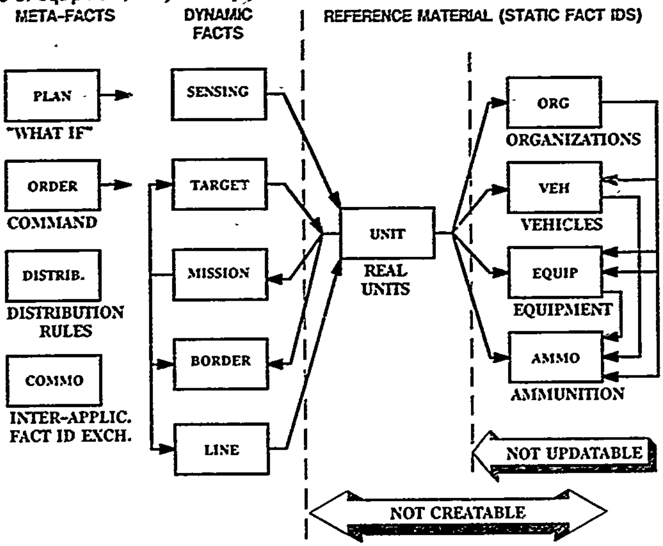


FIGURE 2: Data Abstractions (Fact Types) and Relationships

are also used to augment the rules governing information transfers between combat nodes. Everything about a node (a military unit) is stored as a fact to allow any information to be easily exchanged with any other nodes. Figure 2 shows the relationship between the canonical list of fact types (primitive data abstractions) implemented thus far. Arrows indicate references to facts of other types.

An important conceptual feature of the DFB is its ability to automatically initiate actions upon the reception of new information whether the information comes from other DFB's or local application programs. The mechanisms that enables a DFB to take automatic action are known as *distribution rules*. (Previously, a simpler form called a *trigger* was used.) A distribution rule is composed of a set of criteria that can be compared to incoming fact exchanges and a set of actions to be executed when the criteria is met. Whenever a fact exchange enters the DFB a comparison is made to the criteria of each distribution rule. Distribution rules will be explained further in the section about the Security Control Module.

FACT EXCHANGE PROTOCOL

Information exchange between individual factbases is supported by the fact exchange protocol (FEP). The FEP is a connectionless, reliable datagram protocol (transport layer) that can be wrapped in the standard DoD Internet Protocol (IP). The initial focus of the implementation centers on reliability with terseness utilizing overhearing, multicast, and other techniques to minimize transmissions. In the past, commanders and their staffs have kept themselves informed by simply listening to the voice transmissions occurring on several radio nets. Similarly, the collection of "free" digital data (at no cost in bandwidth) is a feature of the FEP. The possibility of collecting overheard information is provided to reduce retransmission requests that consume limited transmission bandwidth. To support this ability, the network layer protocol (DoD IP) must be modified to allow the passage of datagrams meant for other hosts. This produces two categories of datagrams received by the FEP, datagrams meant for the resident DFB and overheard datagrams. Datagrams meant for the resident DFB are acknowledged, while all datagrams received are forwarded to the Security Control Module (SCM) where user defined rules concerning overheard information determine the pertinence of the overheard information (for entry into the DFB).

Overheard datagrams must be meaningful. However, standard network protocols, like the DoD s IP, may arbitrarily fragment packets when they exceed the maximum transmission unit size (MTU) of the datalink protocol. Arbitrary packet fragmentation can cause a comprehension problem since information describing a fact may arrive in incomplete, meaningless pieces. Therefore, no information transmission can be larger than the datalink layer MTU of the communications channel. Since MTU values are easily determined in the low echelon tactical environment, this requirement is easily met. (In the BRL implementation, the channels will be Ethernet and FSK modems for VHF-FM radio, both with MTUs of approximately 1500 bytes.) Although unlikely, facts larger than the MTU will have to be divided at logical internal boundaries before being passed to the lower protocol layers (transport and below) so that any overheard datagrams are meaningful.

Conversely, several fact exchanges may be packed into a single MTU sized packet to reduce bandwidth utilization. A significant reduction in bandwidth usage is achieved by reducing the number of radio transmissions because each transmission requires a radio "preamble" that is often relatively large in comparison to the size of an information packet. Making each packet as large as the MTU minimizes the number of transmissions required to send information, which in-turn, minimizing the effect of the radio preambles. A A datagram no longer has a single destination host address associated with it since every fact exchange contained within the datagram may have a different destination.

FEP datagram are sent between hosts (or groups of host for multicast addressing). A datagram can contain several fact exchanges, each with its own header that contains a 32 bit fact exchange identification number. Fact exchanges, *not* datagrams, are acknowledged. The connectionless nature of the FEP required a selective (out of sequence) acknowledgment scheme for each fact exchange entity. The FEP decomposes incoming packets into separate fact exchanges and returns an acknowledgement for those fact exchanges intended for the resident host. It then forwards all fact exchanges to the SCM tagged as either intended for the resident factbase ("direct") or overheard. Figure 3 illustrates the FEP message header design.

Due to the wide variation in communication channel bandwidths (e.g., 10 Mbps Local Area Net (LAN) and 1200 bps FSK modems over the VHF-FM radios used at the fighting echelons of the Army), flow control concepts must be re-examined. For example, "windowing" (i.e., the allowed number of unacknowledged fact exchanges), "wait time-out" (i.e., elapse time to wait for an acknowledgement), and number of retry (before giving up) parameters are based on a priori knowledge of the destination host and communication channel being used. These parameters can also be set explicitly by the SCM when it passes a fact exchange to the FEP. Eventually, the FEP will provide the SCM with fact exchange timing measurements to be used to dynamically adjust these parameters. It is the low

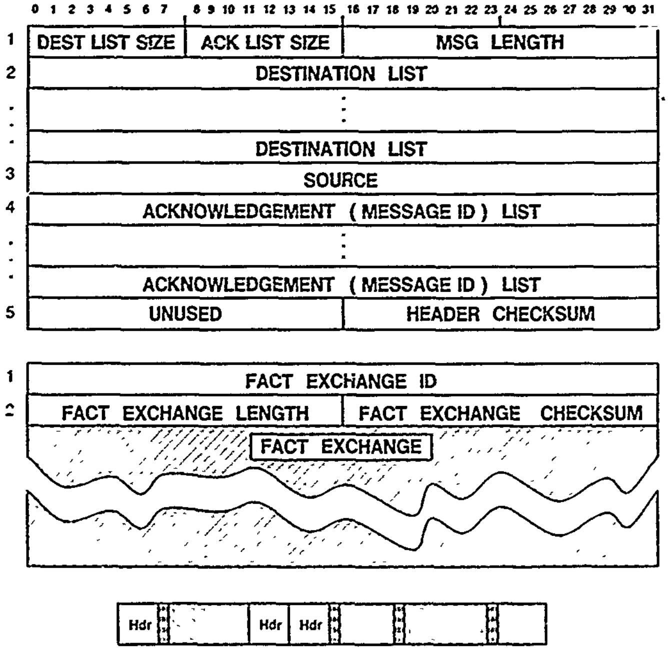


FIGURE 3. Fact Exchange Protocol Header

bandwidth communications channel and not the hosts that limit information exchange, since a single hop network (not an internet) is the communications medium, these network parameters are relatively easy to obtain.

Future plans include the implementation of a radio silence mode of operation to emulate the common voice communications practice of silently listening to radio nets, even for messages intended for the listener. A node may enter into EMCON mode and continue to receive information without returning acknowledgments. Upon discontinuance of EMCON mode, a "bulk" acknowledgment scheme will be developed to update only the most recently received fact exchanges (for a particular fact). Actual transmission of the bulk acknowledgement can be accomplished through the use of any appropriate media (e.g., radio, floppy disk and motorcycle, etc.).

SECURITY CONTROL MODULE

The Security Control Module (SCM) will concentrate initially on fact exchanges over the FEP (information to/from other factbases outside the local host) rather than with the application programs. The SCM has four major functional tasks: one, to determine when local information is significant enough to be transmitted to remote factbases, two, to determine if received fact exchange information should be entered into the resident factbase, three, to determine whether to honor requests for information from remote factbases, and four, to adjust FEP parameters (e.g., window size and number of retry) based upon communication channel measurements reported from the FEP (e.g., average fact exchange-acknowledgement round trip times).

A key tenet of this project is the elimination of unnecessary information exchanges, to include unnecessary requests for information. To do this, a set of rules governing information distribution is entered into the SCM of each DFB that keeps pre-defined units informed of the battlefield situation and plans. These rules insure that only "significant" data (as defined by the commander and staff) are transmitted. For example, it is normally not significant to report every round of ammunition that an artillery battery fires (although this is currently done). More appropriately, ammunition status should be reported when it reaches pre-designated values or rates. Theoretically, if the distribution rules are defined correctly, one unit should rarely, if ever, have to query another unit for information. *Distribution rules* have the structure:

    fact_type criteria actions

where

> *fact_type* is the name of a fact definition,
> *criteria* defines values to compare to incoming fact items,
> *actions* provide computer instructions should the criteria be matched

There are currently two major actions: *notify* a local application program, and *send* some information to another factbase. Two other actions are *state trigger* and *kill trigger* which maintains backward compatibility for a previous mechanism, called a trigger. This mechanism is similar to a distribution rule except that it compares its criteria to the fact exchange information after, rather than before, the information is entered into the factbase.

Status reporting is being implemented using the additional concepts of reporting depth and value thresholds. *Reporting depth* constrains the type of exchanged status information. For example, the traditional reporting depth of 2 is used by commanders. Therefore, every unit should have information about units 2 echelons below in the command structure (e.g., brigades track companies, battalions track platoons, etc.). This means that each unit sends status information about directly subordinate units to its parent unit (e.g., battalions send information about their companies to their parent brigade). Thus, a brigade would receive status information about all its companies, and if a status report about its battalions is required, one can be compiled by executing a DFB roll-up of all the companies. A units' *distribution envelope* is its purview based on an assigned reporting depth and any other special cases. A unit's distribution envelope can be dynamically modified based on the desires of the unit commander. In a low bandwidth tactical environment, an increase in the distribution envelope will decrease the usable bandwidth available to other units and produce a larger electronic signature. The distribution envelope concept will also be used to control the entry of overheard information since, for obvious security reasons, it would not be wise to incorporate all overheard information (i.e., should a node be captured).

The exchange of information is also controlled by *value thresholds* that appear in the criteria of rules. Thresholds are defined for fact items (fields) within fact types and may be compared to the current value in the factbase, or more often, to the last value transmitted to other units. For example, an threshold value could be placed on an artillery unit's ammunition count to identify when less than 100 DPICM rounds exist, or when 20% of the original basic DPICM load is reached, or whenever the ammunition count drops by 200 rounds of DPICM from the last reported value. The rule to automatically initiate an ammunition status update to a commanding unit could have the following format:

Fact type	Criteria	Actions
unit_type	ammo_num < 100	SEND
	&& ammo == "DPICM"	TO: PARENT

This states: "when your DPICM ammunition gets below 100 rounds then notify your parent unit of this fact". Information is exchanged only when these threshold values are exceeded (i.e., the information is determined to be "significant") thus controlling how often valuable bandwidth is used to transmit facts. The use of reporting depth and threshold values will provide the commander with a method to define and control the exchange of necessary information to reasonable bandwidth usage levels.

CAPABILITY PROFILES (CAPS)

Node initialization requires that the previously described information and constraints be entered into the SCM of each DFB. This information is stored in *fire support control capability profiles*, or "CAPs", and basically describes the standard operating procedures for information distribution of that unit. The CAPs, like all other information, are ultimately stored as facts to facilitate information exchange. Two types of parameters are currently included within the CAPs; 1) a *data dictionary* that defines every valid fact type acceptable to the factbase, and 2) the *distribution rules* that limit the flow of information on the network. The CAPs can be modified dynamically (even from remote locations) allowing all the items previously mentioned to be maintained as best appropriate for a particular situation. For example, specific CAP sets could be developed for offensive, defensive, special situation, or training scenarios and locally stored at each host. As battle conditions change, the CAP defined to regulate the new situation could be loaded into the SCM thus easily adjusting the operating conditions for the DFB. It is envisioned that the CAPs, which basically implement the standard operating procedures (SOPs) for a unit, should be developed by experts on doctrine and tactics (e.g., US Army Training and Doctrine Command schools) and perhaps modified by specialists in the upper echelons of a particular unit (e.g., corps, division, or brigade); the CAPs would not normally be modified by the soldier in the field.

APPLICATION PROGRAMS

Since one cannot "see" a factbase or the FEP, application programs were developed to demonstrate the capabilities of the system. Application programs permit controlled access to the information stored within a DFB. Information (facts) can be retrieved manually using queries or automatically using DFB triggers. New information may be entered by stating new facts or updating existing facts. Further, application programs depend upon the DFB for all local or remote information exchange. Applications do not transmit information to other applications or factbases; rather, information is entered into the local factbase and distributed to other hosts only when a distribution rule is satisfied. Direct DFB queries and triggers constitute the only information gathering methods available to application programs. This is a radical change from the typical way of handling information, but allows a defined set of distribution rules to regulate information exchanges. However, if the users wants to send a free-text character string, this can be easily facilitated (although discouraged). Hopefully, combat developers (military scientists) will eventually have data abstractions for all such situations so that free-text messages are rarely, if ever, required.

Application programs have two main functions. to manipulate facts (i.e., create, update and kill facts) and to convert facts into a form understandable by the user (often in a graphical form). There are four applications programs planned in the BRL IDS effort to support the preparation, maintenance, and dissemination of the information associated with a fire support plan in a maneuver operations order (OPORD). The creation of an OPORD is an excellent vehicle to demonstrate the system's capabilities in a dynamic, real-time environment.

One primary purpose of application programs is to serve as an interface between the user and a DFB. Through applications, the user may display or enter DFB facts. Since facts are data abstractions of military concepts in their "purest" form, they must be converted into a form appropriate for human assimilation. The implementation experience of the BRL suggests that any difficulty in displaying a fact correlates directly to how "correctly" the data abstractions

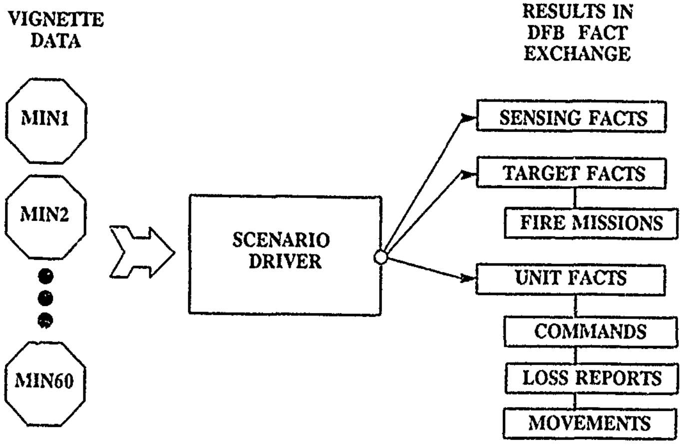


FIGURE 4. Scenario Driver Fact Type Conversion

of the military concepts are represented. That is, if the data abstractions are correct, then it is easy to develop ways to display that information, and vice versa. Many commonly used terms and phrases, such as "target" and "border", have more subtle (and often simpler) meanings than spontaneous intuition affords. The current set of fact types are constantly being scrutinized and revised based on implementation experience; fortunately, they are getting simpler and more general.

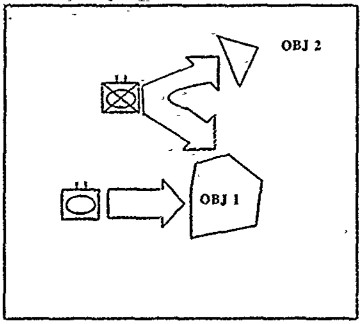
In order to accurately portray real-time combat situations, a *Scenario Driver* program was developed (see **Figure 4**). The scenario driver converts tactical input data into facts and enters them into a separate factbase for dissemination to the command and control nodes being exercised. Thus, in light of the experiment currently under consideration, the scenario driver will serve as many non-exercised (simulated) units in a typical US brigade. The tactical inputs for the scenario driver were developed under contract and provide an unclassified, 60 minute, Fulda Gap battle between a friendly mechanized infantry brigade task force and an enemy tank division with one minute resolution down to the platoon level. From this master event list (i.e., the "scenario"), events are selected to build a "vignette", that is, a particular blue perception of the battle; many vignettes can be built from the master list. During an exercise, the traffic exchanged between five nodes (factbases) will be collected as well as data on the information exchanged between the application programs and the factbases. This data can be analyzed to study and evaluate the efficiency and worth of the factbase concepts, the FEP features (e.g., overhearing), and the appropriateness of the data abstractions. In addition, subjective information will be obtained from the users concerning their ability to use the applications and to understand the situation that was presented.

The Organization Facts (ORGCHART) application displays unit echelon diagrams and Tables of Organization and Equipment (TO&E) This application provides the ability to modify & observe *unit* and *organization* fact types. The organization of any unit within the battlefield command structure may be examined via graphics output. *Commanders may reassigned* their subordinate units using the interactive work panel supported in ORGCHART to build a task organization (changes in unit fact attached links). The current status of unit vehicles, equipment, and ammunition is provided through a "roll-up" procedure. A "roll-up" is the compilation of all significant items assigned to subordinate units making up a unit under examination. A tabular listing is also provided that compares the current status with original unit conditions.

The *Working Map* (WMAP) application displays and manipulates geographical information detailing the locations of friendly and enemy unit, sensing, line (fire control measures), border, and target facts are presented to the user. *All graphical output can be superimposed onto a topographic map background representing* the Fulda Gap in Germany. The WMAP interactive panels allows a user to create geographical line, border and target facts and to modify unit location. Further, special features, such as a the ability to generate a range fan for any weapon within a unit, are offered. Finally, communication *of proposed unit actions and movement may be* sent between independent IDS nodes. This "conversational graphics" capability is provided through the use of the "what_if" meta-fact types (as previously described) that refer to existing facts for the discussing of potential or required *unit location change. It is hoped that the* communication provided by WMAP will avert duplications of effort in the planning of future coordinated unit actions.

The *Fire Plan* application supports the display and creation of dynamic unit operations orders using the *mission* fact type that contains much of the information normally associated with an operations order (see **Figure 5**) Currently, Fire Plan is being developed to enable. During the development of the mission fact definition, many postulated mission concepts were expressed in terms of already existent fact definitions. For example, a unit mission objective can be expressed as a line fact (or a series of line facts)

WORKING MAP



ORGANIZATIONAL CHART

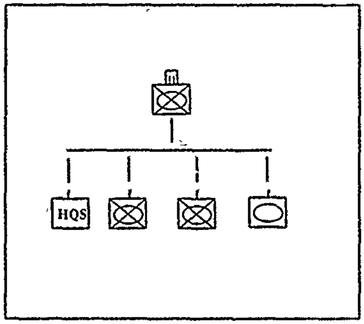


FIGURE 5. Fire Plan Application Uses Existing Applications

on a map while the mission task organization could be compiled as a list of unit facts. As a result of the developed mission fact, OPORD generation and presentation by Fire Plan became dependent upon both the ORGCHART and Working Map applications. In order to facilitate the compilation of included information within the scope of a unit mission, the development of an inter-application communication system became necessary. A new meta-fact type, called a *commo* fact, is used to exchange fact IDs between application programs. Much of the information required to build and display an operations order is created and manipulated using the ORGCHART and WORKING MAP applications in conjunction with commo facts. However, an interactive Fire Plan user input panel is included to control this input or display and record items such as the mission starting and ending times. Thus, the Fire Plan application provides a common ground for the association of otherwise disjoint information generated by both other applications and itself during the creation of an OPORD.

Standard TCP/IP sockets (standard DoD protocols) serve as an interface between applications and the factbase. This allows the applications to reside either on the same host as the factbase or on a separate processor connected by a reasonably reliable data link medium (e.g., a LAN). Although inter-factbase information exchange compensates for unreliable, low-bandwidth data links using the FEP, application to factbase information exchange is designed to operate most efficiently over reliable, high speed links (e.g., a dispersed command post using a LAN). To facilitate this capability, the "package protocol" utility was developed to handle most of the work in establishing the factbase interface (TCP/IP socket) for application program developers. The "package protocol" is available to any machine supporting the standard TCP/IP protocols and a "C" programming language compiler.

All applications make extensive use of of the Sun Microsystem workstation's graphics capability for both display and input These application programs will be used together to demonstrate and evaluate the aforementioned DFB features and to assist the soldier by identifying incoming information and alerting an operator, extracting current situation information from the factbase, graphically depicting unit mission and situation information, insuring that appropriate information is in the fact base, controlling the dissemination of fire support plan information, updating the prescribed CAPs, and making maximum use of graphics "tools" and other software available from the various SWS packages.

CONCLUSION

Through the careful consideration of both military and computer science concepts, a canonical list of fact primitives is being derived. While the list of primitive fact types has changed throughout the implementation of this project, it is hoped that every type of military concept will eventually be represented by a data abstraction rather than text.

The goal of this project is to develop an experimental system that is responsive so that the user will still be prefer it during degraded modes of operation. At the fighting echelons of the brigade and below, information distribution is limited by the low bandwidth communications systems required for highly variable terrain (e.g., non-line-of-sight conditions). Hopefully, new information distribution technology concepts such as the distributed factbase, capability profiles, the features of the Fact Exchange Protocol (overhearing, and listening silence), and the application programs when combined with carefully developed data abstractions of military concepts will provide the capability to operate in spite of severely limited communications. If not, the equipment will be thrown aside during crisis situations and commanders and their staffs will continue to huddle in circles making figure drawings in the dirt when the battle begins.

OBJECT ORIENTED TACTICAL GRAPHIC DISPLAY FOR ARMY COMMAND AND CONTROL

Harlan H. Black

US Army Communications/Electronics Command
(201) 363-2127

ABSTRACT

This paper describes graphics research that was performed to support and manage the tactical display of an AI based prototype Command and Control (C2) decision aid for the staff of an Army corps commander. Graphical data representation, requirements, common access techniques, screen control, and future areas of research are discussed.

I THE RESEARCH DOMAIN

A. CORPS MANEUVER CONTROL PLANNING

The US Army Communications-Electronics Command at Fort Monmouth, New Jersey, has been performing exploratory research to apply artificial intelligence technology to the problem of maneuver control planning for a corps commander. One project, called ARES, consisted of a group of coordinated research efforts in object-oriented tactical graphics, man-machine interface, terrain reasoning, planning, plan recognition, knowledge acquisition, and representation.

B. THE DEVELOPMENT ENVIRONMENT

An experimental ARES test-bed was constructed, consisting of a network of Symbolics™ Lisp machines. This provided a state-of-the-art AI development environment in which the capabilities of an object-oriented approach could be explored for tactical graphics and decision aiding. In this environment, an icon on the screen was able to be directly linked to a database object in the Lisp world, having access to its graphical and reasoning attributes, as well as its functionality, via message passing.

C. THE MAN-MACHINE INTERFACE

To the user, the prototype system was an intelligent plan editor. It monitored his inputs during plan development and provided critiques. It was designed to support planning, not to do the planning.

The prototype's man-machine interface provided the following functionalities:

- It brought system planning capabilities to the user.

- It showed the state of the planning system and database to the user.

- It allowed the user to provide textual and graphical input.

- It permited the user to asynchronously modify the situation, goals, and resources present in the various knowledge bases.

- It presented a computer mediated planning environment as close as possible to that in which con entional, manual planning activities are carried out.

Additional interface functionality, not yet implemented, will allow the user to control the display of information and graphics on the tactical displays.

The prototype was configured to use two display monitors. A monochrome screen displayed a command menu and four plan-editing windows, for textual input. Each window type was customized for a particular planning function. The user was able to use the command menu to select a particular type of window for display. The second monitor was a color graphical display of the battlefield background, overlaid with symbology.

D. THE PROCESS MODEL

With regard to machine reasoning, the maneuver control planning problem was seen to be best expressed in terms of a collection of asynchronous cooperating processes. The user himself was considered a process. The processes were to perform different planning tasks and communicate with each other directly through message passing and indirectly through one or more shared knowledge bases. They needed to work in parallel, just like the corps command staff. Each type of editing window on the monochrome display was associated with a unique reasoning processes and provided the user interface to it.

For the reasoning subsystem, user control was causal. Reasoning was data driven by modifications to the tactical database. Plans were evaluated as new information arrived or as old information changed, and other processes were invoked or spawned to evaluate plan consistency.

II AREAS OF GRAPHICS RESEARCH

A. OUTPUT AND REPRESENTATION

The object-oriented environment provided a powerful development tool. Critical display parameters were able to be stored in the database and referred to, instead of being passed as parameters. This greatly facilitated debugging, as these attribute values were easily accessible. There was, however, a perceptable trade-off in performance, as the support environment heavily relied upon memory paging. While ideal for development, it appeared that a conventional or a smaller, customized run-time object-oriented environment would be better suited for the rapid graphical needs of the battlefield.

The prototype's color graphical display required the overlay of tactical symbology on a map background. Utilizing the object-oriented Flavors™ environment, relational-type record structures and functionalities of icons were classified and heirarcially defined.

After studying standard Army symbology, a distinction was made between icon shape and location. While many icons were best drawn as bit-map images, the location of all single tactical icons were represented as either a point, a polyline, a simple region, or a compound region. This was used in the heirarchial definition to facilitate the merger of a graphical classification of all icons by tactical functionality together with their classification by shape. The functional-oriented definition enabled a smooth interface between the icon and the reasoning subsytems, as is shown in Figures One and Two.

In Figure One, one scheme of attribute inheritance for friendly unit icons, is depicted. The middle and right top-level classes were used to provide graphically related definition of the class of an artillery unit's icon, pictured on the bottom. The left-most 'parent' class of this unit was used to add reasoning-related and non-graphic attributes and functionality. It was found that the most concise representation resulted when icon attributes were defined at the highest practical level and when the merger with parent classes occurred at the lowest possible level.

Figure Two shows a scheme of functional or behavioral inheritance. The heirarchial approach permitted top-level definition of functionality, common to a large set of icons, to be made once, separating it from low-level functionality, relating to a specific class of icons. Thus, the move-icon function was defined once for all icons. It relied

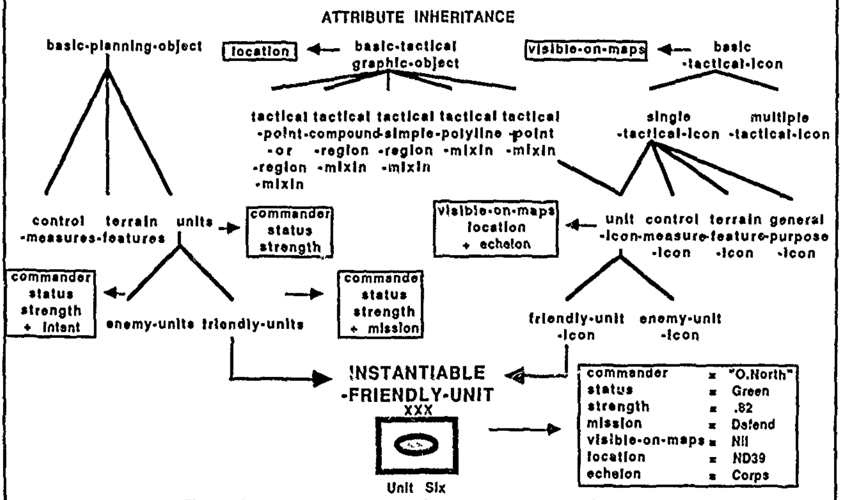


Figure One: Selected Attribute Inheritance for a Tactical Unit

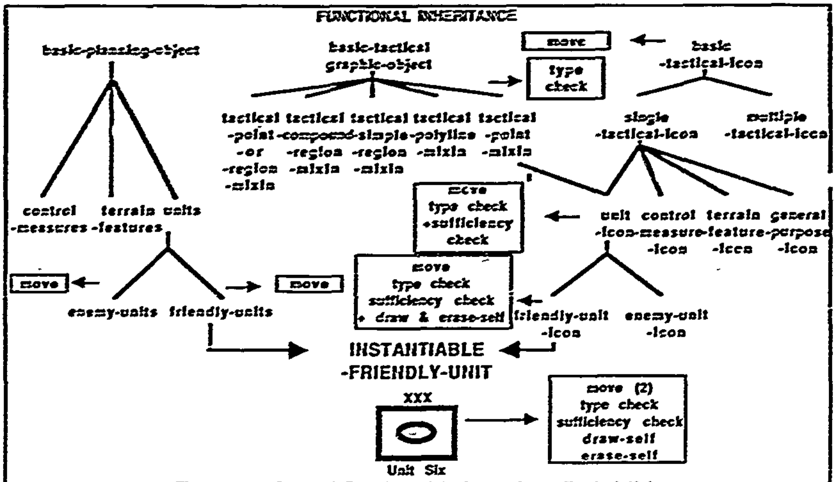


Figure Two: Selected Functional Inheritance for a Tactical Unit

upon erase and re-draw functionalities, defined on an icon class-specific lower level. This provided a rapid and easy method of adding additional icon classes. By merging icon location type at the lowest possible level, an effective type checking mechanism was easily implemented.

B. OVERLAYS AND MULTI-SCREEN CONSISTENCY

Conventional, manual battlefield graphics and interaction was used as the design baseline. In the battlefield, the commander and his staff do their planning on a large paper map collage of the area of interest upon which one or more layers of clear acetate, or overlays, are fastened. Some overlays are prepared beforehand by support staff to hilight areas and features of interest. Others consist of icons, sketched by the planners as they consider alternatives. To provide this functionality in the prototype, a virtual overlay, a Lisp object was designed. It provided a grouping mechanism for associated icon objects for collective operations and was also used for attribute and value inheritance. As pictured in Figure Three, an icon object was able to be associated with one or more overlay objects-and an overlay object was able to be associated with one or more map objects. Tactical-map Lisp objects provided the link between the overlays and the physical display devices. These associations provided a way of specifying default output devices.

An important icon attribute was visible-on-maps, a list of lists. If no process requested the icon's display, the list was nil. Otherwise, each sub-list consisted of the name of a map display, the name of the color(s) that were used to draw the icon, and the Lisp atom t or nil. The latter was used to designate whether the user requested the icon's erasure, for declutter. Every map name was unique. Virtual overlays, together with the icon attribute visible-on-maps, provided an easy way to manage multi-screen displays. This capability was required to enable the user to simultaneously view an area of interest on different screens, each screen being drawn on a different map scale. With icon attribute and display device specification being separately represented and managed, inter-screen consistency was easily maintained.

This provided a way of determining whether an icon was already visible on the screen. This attribute was used to minimize calls for screen update, improving efficiency.

C. UNIFORM DISPLAY FUNCTIONALITY ACCESS

The project was a group effort, spread out over a year. While the mechanisms for graphical functionality were not immediately needed, the access language for this functionality was required

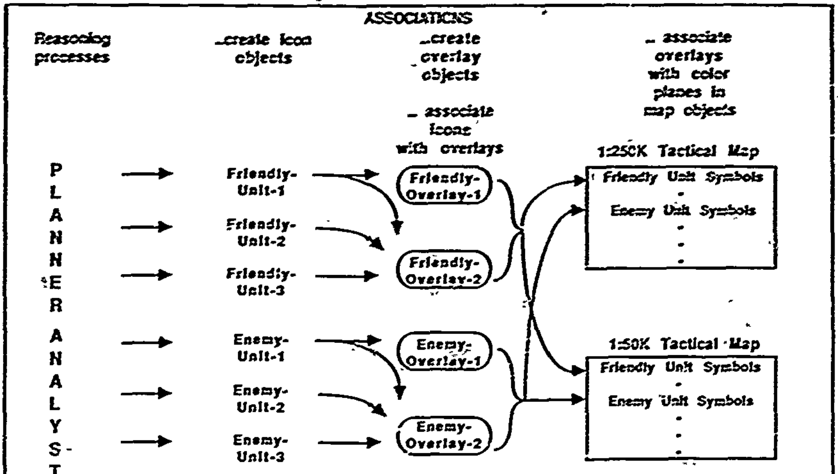


Figure Three: Tactical Object Associations

in the project's early stages. This translated to the definition of the minimal graphical functional requirements for corps maneuver control, and logically expressing them as a language, the Display Access Language.

Five classes of functionality for icons and overlays were identified in the Display Access Language, summarized in Table One. The first two, display and hilighting, were provided for all processes that need to interact with the display. "Screen clutter is a

DISPLAY ACCESS LANGUAGE

Fuctionality	For Tactical Icons	For Tactical Overlays
Display:	Show-Icon Erase-Icon	Show-Overlay-Icons Erase-Overlay-Icons
Highlight:	Highlight-Icon Dehighlight-Icon	Highlight-Overlay-Icons Dehighlight-Overlay-Icons
User Override:	Declutter-Icon Restore-Icon	Declutter-Overlay-Icons Restore-Overlay-Icons
Grouping:	Associate-Icon-With-Overlays Dissociate-Icon-From-Overlays	Associate-Overlay-With-Icons Dissociate-Overlay-From-Icons Clear-Overlay-Of-Icons
Utility:	Move-Icon	Mount-Overlay-Onto-Maps Remove-Overlay-From-Maps

Table One : Display Access Language for Tactical Graphics major concern" [1]. So much so that "for tactical applications, the transition to ADP systems depends, in part, on a viable resolution to the clutter problem" [2]. User-override capabilities were, thus, another vital functional requirement. Overlay grouping, mentioned above, and general utilities added the needed functionality to round out the toolset.

Figure Four provides a detailed sample of the Show-Icon function. Using Zetalisp's optional keyword arguments construct, the user was able to specify arguments in any order. Alternatively, the user could omit one or all arguments and use the default values. The map-alu argument permitted the caller to specify the map that an icon is to be displayed on and the color with which the icon is drawn. If omitted, these crucial argument values were inferred from the overlay object(s) that the icon is associated with, or from overlay(s) that were specified in the function call. The conditional-show argument provided a mechanism whereby an icon's display was able to be posponed, waiting for its associated overlay to become 'mounted' on the tactical map.

Current research at the Communication/Electronics Command is investigating the possibility of designing a common display language across dissimilar display systems to provide a communication channel for graphic screen updates.

D. INTERPROCESS COOPERATION AND DISPLAY CONTROL

On the textual display, declutter was a minimal concern, as there were always four visible windows. When contention arose from conflicting requests by reasoning processes, the user was notified and decided. This was not distractive, as it related to the reasoning itself, and provided valuable insight to the user about how the system was processing or viewing the problem at hand. For the tactical display, however, screen content needed to be kept at a minimum. When a process no longer required a symbol to be seen on the screen, it needed to issue a request for its erasure. This created potential conflicts, when unbeknown to this process, another process was relying on the icon's visibility on the screen. To minimize user overload, an automated mechanism which supports interprocess cooperation and display control was needed.

To provide this control, the contextual data associated with this operation were stored in the icon's visibility-reasons attribute. The reasons contained the map(s) that the icon is visible on, the process(es) that requested the operation, and whether the request was for the icon to be displayed or whether it was for the overlay that the icon was associated with to be displayed. The attribute value was a list of lists. Each sub-list was for a unique tactical map display that the icon was visible on. The sub-lists were of the form (Map-name (Vis-reason) (Vis-reason)...). Map name was the name of a tactical-map object. Vis-reason was of the form (Ob-name Process), where ob-name was the name of either the icon or an overlay object and process was the name of the process that requested the operation. Every Vis-reason for a given map was unique. This structure enabled an icon-related graphic operation on a map to be made and identified for more than one process, and it enabled more than one overlay-related graphical operation by a single process to be made and identified. The latter could be used in cases when a given process had more than one reason for an icon to be seen, in which case it could make an overlay for that reason, associate the overlay and icon with each other, and have the reason utilized for further graphical operations.

With display reason attributes, erasure control for a given tactical map was provided by enforcing a rule that an icon would only be erased if its display reason attribute for that map was nil. This relied on an agreement between processes (and respective system developers) that no process would use another process' name or overlay.

SHOW-ICON

(Show-Icon ICON (&key (map-alu nil) (overlays in-overlays) (conditional-show t) (caller owner)))

Required Arguments : ICON, Name of icon object.

Optional Arguments:

	Data Type	Default
Map-alu	List of two elements. First is the name of a map object. Second is the indicator of icon color.	Nil.
Overlays	List of names of overlay objects.	Overlays that icon is associated with.
Conditional-show	T or nil.	T.
Caller	Name of process.	Name of process that owns/created icon.

Figure Four : Show-Icon function

For example, given a request by process P to erase icon C which is visible on map M, if the vis-reason (C P) was a member of the sub-list for M in the icon's visibility-reasons attribute, then it was removed. If there were no more vis-reasons for M, the sub-list for M in the icon's visible-on-maps was removed, and the icon was then erased from M's display screen. A similar rule was followed for a request to erase an overlay that C was in. Thus, a process could freely call for symbology erasure and not conflict with the display needs of other processes.

D. POTENTIAL ENHANCEMENT AREAS

Several problems arose during the project, relating to AI and computer graphics.

Unit symbols frequently overlap on large scale maps, requiring repositioning and offset indicators. A mechanism for automated repositioning that would be sensitive to optimal location and that would insure that the repositioned icon would not cover another icon was a desirable enhancement for the prototype.

Queries relating to troop movement rely on terrain characteristics. Symbols like the avenue of approach icon, in Figure Five B, are used to outline the appropriate area and movement, drawn as a polygon in figure Five A. Given a digital terrain database, a binary coded two-dimensional array can be constructed to represent the query area with the portions of the area which support troop movement. The capability to efficiently automate the drawing of this icon for any region, irregular or compound, was another desirable enhancement for the prototype.

Finally, although not AI, a compact graphics engine that could manage a large number of bit plane overlays was desireable to demonstrate fielding capability.

III CONCLUSION

This effort provided insight into the applicability of object-oriented tactical graphics. It provided new techniques, notably those imbedded in the Display Access Language, and defined areas for future research.

IV ACKNOWLEDGEMENTS

The author wishes to acknowledge the support and encouragement of Dr. Martin Wolfe and Mr. Edward Beach, of the US Army's Communication/Electronics Command. The author also wishes to acknowledge the contribution of Dr. Norman Badler, University of Pennsylvania, who provided valuable insight to this effort.

V REFERENCES

[1] H. Black, High Technology Initiatives in C3I, Washington DC: AFCEA International Press, 1986, Part IV, p. 249.

[2] J. Chanon, D.Ciccone, and M.Samet, A Framework for the Development of Improved Tactical Symbology, Alexandria VA: US Army Research Institute Technical Report 403, August 1979, p. 4.

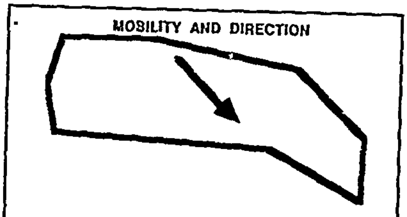


Figure Five A:
Region of Mobility and Desired Direction

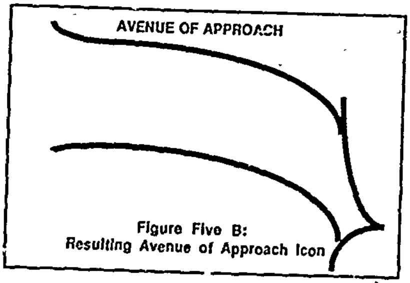


Figure Five B:
Resulting Avenue of Approach Icon

USER INTERFACE MANAGER FOR C3I APPLICATIONS

James B. Nelson

Lockheed Austin Division
O/T0-10, B/30F, 6800 Burleson Road, Austin, Texas, 78744
(512) 44_-5433

ABSTRACT

The construction of user interfaces for computer-based systems is often one of the most troublesome and costly aspects of a system. This is because requirements and functions often change during the design, development, and implementation phases, and because most user interface software is constructed using traditional programming techniques. Thus, each time the user interface must be changed, the software must be redesigned, recoded, and retested. This is very costly to the developer and the user.

These costs have motivated the development of user interface software packages that can be used to prototype user interfaces long before embedding them into a system. Although this is an excellent start, most of these packages fall short of providing a true user interface manager. Most of them provide either a tool for drawing displays or an applications subroutine call, but not both, and very few are portable, extensible, and usable by both programmers and nonprogrammers.

This paper presents the Lockheed User Interface System (LUIS), a user interface manager system that is hardware independent, portable, flexible, extensible, and usable by technical programmers and operational users alike. LUIS provides rapid prototyping of user interfaces that can be moved easily into operational environments.

1. THE PROBLEM

The user interface aspect of a software system is one of the most problematic and frequently one of the most costly components of the system. This is due in part to requirements changes during the typically long development phase and due to the use of traditional programming techniques to produce the man-machine interface.

In the past, most user interfaces have been created by computer programmers using traditional programming methodologies. Thus, when requirements or design changes occur, the software developer must redesign, recode, and retest the interface. Further, if during the review process the new interface is found to need modification, the entire process of design, code, and test must be repeated. This can result in significant costs to the developer and to the user.

Over the past few years, these costs have motivated the development of software packages to provide the software developer, and, yes, even the user, with tools that can be used to create, modify, and evaluate user interfaces long before they are embedded into a system. Although there are several software packages for providing user interface prototyping on the market, most of them are hardware dependent or focus on either the prototyping aspect (allowing a user to draw or create screen displays that do not connect to an application) or on the programmer's subroutine library approach to manage objects on the screen. Few of these packages combine capabilities, that is, creation and management of the display. Even fewer are extensible, portable, provide the capability to specify control relationships at the screen-creation level without software development, or provide a logging feature to assist in identification of problem areas.

2. THE REQUIREMENT FOR A USER INTERFACE MANAGER

Because a flexible, portable, broadly usable, and robust method of constructing user interfaces can significantly enhance software development and reduce costs, a requirement exists to develop a User Interface Manager (UIM). The UIM must be able to do the following

- Provide a basic set of primitives or building blocks from which a wide variety of interfaces may be constructed without restriction to a particular style
- Provide facilities for logging user interactions such as user picks, keystroke entry, and operational events
- Separate user interface management functions from applications functions
- Contain mechanisms for interprocess communications that allow the UIM to manage applications running in separate processes in a distributed environment
- Support extensibility to add new features and new applications
- Support portability to other hardware environments and be able to communicate with other software components

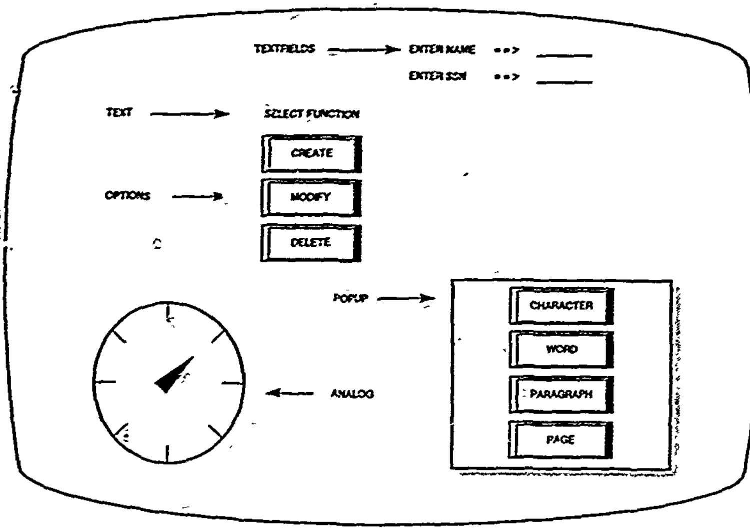


Figure 1. LUIS provides basic objects to allow the designer the freedom to customize user interfaces.

- Provide direct use of applications
- Support a variety of interface styles
- Be usable by both programmers and nonprogrammers.

3. THE APPROACH

Over the past 4 years, the Lockheed Austin Division has been developing a UIM for use in C3I systems: the Lockheed User Interface System (LUIS). LUIS is a software tool used in many applications that provides rapid prototyping capabilities and functions as a mature user interface manager. It currently operates within the UNIX operating environment on both Sun and Silicon Graphics workstations and within the AEGIS operating system on Apollo workstations. It is currently being ported to the ULTRIX operating system on the MicroVax GPS workstation using X Windows and to the UNIX operating system on the Hewlett Packard 330 using X Windows.

The LUIS UIM is based on a dialogue model approach to user interface management. Dialogue models consist of basic user interface objects and operations that are versatile. Dialogues can be created and executed without writing code in traditional programming languages. In a later phase of systems implementation, the LUIS-created prototypes can be integrated with other applications programs. The dialogue approach has enabled the LUIS UIM to construct functional prototypes, modify them, and move them into operational systems, all very easily.

LUIS consists of a set of subsystems that support user interface specification and management. The subsystems facilitate the efficient implementation of user interfaces consisting of a set of basic objects types. These objects provide different visual and functional effects in an interface system, can be combined, and can be tailored to implement a wide variety of interface designs. Both the subsystems and the objects are at a sufficiently high level that the LUIS UIM can be used effectively by both programmers and nonprogrammers.

a. LUIS Objects

The LUIS object structure provides both graphic and textual user interfaces and consists of two object classes, visual and functional. Visual objects provide graphic and textual components panels, popups, options, text, textfields, and analogs. Panels are individual displays in a user interface and consist of some combination of other objects. Popups are windows within a panel in which related objects can be grouped. Options represent alternatives from which the user can select, either within a panel or a popup. Text objects are character strings that are used for the display of textual messages and instructions, and textfields are regions in a popup or a panel where the user can input textual data. Analogs allow a user to specify values in a direct and natural way. Figure 1 illustrates the basic LUIS visual objects.

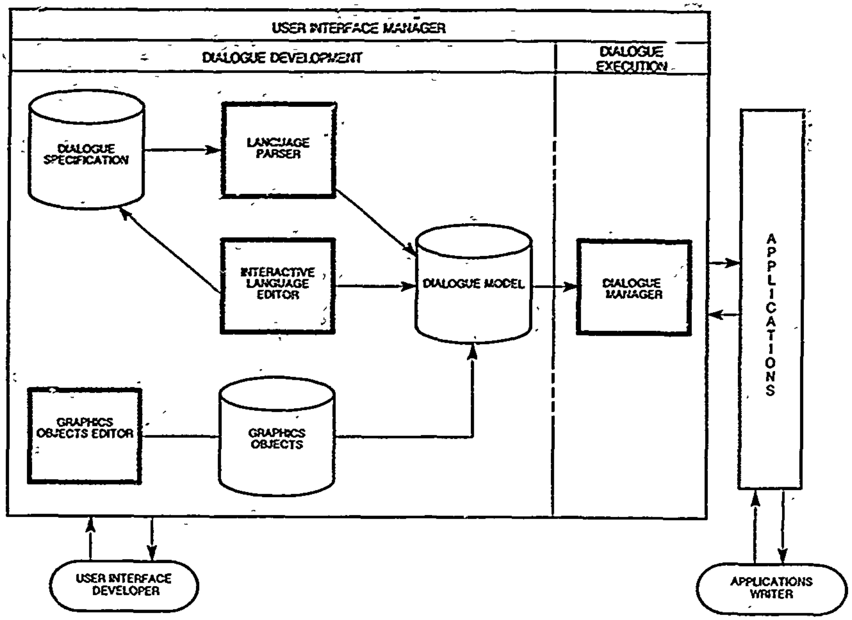


Figure 2. The LUIS subsystems allow rapid development and full-scale management of user interfaces

Functional objects within LUIS control and allow the user to produce the desired effects after the input event. Five basic functional objects exist: implications, states, actions, transfers, and logical sets. Implications are lists of interface operations that are carried out in sequence after a designated event. They are used to define the semantic behavior of LUIS objects. Example implications include: enabling/disabling objects, invoking applications subroutines, moving the cursor to designated points on the screen, and saving user inputs. States enable more conditionality in specifying when implications are executed. These are logical expressions that are used to test what the end-user has selected or entered in the user interface. Different implications will be carried out depending on whether the expression is true or false.

Actions are used to establish calls to applications subroutines from the user interface. They describe characteristics of the subroutine that are necessary for successful invocation. Transfers provide a vehicle for interpanel communications in LUIS, transferring the states of options and textfields in one panel to options and textfields in another panel. Finally, logical sets are used to define mutually exclusive and/or associative relationships for groups of options within a panel.

b. LUIS Subsystems

The subsystems in LUIS are used to construct and execute the basic objects. Subsystems for interface or dialogue specification allow user interfaces to be built without conventional coding. Figure 2 illustrates the LUIS UIM subsystem functional design.

(1) The Language Parser processes object definitions contained in a dialogue specification file. The object definitions are constructed with an English-like authoring language that provides an abstract, device-independent specification of the user interface. The goal of the Parser is to generate a computationally efficient representation of the textual object definitions. This representation is referred to as a dialogue model and consists of binary data structures that are accessed by LUIS to execute the specified user interface.

(2) The Language Editor subsystem is used to specify objects interactively through menus and single keystroke commands. The Language Editor provides an alternative to the Language Parser as a dialogue specification method for developers. It supports direct creation and manipulation of the functional and visible objects in a dialogue model, a what-you-see-is-what-you-get approach. It can be invoked when the user interface is executing to

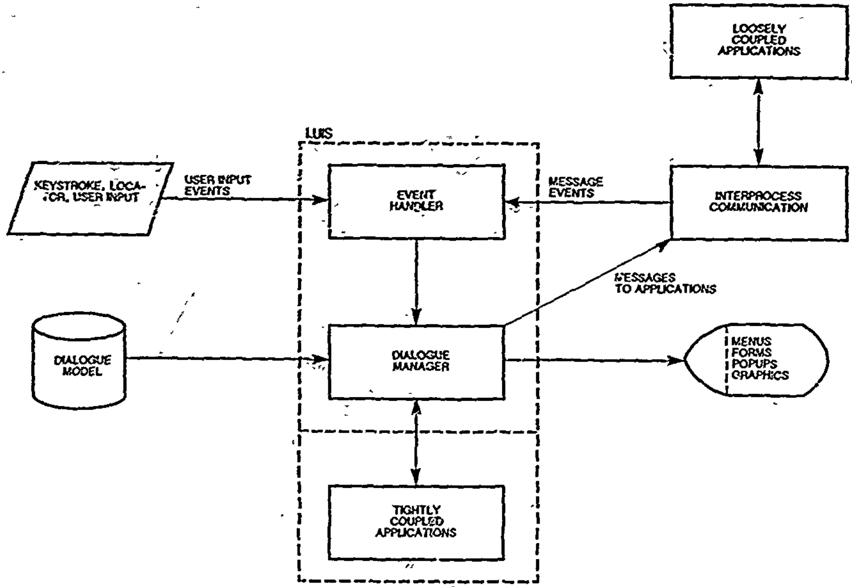


Figure 3. LUIS controls the creation and operation of displays, along with the integration of the user-interface within a system of applications.

enable rapid interface modifications, yet it allows a return to execution mode to enable the developer to assess the changes immediately. The Language Editor produces a dialogue specification file that can be modified and run through the parser subsystem. Thus, the Parser and Editor subsystems are distinct but complementary specification subsystems.

(3) The Graphics Editor supports the interactive creation and modification of special graphic images that may be required in user interfaces. Diagrams, icons, symbols, and analog objects (e.g., dials, gauges, histograms), can be developed with this editor. The Graphics Editor provides a set of primitive shapes from which complex images can be built. It includes facilities for color and font specification, image resizing, and analog indicator and scale definition. Constructed images are saved and then associated with the dialogue model by specifying them in either the Language Editor or dialogue specification file.

(4) The Dialogue Manager subsystem executes the user interface by accessing and interpreting the dialogue model This subsystem presents the visual objects in the model, processes user inputs, invokes application functions, performs action sequences, and responds to requests from applications for dialogue services. The Dialogue Manager can execute the user interface with or without applications. This allows dialogue prototypes to be executed and evaluated before applications are developed After application development, applications can be integrated with the user interface to produce an operational system.

c. LUIS Run-Time Environment

User interfaces that are built with the LUIS UIM can execute alone or as a part of functional applications systems. In both cases, the LUIS UIM Dialogue Manager waits for end-user input from either a locator device, such as a mouse or trackball, or from the keyboard When an event is returned, the Dialogue Manager carries out the operations (implications) that were specified for that event in the dialogue model.

The Dialogue Manager controls the creation and operation of displays and the integration of the user interface within a system of applications Figure 3 illustrates the LUIS UIM Run-Time environment. User selections or inputs can be designated in the user interface design to invoke specified applications routines that carry out specific system functions The Dialogue Manager processes user inputs and invokes the necessary applications routines During execution of the applications routines, the Dialogue Manager assumes the role of dialogue server. The Dialogue Manager passes data to the invoked routines and changes the user interface based on data received from the applications.

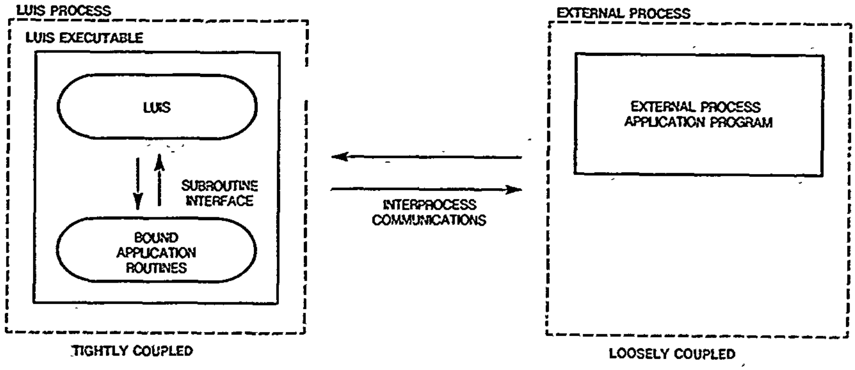


Figure 4. Different architectures for defining interaction with application programs allow developers to use the method best suited to the application.

Applications code can be interfaced through two alternative types of architectures, allowing the developer/user great flexibility when specifying the methods for interfacing applications programs with the user interface. These alternatives are based on the distinction between tightly coupled and loosely coupled applications. Developers can use either architecture or combine the two architectures to integrate applications into the LUIS UIM Figure 4 illustrates the functional use of tightly and loosely bound applications programs.

(1) Tightly coupled applications routines are bound into the LUIS UIM executable image. These routines are linked with the other routines comprising LUIS and execute in the same process as LUIS.

(2) Loosely coupled applications routines provide a true parallelism of operation· LUIS and the external process operate at the same time. A loosely coupled applications program operates in a separate operating system process or window from the LUIS process and executes concurrently with the user interface. Data between LUIS and the applications are sent across the operating system interprocess communication channels Applications use a subroutine interface to send and receive messages across these channels. These messages reflect changes in the user interface, obtain the current status of specified objects in the user interface, and notify LUIS of the application's status.

4. THE RESULTS

The LUIS UIM is being used very effectively in many applications where a UIM is the most efficient, least costly, and fastest method of producing a wide variety of user interfaces for a system The LUIS UIM applications-oriented approach has enabled the rapid-prototyping of user interfaces and the easy migration of those prototypes into operational systems. The LUIS UIM contains all of the significant UIM features.

Extensibility. Steady growth over 4 years into multiple architectures, map graphics control, data base interface and management, and decision-aid applications

Portability. The LUIS UIM is written in C programming language and is based on proven kernels that supply graphics and operating system level functionality Current hardware. Sun, Silicon Graphics, Apollo Soon MicroVax and HP.

Usability. The requirement for ease of use was started in 1984 and has continued. Both programmers and nonprogrammers can use the LUIS UIM to produce displays and to manage the screen Applications programmers and developers find it very quick and easy to use, permitting them to concentrate on the application, not the display.

Performance Evaluation. The LUIS UIM provides logging/timing of user inputs, thereby permitting evaluation of specific user interfaces for particular applications and evaluation of system performance.

Flexibility. The LUIS UIM supports a variety of interface styles (menus, forms, graphical interactions) and input devices (trackball, mouse, touch pane) and provides complete two-way communication between the user interface and applications.

The LUIS UIM has been used to support Government contract work in the following areas Command & Control, Weapons Control, Battle Management, Communications. Intelligence, and Human Engineering Evaluation, Weather Display, and Crisis Management

A Priority Driven Approach

To Real-Time Concurrency Control

Lui Sha[1], Ragunathan Rajkumar[2] and John P. Lehoczky[3]

[1]Department of Computer Science
[2]Department of Electrical and Computer Engineering
[3]Department of Statistics
Carnegie Mellon University

Abstract

The concurrency control of transactions in a real-time database must satisfy not only the consistency constraints of the database but also the timing constraints of individual transactions. In this paper, we focus on the issues in the integration of a locking concurrency control protocol and a priority driven real-time scheduling protocol. We present a real-time concurrency control protocol called the *rw_priority ceiling protocol* which not only prevents mutual deadlocks but also tightly bounds the duration of blocking due to concurrency control in a centralized database environment. In addition, we investigate the application of this protocol in a distributed environment.[a]

1. Introduction

1.1. Motivation

Real-time databases are used in a wide range of applications such as aircraft tracking and the monitoring and control of modern manufacturing facilities. In a real-time database context, concurrency control protocols must not only maintain the consistency constraints of the database but also satisfy the timing requirements of the transactions accessing the database. In standard database applications, such as banking, one can lock data objects and prevent other transactions from accessing them in order to maintain consistency. In a real-time application such as tracking, the consistency is still important, but it is also critical that the values of the data objects are timely. If an object being tracked is moving fast, very stringent timing requirements will be placed on transactions that update the object location. Failure to meet these timing requirements will render the tracking exercise a failure, because the data will be out of date.

To satisfy both the consistency and real-time constraints, there is the need to integrate concurrency control protocol with real-time scheduling protocols. In this paper, we focus on the integration of a locking concurrency protocol with a static priority real-time scheduling protocol, since both these two types of protocols are widely used in practice. A major source of problems in integrating the two protocols is the lack of coordination in the development of concurrency control protocols and of real-time scheduling protocols. In the design of many real-time scheduling protocols, a commonly used assumption is that tasks are independent. Owing to the effect of blocking due to concurrency control, a direct application of a real-time scheduling algorithm to tasks (transactions) may result in a condition known as unbounded *priority inversion*, where a higher priority task is blocked by lower priority tasks for an indefinite period of time. Consequently, the management of task priorities during task synchronization is a major issue in our investigation. On the other hand, in the design of concurrency-control protocols, it is often assumed that the more concurrency offered by the protocol, the better is the performance. For example, the two-phase locking protocol[1] with read and write semantics is considered to be superior than two phase lock protocol using exclusive locks. However, as we will see in Section 2.2, a direct application of the read and write semantic can actually lead to poorer

---

[a]This work was sponsored in part by the Office of Naval Research under contr N00014-84-K-0734, in part by Naval Ocean Systems Center under contract N66001-87-C-0155, and in part by the Federal Systems Division of IBM Corporation under University Agreement YA-278067.

schedulability[b]. The sharing of a read-lock can improve real-time performance only if certain conditions regarding the priorities of readers and writers are satisfied. In this paper, we will address these issues arising from the integration of a concurrency control protocol with a real-time scheduling protocol. However, before we address the integration issues, we give a brief review of the concurrency control protocol and the real-time scheduling protocol that we aim to integrate.

1.2. Related Work

Both concurrency control[2,3,4,5,6,7,8,9,10,11,12] and real-time scheduling algorithms[13,14,15,16,17,18,19,20] are active areas of research in their own right. It is beyond the scope of this paper to examine the possible combinations of concurrency control and real-time scheduling protocols. Rather, we limit ourselves to the integration of two particular protocols: the rate monotonic scheduling algorithm[17] and the modular concurrency control protocol[21]. The rate monotonic scheduling algorithm is a simple static priority scheduling algorithm for periodic tasks which assigns higher priorities to tasks with shorter periods. It was shown in[17] that this is an optimal static priority scheduling algorithm. This means that any task set which can be scheduled by any static priority scheduling algorithm can be scheduled by the rate monotonic scheduling algorithm. The rate monotonic scheduling algorithm has been extended to address a wide range of practical problems, e.g., scheduling of aperiodic and periodic tasks[22], scheduling of communication media[13], the handling of transient processor overloads[23] and the use of binary semaphores for mutual exclusion[24].

The concurrency control protocol examined in this paper is the modular concurrency control protocol[21]. This protocol offers reduced blocking time through the decomposition of the database and the transactions. The decomposition approach used in[21] has two aspects. First, data objects in the database are decomposed into disjoint sets called *atomic data sets* (ADS). The consistency of each atomic data set can be maintained independent of the other atomic data sets. The union of atomic data sets is still an atomic data set, and the conjunction of the consistency constraints of all the atomic data sets is assumed to be equivalent to the consistency constraints of the database. Secondly, each transaction is decomposed into a partially ordered set of *elementary transactions* and *tasks*. The entire transaction is called a *compound transaction*. The model of a compound transaction in this theory allows us to address the real-time scheduling of a mixture of non-database operations, *elementary tasks*, and database-accessing activities, *elementary transactions*, in a single task.[c] Each of the elementary transactions in a compound transaction maintains the consistency of the accessed atomic data set when executing alone on the set. In addition, the post-condition of the compound transaction is assumed to be equivalent to the conjunction of the post-conditions of the elementary tasks and transactions of a compound transaction in any execution path. Since compound transaction is a model of our tasks, we will use the terms "tasks" and "compound transactions" interchangeably in the following discussions.

Having decomposed the database and the transactions, we use the setwise two phase lock protocol[d] to make sure that elementary transactions are run serializably with respect to each of the atomic data sets. It was shown in[21] that the resulting schedules form a superset of sets of serializable schedules. In addition, these schedules possess the properties of consistency, correctness and modularity[21]. Compound transactions and atomic data sets provide us with a useful model for real-time database applications such as tracking, and we illustrate their applications in the following example.

Suppose that an airplane is being tracked by two radar stations, and the collection of data objects, O_1 and O_2, represent the local views of these two stations. These data objects might include the current location, velocity, and identification of the airplane as seen by the particular station. Each of these two data objects forms an atomic data set. This is because the consistency constraints associated with

[b]Schedulability is the level of processor utilization at or below which the deadlines of a set of periodic tasks can always be met.

[c]Although the original theory[21] did not address non-database operations, non-database operations such as data-processing tasks can be viewed as a special elementary transaction which locks some dummy data object that no other transaction will either read or write. Hence, no special treatment is needed from a concurrency control point of view. However, from a real-time scheduling point of view, it is important to distinguish the data processing and the database operations, because an elementary task will always be preempted by the execution of higher priority tasks, but an elementary transaction can block the execution of higher priority ones. The blocking of high priority tasks, as we will see, has important implications to the schedulability of a real-time system.

[d]Under this protocol, a transaction cannot release any lock on any atomic data set until it has obtained all the locks on that atomic data set. Once it has released a lock on any atomic data set, it cannot obtain a new lock on that atomic data set. The transaction can, however, obtain new locks on other atomic data sets.

each track can be checked and validated locally. Each new scan creates a new version of the data object, and in the course of time, the values of O_1 and O_2 form two correlated multivariate time series. The correlation can be used to create a new atomic data set consisting of the global track, represented by data object O_3. The global ADS, i.e., the union $G = \{O_1, O_2, O_3\}$, represents the global and local views of the airplane being tracked.

The notion of atomic data sets is especially useful for tracking multiple targets. Before the formation of the global atomic data sets, we need to correlate all the local tracks near each other to find out which tracks are associated with which targets. Once a global ADS associated with a particular target is formed, the information from the global track can be referenced as a global context to aid the local operations. In addition, the knowledge of a set of local tracks belonging to the same global ADS helps to reduce the number of correlation operations. When data from new scans are used to update the local tracks of a global ADS, we will correlate them first. If the correlation is successful, then the hypothesis that all the ADS's in G are tracking the same target is considered to be valid and there is no need to further correlate the data with the tracks from other ADS's. This may substantially reduce the computation time required in tracking.[e] Conceptually, the birth (the creation of a global track), growth (adding/deleting local ADS's in G) and death (the global track is invalidated by new discoveries) of global ADS's model the dynamics of a tracking database.

In real-time tracking applications, an instance of a periodic task consists of both non-database operations and database operations. In fact, the non-database operations such as signal processing is often more time-consuming than reading and writing the tracks in the database. The structure of a compound transaction provides a useful model for such tasks. To illustrate the syntax of a compound transaction, consider a simple distributed tracking database consisting of a global database GD and a set of local databases $LD_1, ..., LD_n$. Each local database is associated with a particular sensor, and the data from the sensor is processed to create local tracks. Compound transaction Global_View reads the local tracks of a global ADS, correlates them and then updates the global track if the correlation is successful. The following is the pseudo-code of the transaction *Global_View*.

```
Compound_Transaction Global_View;
AtomicVariable obj_new_vector: Track_Vector;
BeginSerial
    BeginParallel
        Elementary_Transaction Read_Local_ADS_1
        BeginSerial
            Lock Lock_Track_1;
            obj_new_vector := Local_Track_1;
            Commit and Unlock Local_Track_1
        EndSerial;

        ...

        Elementary_Transaction Read_Local_ADS_n
        BeginSerial

            ...

        EndSerial

    EndParallel;

    Elementary_Task
    BeginSerial
     Correlate data in obj_new_vector;
    EndSerial;

    If correlation_successful Then
    Begin
      Elementary_Transaction Global_View
      BeginSerial
        Lock  Global_Track;
        Update Global_Track;
        Commit and Unlock Global_Track;
      EndSerial;
    End;
    Else Begin
       Correlate the data with nearby
                tracks from other ADS's
    End;
EndSerial;
```

This example illustrates the following characteristic of our approach. The database is decomposed, and the compound transaction models a generalized task which has both database and non-database operations. In addition, elementary transactions are executed serializably with respect to the atomic data sets.[f] Having reviewed the protocols that we want to integrate, we now outline the organization of the rest of the paper. In Section 2.1, we examine the priority inversion problem that must be solved in the development of a real-time concurrency control protocol. In Section 2.2, we develop the *read-write (rw) priority ceiling* protocol for real-time concurrency control in the context of a centralized

[e] It is important to emphasize that while the ADS's in G are linked by correlation, the degree of correlation is not a consistency constraint. Consistency constraints are relationships between data objects that must be maintained by all the transactions. On the other hand, transactions do not have any responsibility to maintain the correlation between the ADS's in G. In fact, if the initial correlation is due to some signal processing error, then the global ADS will become invalid and must be eliminated, the sooner the better.

[f] The atomic variables are global variables of a compound transaction and are shared by the elementary transactions and tasks.

database. In Section 2.3, we address the real-time concurrency control problem in a distributed environment. Finally, Section 3 presents the conclusion.

2. Real-Time Concurrency Control Issues

2.1. The Priority Inversion Problem

The major problem in integrating a locking protocol with a priority driven scheduling protocol is the problem known as *priority inversion*, which is said to occur when a high priority task is blocked by lower priority tasks. Priority inversion is inevitable in transaction systems. However, to achieve a high degree of schedulability in real-time applications, we must overcome the problem of uncontrolled priority inversion, where the priority inversion occurs over an indefinite period of time. This is illustrated by the following example.

Example 1: Suppose τ_1, τ_2 and τ_3 are three tasks arranged in descending order of priority with τ_1 having the highest priority. Assume that elementary transactions T_1 of task τ_1 and T_3 of τ_3 access the same data object O. Suppose that at time t_1 transaction T_3 obtains a write-lock on O. During the execution of T_3, the high priority task τ_1 arrives, preempts T_3 and later attempts to execute T_1 to access the object O. Task τ_1 will be blocked, since O is already locked. We would expect that τ_1, being the highest priority task, will be blocked no longer than the time for transaction T_3 to complete and unlock O. However, the duration of blocking may, in fact, be unpredictable. This is because transaction T_3 can be preempted by the intermediate priority task τ_2 that does not need to access O. The blocking of T_3, and hence that of τ_1, will continue until τ_2 and any other pending intermediate priority level tasks are completed.

The blocking duration in Example 1 can be arbitrarily long. This situation can be partially remedied if transactions are not allowed to be preempted; however, this solution is only appropriate for very short transactions, because it creates unnecessary blocking. For instance, once a long low priority transaction starts execution, a high priority transaction not requiring access to the same set of data objects may be needlessly blocked[8]. An objective of this paper is to design an appropriate priority management protocol for a given concurrency control protocol so that deadlocks can be avoided and the duration of blocking is tightly bounded.

The use of *priority inheritance* is one approach to bound the arbitrary delays caused by a locking protocol. The basic idea of priority inheritance is that when a task τ's transaction T blocks higher priority tasks, it executes its elementary transaction at the highest priority of all the transactions blocked by T. To illustrate this idea, let us apply this protocol to Example 1. Suppose that task τ_1's transaction T_1 is blocked by task τ_3's transaction T_3. The priority inheritance protocol requires that transaction T_3 execute its transaction at task τ_1's priority. As a result, task τ_2 will be unable to preempt transaction T_3 and will itself be blocked. In other words, the higher priority task τ_2 must wait for the elementary transaction of lower priority task τ_3 to be executed because transaction T_3 "inherits" the priority of task τ_1. Otherwise, T_1 may be forced to wait for task τ_2 to complete. When task τ_3's elementary transaction T_3 completes, τ_3 returns to its assigned lowest priority and awakens transaction T_1 waiting for the lock on shared data object O. T_1, having the highest priority, immediately preempts τ_3 and runs to completion. This enables τ_2 and τ_3 to resume in succession and run to completion.

As we can see, this simple priority inheritance idea reduces the blocking time of a higher priority task from the entire execution times of lower priority tasks to only the duration of lower priority tasks' elementary transactions. However, this simple idea is inadequate for two reasons. First, the problem of deadlock has not been solved. Second, the blocking duration for a task, though bounded, can still be substantial, because a *chain* of blocking can be formed. For instance, suppose that task τ_1 needs to sequentially access objects O_1 and O_2. Also suppose that τ_2 preempts τ_3 which has already locked O_2. Then, τ_2 locks object O_1. Transaction τ_1 arrives at this instant and finds that the objects O_1 and O_2 have been respectively locked by the lower priority transactions τ_2 and τ_3. As a result, τ_1 would be blocked for the duration of two elementary transactions, once to wait for τ_2 to release O_1 and again to wait for τ_3 to release O_2. Thus, a chain of blocking can be formed.

The above two reasons motivate us to develop the *rw priority ceiling protocol*, which not only minimizes the blocking time of a task τ to the duration of at most one elementary transaction but also prevents the formation of deadlocks. The under-

[8]The priority inversion problem was first discussed by Lampson and Redall[25] in the context of monitors. They suggest that each monitor be always executed at a priority level higher than all tasks that would ever call the monitor.

lying idea of this protocol is to ensure that when a transaction T preempts another transaction, the priority at which this new transaction will execute must be guaranteed to be strictly higher than the priorities of all the preempted transactions, taking the priority inheritance protocol into consideration. If this condition cannot be satisfied, transaction T is suspended, and the transaction that blocks T inherits T's priority. Example 2 illustrates this idea and the deadlock avoidance property of this protocol, while Example 3 illustrates the avoidance of chained blocking. For simplicity of illustration, we assume exclusive locks in these two examples.

<u>Example 2:</u> Suppose that we have three tasks τ_0, τ_1 and τ_2 arranged in descending order of priority. In addition, there are two data objects O_1 and O_2 belonging to the same ADS A. We define the *priority ceiling* of a data object as the priority of the highest priority transaction that may lock this object. The priority ceiling of an object O represents the highest priority task that a transaction that has locked only O can inherit. Suppose the sequences of processing steps for the transactions embedded in the three tasks are as follows:

T_0 = {···, *Lock*(O_0), ···, *Unlock*(O_0), ···}

T_1 = {···, *Lock*(O_1), ···, *Lock*(O_2), ···, *Unlock*(O_2), ···, *Unlock*(O_1), ···}

T_2 = {···, *Lock*(O_2), ···, *Lock*(O_1), ···, *Unlock*(O_1), ···, *Unlock*(O_2), ···}

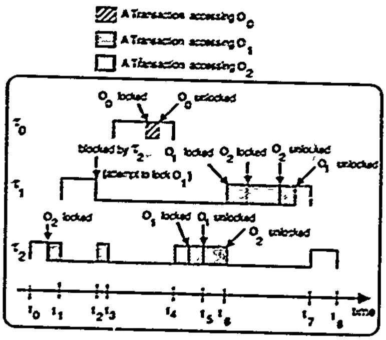


Figure 2-1: Sequence of Events described in Example 2.

The sequence of events described below is depicted in Figure 2-2. A line at a low level indicates that the corresponding task is blocked or has been preempted by a higher priority task. A line raised to a higher level indicates that the task is executing. The absence of a line indicates that the task has not yet arrived or has completed. Shaded portions indicate execution of transactions.

Recall that the priority of transaction T_1 is assumed to be higher than that of transaction T_2. Thus, the priority ceilings of both objects O_1 and O_2 are equal to the priority of transaction T_1. Suppose that at time t_1, transaction T_2 has executed Lock(O_2). At this instant, task τ_1 is initiated and preempts transaction T_2. However, when task τ_1 tries to enter its elementary transaction at time t_2 by making an indivisible system call to execute Lock(O_1), the scheduler will find that task τ_1's priority is *not* higher than the priority ceiling of *locked* data object O_2. Hence, the scheduler suspends transaction τ_1 without locking O_1. Note that τ_1 is blocked outside its elementary transaction. Transaction T_2 now *inherits* the priority of task τ_1 and resumes execution. Since τ_1 is denied the lock on O_1 and suspended instead, a potential deadlock between T_1 and T_2 is prevented. If τ_1 were granted the lock on O_1, then τ_1 would later wait for τ_2 to release the lock on O_2, while τ_2 would wait for τ_1 to release the lock on O_1.

On the other hand, suppose that at time t_3, while T_2 is still in its transaction, the highest priority task τ_0 arrives and attempts to lock data object O_0. Since the priority of τ_0 is higher than the priority ceiling of locked data object O_2, task τ_0's transaction T_0 will be granted the lock on the data object O_0. Task τ_0 will therefore continue and execute its transaction, thereby effectively preempting T_2 in its transaction and not encountering any blocking. At time t_4, T_0 completes execution and T_2 is awakened, since T_1 is blocked by T_2. T_2 continues execution and locks O_1. At time t_5, T_2 releases O_1. At time t_6, when T_2 releases O_2, task τ_2 resumes its assigned priority. Now, T_1 is signaled. Having a higher priority, it preempts T_2 and completes execution. Finally, T_2 resumes and completes.

Note that in the above example, τ_0 is never blocked. τ_1 was blocked by the lower priority task τ_2 during the intervals $[t_2, t_3]$ and $[t_4, t_6]$[h]. However, these two

---

[h]The interval $[t_3, t_4]$ is not considered to be a blocking interval for τ_1, since it was only preempted by the higher priority task τ_0.

intervals correspond to the duration that T_2 needs to lock the two data objects in A. Thus, the blocking duration of τ_1 is equal to the duration of a single elementary transaction of a lower priority transaction T_2, even though the actual blocking occurs over disjoint time intervals. It is, indeed, a property of this protocol that any task can be blocked by at most one lower priority elementary transaction until it suspends itself or completes. This property is further illustrated by the following example.

Example 3: Suppose that $A_1 = \{O_1\}$ and $A_2 = \{O_2\}$. Consider the example where a chain of blocking can be formed. We assumed that task τ_1 needs to access data objects O_1 and O_2 sequentially, while τ_2 accesses O_2, and T_3 accesses O_1. Hence, the priority ceilings of data objects O_1 and O_2 are equal to P_1, the priority of τ_1. As before, let τ_3 lock O_1 at time t_0. At time t_1, task τ_2 arrives and preempts τ_3. However, at time t_2, when task τ_2 attempts to enter its transaction T_2 and lock O_2, the run-time system finds that the priority of τ_2 is *not* higher than the priority ceiling P_1 of the *locked* object O_1. Hence, τ_2 is denied the lock on O_2 and is blocked. Transaction T_3 resumes execution at τ_2's priority. At time t_3, when T_3 is still executing, τ_1 arrives and preempts T_3. At time t_4, τ_1 attempts to enter its transaction T_1 to lock O_1 and is blocked by τ_3 which holds the lock on O_1. Hence, τ_3 inherits the priority of τ_1. At time t_5, task τ_3 exits its transaction, resumes its original priority and awakens τ_1. Task τ_1, having the highest priority, preempts τ_3 and runs to completion. Next, τ_2, which is no longer blocked, completes its execution and is followed by τ_3. Again, note that τ_1 is blocked by τ_3 during the intervals $[t_4, t_5]$ which corresponds to the single elementary transaction. Also, task τ_2 is blocked by τ_3 during the disjoint intervals $[t_2, t_3]$ and $[t_4, t_5]$ which also correspond to the duration of task τ_3's elementary transaction.

2.2. The Read-Write Priority Ceiling protocol

Having reviewed the basic concepts of our approach, we now formalize our approach. Before we begin the technical investigation, we first list our assumptions and state the notation used. We assume that we have a set of loosely coupled uni-processors. In each processor, there is a set of statically allocated periodic tasks. A task can execute in parallel on more than one processor. For example, task τ can update an ADS and its replication on processors X and Y in parallel. We assume that each stream of aperiodic tasks, if any, will be converted to periodic activities via a periodic server task. For example, to handle random requests from an operator, we can buffer his input and use a periodic server task to respond to his requests periodically[1]. Since tracking operations consist of both signal processing and database accessing, we assume that each instance of a periodic task is an interleaving of data-processing code and database operations modeled as elementary tasks and elementary transactions of a compound transaction respectively. We shall assume that the database is decomposed into atomic data sets, and the setwise two-phase lock protocol is used by elementary transactions for concurrency control. We assume that the rate-monotonic algorithm is used to assign a priority to each task. This algorithm assigns higher priorities to tasks with shorter periods and is an optimal static priority algorithm for periodic tasks[17]. If two tasks are ready to run on a processor, the higher priority task will run. Equal priority tasks are run in a FCFS order. We also assume that a transaction does not attempt to lock an object that it has already locked and thus deadlock with itself. In addition, we assume that in each processor the runtime system will serialize the execution of syntactically parallel elementary tasks and transactions. For example, in a uni-processor if a compound transaction has the construction { BeginParallel Elementary_Transaction_1; Elementary_Transaction_2; EndParallel ··· }, then either Elementary_Transaction_1 completes before the start of Elementary_transaction_2, or vice versa. These two elementary transactions will, of course, execute in parallel, should they execute on different processors. A task can suspend itself during its execution of non-database operations, e.g. waiting for I/O. However, self suspension is not permitted when it holds locks on database objects. This also implies that we do not permit a transaction to lock across the network. We also assume that either multiple "read" locks or a single "write" lock can be held on a data object.

Notation: We use the notation $\{A_1, \ldots, A_k\}$ to denote the atomic data sets of database D.

Notation: We denote the given tasks as an ordered set $\{\tau_1, \cdots, \tau_n\}$ where the tasks are listed in descending order of priority, with τ_1 having the highest priority.

Notation: We use $T_{i,j}$ to denote an elementary transaction of task τ_i that accesses ADS A_j. We will also use the simplified notation T_i when the identity of the ADS is not important.

[1]For an advanced treatment of aperiodic tasks, readers are referred to[22].

Notation: We use the notation P_i to denote the priority of task τ_i.

Definition: The lock on a data object can either be a read-lock or a write-lock. A task τ that holds a read-lock (write-lock) on a data object O is said to have read-locked (write-locked) object O. The *write priority ceiling* of a data object is defined as the highest priority task that may write this object. The *absolute priority ceiling* is defined as the highest priority task that may read or write this data object. When a data object O is write-locked, the *rw_priority ceiling* of O is defined to be equal to the absolute priority ceiling of O. When a data object O is read-locked, the rw_priority ceiling of O is defined to be equal to the write priority ceiling of O.

Having stated our objectives and our assumptions, we now define the rw_priority ceiling protocol that is used on each processor.

1. Task τ, having the highest priority among the tasks ready to run, is assigned the processor. Before task τ enters an elementary transaction T, it must first obtain the locks on the data objects that it accesses. Suppose that τ attempts to lock data object O. Let O^* be the data object with the highest rw_priority ceiling of all data objects currently locked by transactions other than those of τ. Task τ will be blocked and the lock on an object O will be denied, if the priority of task τ is not higher than the rw_priority ceiling of data object O^*. In this case, task τ is said to be blocked by the task whose transaction holds the lock on O^*. If task τ's priority is higher than the rw_priority ceiling of O^*, then τ is granted the lock on O[1]. When a task τ exits its elementary transaction, the data objects associated with the transaction will be unlocked.

2. A task τ's transaction T uses its assigned priority, unless it is in its transaction and blocks higher priority transactions. If transaction T blocks higher priority tasks, T *inherits* P_H, the highest priority of the tasks blocked by T. When task τ exits its transaction, it resumes its original priority. Priority inheritance is transitive. Finally, the operations of priority inheritance and of the resumption of original priority must be indivisible.

3. When a task τ does not attempt to enter an elementary transaction, it can preempt another task τ_L executing at a lower priority, inherited or assigned.

Remark: The objective of this protocol is to ensure that each elementary transaction is executed at a higher priority level than the priority levels which can be inherited by preempted elementary transactions. When a task τ write-locks a single object O, the rw_priority ceiling of O represents the highest priority that τ can inherit through O. For example, when τ write-locks O, it can block the highest priority task τ_H that may read or write O and hence inherit τ_H's priority. Therefore, the rw_priority ceiling of a write-locked object is defined to be equal to the absolute priority ceiling. Alternatively, let a low priority task τ hold a read-lock on a data object O and let task τ_w be the highest priority task that may request a write lock on O. Task τ can block τ_w and inherit τ_w's priority. Therefore, the rw_priority ceiling of a read-locked data object is defined as the data object's write priority ceiling. A read-locked object O can be, of course, read-locked again by a task τ_r which has priority higher than that of τ_w. However, in this case, task τ_r has *preempted* τ and there is *no* blocking.

Remark: Under this protocol, we need not explicitly check for the possibility of read-write conflicts. For instance, when an object O is write-locked by a task τ, the rw_priority ceiling is equal to the highest priority task that can access O. Hence, the protocol will block a higher priority task that may want to write or read O. On the other hand, suppose that the object O is read-locked by τ. Then, the rw_priority ceiling of O is equal to the highest priority task that may write O. Hence, a task that attempts to write O will have a priority no higher than the rw_priority ceiling and will be blocked. Only the tasks that read O and have priority higher than the rw_priority ceiling will be allowed to read-lock O, and read-locks are compatible.

Under the rw_priority ceiling protocol, mutual deadlock of transactions cannot occur and each task can be blocked by at most one elementary transaction until it completes or suspends itself. We shall now prove both these properties of the rw_priority ceiling protocol.

Lemma 1: Under the rw_priority ceiling

[1]Under this condition, there will be no read-write conflict on the object O, and we need not check if O has been locked. See the second remark that follows the protocol definition.

protocol, each transaction will execute at a higher priority level than the level that the preempted transactions can inherit.

Proof: By the definition of the rw_priority ceiling protocol, when a task τ locks a set of data objects, the highest priority level τ can inherit is equal to the highest rw_priority ceiling of the data objects locked by τ. Hence, when a task τ_H's priority is higher than the highest rw_priority ceiling of the data objects locked by a transaction T of task τ, the transactions of τ_H will execute at a priority that is higher than the preempted transaction T can inherit.

Theorem 2: There is no mutual deadlock under the rw_priority ceiling protocol.

Proof: Suppose that a mutual deadlock can occur. Let the highest priority of all the tasks involved in the deadlock be P. Due to the transitivity of priority inheritance, all the tasks involved in the deadlock will eventually inherit the same highest priority P. This contradicts Lemma 1.

Lemma 3: Under the rw_priority ceiling protocol, until task τ either completes its execution or suspends itself, task τ can be blocked at most once by a single elementary transaction of a lower priority task τ_L, even if τ_L has multiple elementary transactions.

Proof: Suppose that task τ is blocked by a lower priority task τ_L. By Theorem 2, there will be no deadlock and hence task τ_L will exit its current transaction at some instant t_1. Once task τ_L exits its transaction at time t_1, task τ_L is preempted by τ. Since τ_L is no longer within a transaction, it cannot inherit a higher priority than its own priority unless it enters another transaction. However, τ_L cannot resume execution until τ completes or suspends itself. The Lemma follows.

Theorem 4: Under the rw_priority ceiling protocol, a task τ can be blocked by at most a single elementary transaction of one lower priority task until either τ completes its execution or suspends itself.

Proof: Suppose that τ is blocked by n lower priority transactions. By Lemma 3, τ must be blocked by the transactions of n different lower priority tasks, $\tau_1, ..., \tau_n$, where the priority of τ_i is assumed to be higher than or equal to that of τ_{i+1}. Since a lower priority task cannot block a higher priority task unless it is already in its transaction, tasks $\tau_1, ..., \tau_n$ must be in their transactions when τ arrives. By assumption, τ is blocked by τ_n and τ_n inherits the priority of τ. Since τ can be blocked by τ_n, task τ's priority cannot be higher than the highest priority P that can be inherited by τ_n. On the other hand, by lemma 1, task τ_{n-1}'s priority is higher than P. It follows that task τ_{n-1}'s priority is higher than that of task τ. This contradicts the assumption that τ's priority is higher than that of tasks $\tau_1, ..., \tau_n$.

Corollary 5: If a task τ_i suspends itself at most k times, then the above theorem holds with the duration of blocking equal to $k+1$ elementary transactions.

Remark: The rw_priority ceiling protocol is selectively restrictive on the sharing of read-locks. The reason is that a direct application of the read and write semantic can lead to prolonged durations of blocking. For example, suppose that we have a single write transaction at the highest priority level and ten lower priority read transactions. If we let ten transactions concurrently holding read-locks on data object O, then when a higher priority task arrives later and attempts to write O, it has to wait for all ten of these transactions to complete. That is, some forms of concurrency can lengthen the worst-case duration of blocking, resulting in poorer schedulability.

We now develop a set of *sufficient* conditions under which a set of periodic tasks with hard deadlines at the end of the periods can be scheduled by the rate-monotonic algorithm[17] when the rw_priority ceiling protocol is used.[k]

We quote the following theorem due to Liu and Layland which was proved under the assumption of independent tasks, i.e. when there is no blocking due to data sharing and synchronization.

Theorem 6: A set of n periodic tasks scheduled by the rate-monotonic algorithm can always meet their deadlines if

$$\frac{C_1}{T_1}+\cdots+\frac{C_n}{T_n}\le n(2^{1/n}-1)$$

[k]An aperiodic task can be converted to a periodic task by the use of a periodic server task and buffering the arriving aperiodic tasks.

where C_i and T_i are the execution time and period of task τ_i respectively.

Theorem 6 offers a sufficient (worst-case) condition that characterizes the rate-monotonic schedulability of a given periodic task set. An exact characterization of rate-monotonic schedulability can be found in[22].

When tasks are independent of one another and do not access shared data, Theorem 6 provides us with the condition under which a set of n periodic tasks can be scheduled by the rate-monotonic algorithm.[1] Although this theorem takes into account the effect of a task being preempted by higher priority tasks, it does not consider the effect of blocking caused by lower priority tasks upon schedulability analysis. We now consider the effect of blocking.

Theorem 7: A lower priority write transaction T_w can block a higher priority task τ with priority P, if and only if T_w may write-lock a data object whose absolute priority ceiling is higher than or equal to P. A lower priority read transaction T_r can block a higher priority task τ with priority P, if and only if T_r may read-lock a data object whose write priority ceiling is higher than or equal to P.

Proof: It directly follows from the definitions of the rw_priority ceiling protocol.

Let Z be the set of elementary transactions that could block task τ. By Theorem 4, task τ can be blocked for at most the duration of a single element in Z if it does not suspend itself. Hence the worst-case blocking time for τ is the duration of the longest elementary transaction in Z when τ does not suspend itself. If the task τ suspends itself k times, then the worst-case blocking time is equal to the sum of the $k+1$ longest elements in Z. We denote this worst-case blocking time of task τ_i as B_i. Note that given a set of n periodic tasks, $B_n = 0$, since there is no lower priority task to block τ_n.

Theorem 6 can now be generalized in a straightforward fashion. In order to test the schedulability of τ_i, we need to consider both the preemptions caused by higher priority tasks and blocking by lower priority tasks along with τ_i's own utilization. The blocking of any instance of τ_i is bounded by B_i. Thus, Theorem 6 becomes

Theorem 8: Suppose that a task does not suspend itself from initiation to completion. A set of n periodic tasks using the rw_priority ceiling protocol can be scheduled by the rate monotonic algorithm if the following conditions are satisfied:

$$\forall i, 1 \le i \le n, \frac{C_1}{T_1} + \frac{C_2}{T_2} + \cdots + \frac{C_i}{T_i} + \frac{B_i}{T_i} \le i(2^{1/i} - 1)$$

Proof: Suppose that for each task τ_i the equation is satisfied. It follows that the equation of Theorem 6 will also be satisfied with $n = i$ and C_i replaced by $C_i^* = (C_i + B_i)$. That is, in the absence of blocking, any instance of task τ_i will still meet its deadline even if it executes for $(C_i + B_i)$ units of time. It follows that task τ_i, if it executes for only C_i units of time, can be delayed by B_i units of time and still meet its deadline. Hence the theorem follows.

Remark: The first i terms in the above inequality constitute the effect of preemptions from all higher priority tasks and τ_i's own execution time, while B_i of the last term represents the worst case blocking time due to *all* lower priority tasks for one instance of task τ_i.

Corollary 9: A set of n periodic tasks using the rw_priority ceiling protocol can be scheduled by the rate monotonic algorithm if the following condition is satisfied:

$$\frac{C_1}{T_1} + \cdots + \frac{C_n}{T_n} + max\left(\frac{B_1}{T_1}, \cdots, \frac{B_{n-1}}{T_{n-1}}\right) \le n(2^{1/n} - 1)$$

Proof: Since $n(2^{1/n} - 1) \le i(2^{1/i} - 1)$ and $max\left(\frac{B_1}{T_1}, \cdots, \frac{B_{n-1}}{T_{n-1}}\right) \ge \frac{B_i}{T_i}$, if this equation holds then all the equations in Theorem 8 also hold.

2.3. Distributed Database Issues

In this section, we investigate the use of rw_priority ceiling protocol as a basis for real-time concurrency control in a distributed environment. While a locking protocol is still an efficient method to ensure consistency within a node, holding locks across the network is unattractive. Owing to communication delay, locking across the network will only force the local copy to be as outdated as the remote copies and this is counter-productive for real-time applications such as tracking. It is better to have an up-to-date local copy and let the remote copies be histori-

[1] That is, the conditions under which all the instances of all the n tasks will meet their deadlines.

cal versions. That is, we will adopt a multiple version approach for concurrency control for distributed data objects. The priority driven and locking based concurrency control protocol will be used within each node in the the network. In addition, we assume a single writer and multiple readers model for distributed data objects. This is a simple model that effectively models applications such as distributed tracking in which each radar station maintains its view and makes it available to other nodes in the network. The extension to multiple writers is presented at the end of this section.

In a distributed database environment, an atomic data set is a logical unit of data objects for distribution, because it represents a logically consistent view. An elementary transaction operating upon an ADS represents an atomic unit of operation for concurrent execution. For example, a global ADS G can be the union of many local track ADS's and a global track ADS. Each ADS in the union can reside on a different computer. To address the problem of reliability, we can replicate an atomic data set on another processor. It may seem that the replicated ADS and the original ADS are part of a single atomic data set, however, they are *two* atomic data sets if one is allowed to be a historical version of the other[21]. For example, suppose that we have two ADS's residing on two nodes in a network: $A_1 = \{O_1\}$ and its copy $A_2 = \{O_2\}$. If we insist that A_1 and A_2 must be identical with respect to all references, i.e. $O_1 = O_2$, then these data objects will be part of a single ADS. The updates to them must appear to be an instantaneous event with respect to other transactions that access O_1 and O_2. This can be accomplished by using the setwise two-phase locking protocol to perform synchronous updates to O_1 and O_2. However, locking them across the network can lead to long durations of blocking, because of the communication delay. To satisfy the requirement of one being a historical copy of the other, A_1 and A_2 can be modeled as two ADS's and be updated asynchronously. The following is the pseudo-code of the compound transaction Update_Both that maintains a historical relationship between A_1 and A_2. Note that there is no attempt to ensure that other transactions will read identical versions of O_1 and O_2.

```
Compound_Transaction Update_Both;
BeginParallel
  Elementary_Transaction Update_Local_Copy_A1
  BeginSerial
    Lock O1
    O1 = new_value;
    Unlock O1
  EndSerial;

  Elementary_Transaction Update_Remote_Copy_A2
  BeginSerial
    Lock O2
    O2 = new_value;
    Unlock O2
  EndSerial;
EndParallel;
```

While atomic data sets provide us with logically consistent views, the copies of the logically consistent views could be, owing to the delays in the network, temporally inconsistent. That is, some of the views can be out of date. There are applications where a *temporally* consistent view is better than just the latest information that can be obtained at each site. In an application like tracking, a local track would be updated periodically in conjunction with repetitive scans. Hence, in order to provide a temporally consistent view in a distributed environment, we can utilize the periodicity of the writer as a time-stamp mechanism. For example, given an ADS and its replication, if for each data object there is only a single periodic writer and if the deadlines of this single periodic writer can be guaranteed on all the processors, then all these data objects will be updated by the end of the writer's period. That is, on each processor, during period n the versions of period $(n-1)$ are consistent. It is, of course, difficult to observe identical deadlines on local and remote processors because of the network communication delay. Typically, the versions of the data objects at remote sites will lag behind the local site. Thus, the problem of ensuring a temporally consistent view becomes a network-level real-time scheduling problem in which the time lags in the distributed versions are controlled. Once the lags are controlled, distributed tasks can read the proper versions of the distributed data objects and ensure that their decisions are based upon temporally consistent data. When all the tasks in each scheduling element in the network, e.g. processor or communication medium, are scheduled by the rate-monotonic algorithm, the version lag between two replicated data objects in a network is bounded by the number of scheduling elements on its update path.

Lemma 10: Let $A_i(v)$ denote the version v of an ADS residing in processor i. Suppose that ADS A_1 and its replications, $\{A_1, ..., A_n\}$, are distributed in n processors. Assume that a real-time scheduling algorithm guarantees that $max(v_i,v_j) \leq k$, $1 \leq i, j \leq n$. That is, the maximal lag between the versions of these ADS's at different processors is bounded by k. At period n, we have $\{A_1[n-k] = A_2[n-k] = ... = A_m[n-k], n \geq k\}$ available at all the n processors.

Proof: It directly follows from our single writer assumption and our assumption that the maximal lag is k.

Remark: Since we do not have deadlock within each processor and locks are not allowed to be held across processor boundaries, we do not have the problem of distributed deadlocks.

To calculate the version lags, each scheduling element on the path is counted as a node. Thus, if two replicated data objects reside in two processors connected by a communication medium, we consider that there are three nodes on the path: the local processor, the communication medium and the remote processor. For example, suppose that we have a set of processors connected by a communication bus and that all the processors and communication bus are scheduled by the rate monotonic algorithm.[m] Let ADS A_1 and A_2 reside in two different processors and A_1 be at the home site. The rate monotonic algorithm guarantees that A_1 can be updated by the end of the first period. That is, we can initiate the *send* operation no later than the starting time of the second period. The rate monotonic algorithm on the communication medium ensures that the message will be delivered to the receiving processor by the end of the second period. It follows that the "update A_2" request is ready at the initiation time of the third period and can be carried out by the end of the third period.

Theorem 11: When all the tasks in every element of the network are schedulable by the rate-monotonic algorithm, the version lag between two replicated ADS's in a network is bounded by the number of nodes on its update path.

Proof: Suppose that replicated ADS's A_1 and A_2 can be updated within the same period. The version lag is 1, because ADS A_1 can be updated before A_2. But both of them will be updated by the end of the same period. Thus, the lag between the versions of these two ADS's is at most 1. By introducing each additional full period delay between the updates of the two ADS's, the version lag increases by 1. That is, when the updates are separated by n periods, the version lag is at most $(n + 1)$. When there are k nodes on the path, A_1 will be updated by the end of the first period and A_2 will be updated by the end of the k^{th} period. That is, they are separated by at most $(k-1)$ periods. It follows that the version lag between them is at most $(k-1) + 1 = k$.

Corollary 12: When all the tasks in every element of the network are schedulable by the rate-monotonic algorithm, the time lag between the information provided by two replicated ADS's in a network is bounded by the product of the writer period and the number of nodes on its update path.

Remark: This approach can be extended to address the multiple readers and writers problem when a home site of an ADS can be defined. We first require all the writers to write the ADS at the home site by following the setwise two-phase locking protocol with rw_priority ceiling. This ensures the logical consistency of the ADS at the home site. For remote copies of this ADS, we can let the highest frequency writer τ_H to copy the home site ADS versions over to the remote sites. For example, suppose that the latest update to ADS A by τ_H produces A[5] and the next update of A by τ_H produces A[7], i.e., some other transaction updated A once during the interval between τ_H's updates. In this case, τ will copy over both A[6] and A[7] to the remote sites. The time lag for the information between the home site and remote sites is given by Corollary 12. Since τ_H writes all the sites periodically, a logically and temporally consistent view can be obtained by reading the properly delayed versions produced by τ_H.

3. Conclusions

Distributed real-time database is an important area of research with applications ranging from surveillance to reliable manufacturing and production control. In this paper, we have integrated a modular concurrency control theory with a real-time scheduling protocol to create a real-time database concurrency control protocol, the setwise two-phase locking protocol with read-write priority ceiling. We have shown that this integrated approach is free from deadlocks and bounds the blocking encountered by a task at each processor to at most one elementary transaction until it suspends itself or completes. In addition, we have provided a schedulability analysis for a set of periodic tasks with embedded transactions. Finally, we examined the problem of ensuring a both logically and temporally consistent view in a distributed environment, where multiple versions of data with different

[m] Readers who are interested in the scheduling issues of communication media are referred to[13].

ages are provided to users.

References

1. Eswaran, K. P., J. N. Gray, R. A. Lorie and I. L. Traiger, "The Notion of Consistency and Predicate Lock in a Database System", *CACM, Vol. 19, No 11*, Nov. 1976.

2. Attar, R., Bernstein P. A. and Goodman N., "Site Initialization, Recovery and Backup in a Distributed Database System", *IEEE Transaction on Software Engineering*, Nov., 1984.

3. Beeri, C., P. A. Bernstein, N. Goodman, and M. Y. Lai, "A Concurrency Control Theory for Nested Transactions", *ACM SIGACT-SIGOPS Symposium on Principles of Distributed Computing*, 1983.

4. Bernstein, Phillip A., David W. Shipman, and Wing S. Wong, "Formal Aspects of Serializability in Database Concurrency Control", *IEEE Transactions on Software Engineering*, 1979, pp. pages 203 - 216.

5. Garcia-Molina, H., "Using Semantic Knowledge For Transaction Processing In A Distributed Database", *ACM Transaction on Database Systems, Vol 8, No. 2*, June, 1983.

6. Lynch, N. A., "Multi-level Atomicity - A New Correctness Criterion for Database Concurrency Control", *ACM Transaction on Database Systems, Vol. 8, No. 4*, December, 1983.

7. Mohan, C., Fussell, D., Kedem Z. M. and Silberschatz A., "Lock Conversion in Non-Two-Phase Locking Protocols", *IEEE Transaction on Software Engineering*, Jan., 1985.

8. Mohan, C., Lindsay, B. and Obermarck R., "Transaction Management in The R* Distributed Database Management System", *ACM Transactions on Database Systems, Vol. 11, No. 4*, Dec. 1986.

9. Papadimitriou, C. H. and Kanellakis, P. C., "On Concurrency Control by Multiple Versions", *ACM Transaction on Database Systems*, Mar., 1984.

10. Papadimitriou, C., *The Theory of Database Concurrency Control*, Computer Science Press, 1986.

11. Schwarz, P., *Transactions on Typed Objects*, PhD dissertation, Department of Computer Science, Carnegie-Mellon University, 1984.

12. Weihl, W. E. and Liskov, B., "Specification and Implementation of Resilient Atomic Data Types", *Proceedings of The SIGPLAN Symposium on Programming Language Issues*, June, 1983.

13. Lehoczky, J. P. and Sha, L., "Performance of Real-Time Bus Scheduling Algorithms", *ACM Performance Evaluation Review, Special Issue Vol. 14, No. 1*, May, 1986.

14. Lehoczky, J. P., Sha L and Strosnider, J., "Enhancing Aperiodic Responsiveness in A Hard Real-Time Environment", *IEEE Real-Time System Symposium*, 1987.

15. Leinbaugh, D. W., "Guaranteed Response Time in a Hard Real-Time Environment", *IEEE Transactions on Software Engineering*, Jan. 1980.

16. Leung, J. Y. and Merrill M. L., "A Note on Preemptive Scheduling of Periodic, Real Time Tasks", *Information Processing Letters*, Vol. 11 (3), Nov. 1980, pp. 115 - 118.

17. Liu, C. L. and Layland J. W., "Scheduling Algorithms for Multiprogramming in a Hard Real Time Environment", *JACM*, Vol. 20 (1), 1973, pp. 46 - 61.

18. Mok, A. K., *Fundamental Design Problems of Distributed Systems For The Hard Real Time Environment*, PhD dissertation, M.I.T., 1983.

19. Ramaritham K. and Stankovic J. A., "Dynamic Task Scheduling in Hard Real-Time Distributed Systems", *IEEE Software*, July, 1984.

20. Zhao, W., Ramamritham, K. and Stankovic, J., "Preemptive Scheduling Under Time and Resource Constraints", *IEEE Transactions on Computers*, Aug. 1987.

21. Sha, L., Lehoczky, J. P. and Jensen E. D., "Modular Concurrency Control and Failure

Recovery", *to appear in IEEE Transaction on Computers*, 1988.

22. Lehoczky, J. P., Sha, L. and Ding, Y., "The Rate Monotonic Scheduling Algorithm --- Exact Charaterization and Average Case Behavior", Tech. report, Department of Statistics, Carnegie-Mellon University, 1987.

23. Sha, L., Lehoczky, J. P. and Rajkumar, R., "Solutions for Some Practical Problems in Prioritized Preemptive Scheduling", *IEEE Real-Time Systems Symposium*, 1986.

24. Sha, L., Rajkumar, R. and Lehoczky, J. P., "Priority Inheritance Protocols: An Approach to Real-Time Synchronization", *Technical Report (CMU-CS-87-181), Department of Computer Science, CMU*, 1987.

25. Lampson, B.W., Redell, D.D., "Experience with Processes and Monitors in Mesa", *Communications of the ACM*,Vol. 23,No. 2,February 1980, pp. 105-117.

A DISTRIBUTED DECISION-AID FOR ARMY AVIATION*

Kshemendra N. Paul   Gilmer L. Blankenship[†]
Lawrence G. Lebow

Frank McLeskey[‡]
Techno-Sciences, Inc.
7833 Walker Drive, Suite 620
Greenbelt, MD 20770
(301)345-0375

Abstract

This paper describes the Army Aviation Mission Planning System(AAMPS), a distributed decision aid developed for Army Aviation Operations Management. AAMPS is an application for the Army-DARPA Distributed Communications and Processing Experiment(ADDCOMPE) testbed. The decision aid was developed with considerable input from, and is currently resident with, the 82nd Aviation Brigade (Avn Bde) at Fort Bragg, NC.

The paper will be divided into three sections. After an overview of the paper, we will discuss the motivation for the decision aid, the development strategy we are pursuing and the operational concept for implementation. Next, we discuss the technical approach and describe the configuration of the decision aid. Our technical approach consists of combining constraint directed reasoning with stochastic scheduling algorithms. The implementation of the decision aid exploits interactive software and workstation technology, and is distributed at the company, battalion, and brigade level. Finally, we will discuss our results and observations to date.

1 Introduction

Army Aviation will play a major combat and support role in the future AirLand battlefield. Aviation is a highly capable resource adding to the effectiveness of combat operations. Yet they are scarce and expensive, requiring careful management by Commanders. Effective Command and Control (C^2) is a prerequisite to success on the AirLand Battlefield, but Army studies and reviews indicate that the current C^2 is less effective than might be the case. This paper describes an automated, distributed decision aid for Aviation Operations Management.

Operations Management for Army Aviation is an extremely complex and time consuming task, made even more so by the AirLand Battle doctrine. The AirLand Battle doctrine requires Army Aviation to be responsive to a dynamically changing environment and to fight three simultaneous battles, namely deep attack, close in, and rear area. In particular, Army Aviation Operations Management requires

- Rapid, accurate aircraft schedule generation in the face of a large number of time sensitive, dynamically evolving changes for aircraft support.
- Ability for scheduling cell to perform *what if* analysis.
- Accurate Command/Staff integration of crew, aircraft, maintenance, mission, and doctrinal data.
- Ability to understand explicitly the relationship between scheduling constraints and the ability to satisfy mission demand.

Scheduling constraints can be doctrinal, like the *one third* rule[1], aircraft related, like scheduled maintenance which is a function of flight time, or crew related, like crew endurance. These same requirements carry over to peacetime garrison operations, to allow for more efficient use of assets and to maintain readiness and training levels.

The current manual system for operations management of aviation assets has response time which are not compatible with the dynamics of the AirLand Battle. This manual system includes posting and updates of status/update boards, development of mission assignment through *stubby pencil* analysis, detailed mission planning through the use of checklists and manually developed worksheets. Aviation activities are inherently distributed and current doctrine calls for single and multiband voice communications with limited teletype (data) capability. Aviation units will in general have to coordinate activities with not only supported units but also, for example, air defence artillery, and overflown units, as well as internally with maintenance components.

The design of the Man/Machine Interface, as well as much of the functionality of the aid, were driven by 82nd Avn Bde personnel. Currently, the decision aid is resident with the 82nd Avn Bde, and they are providing us with feedback which will allow us to modify the operational concept and enhance the functionality of the decision aid. This will continue through the test and evaluation phase in an iterative fashion to allow the interaction between technology and doctrine to reach a steady state.

The rest of this section provides a framework and motivation for AAMPS. After a discussion of the AirLand Battle Doctrine, the role of Aviation is explored and current methods for operations management are discussed. The Operations Management problem for Army Aviation is then defined, and its characteristics and relationships are described. The second part of this paper is a discussion of the technical approach. The approach used is a combination of constraint directed reasoning, stochastic scheduling theory, and interactive, frame based software. The last part of the paper is a discussion of the results and observations to date.

*This work is supported in part by the Communications Electronics Command (CECOM) of the U S. Army under Contract No. DAAB07 86-C-A036.

[†]This author is also with the Department of Electrical Engineering, University of Maryland, College Park, MD 20742

[‡]This author is with Science Applications International Corporation, 8619 Westwood Center Drive, Vienna, VA 22180

[1]To maintain a continuous presence, one third of the force should be on station, one third en route, and one third reloading and refueling

1.1 AirLand Battle Doctrine

This section provides a background discussion of the Army's AirLand Battle doctrine and its relationship with Army Aviation Command and Control (C^2). It highlights the key areas of the AirLand Battle doctrine that are important to the fulfillment of Army aviation missions on the battlefield. References on AirLand Battle and the ADDCOMPE testbed are [LEIN83], [LEIN84], and [FM100-5]

The AirLand battle concept of extending the battlefield in depth and time, with a focus on three simultaneous battles–deep, close-in, rear, was studied extensively and formulated into a written operational concept in March 1981. The AirLand battle doctrine became a formal part of the Army's tactical doctrine with the revision of the key Field Manual (FM) 100-5, in October 1982. The Field Manual was revised again in May 1986, but maintains the AirLand Battle as its central focus.

The essence of AirLand battle is the deep attack of enemy second echelon and follow-on forces as an absolute necessity. AirLand Battle requires the attack of enemy forces to their full depth simultaneously while defending against similar enemy action. Second, deep attack required tight coordination with the decisive close-in battle and with the rear battle so that scarce resources (means) of attack would not be wasted on targets that had little impact on the end result. The concept of AirLand battle clearly delineated the time aspects of combat operations, particularly the deep battle. At each echelon, times are specified for brigade, division, and corps commanders to attack respective elements of the enemy second echelon formations. It should be noted here that the Corps Commander became the central, coordinating figure in AirLand tactical operations. To make AirLand battle work, the Corps would control sensors and deep attack weapons and would coordinate closely with the Air Force to develop a single, integrated air and land interdiction campaign to defeat follow-on forces. As will be discussed below, the AirLand battle concepts and doctrine have major implications for the planning and employment of Army aviation assets.

The AirLand battle will be influenced by time, distance, and resources on battlefield where the enemy is attacked to the full depth of his formations. Operations, including Army aviation, will be conducted continuously in adverse weather and light conditions. Logistical readiness and sustainability are critical to successful implementation In summary, the AirLand Battlefield has the following characteristics, all of which point out the need for effective C^2

- Nonlinear in flow.
- Highly dynamic.
- Continuous operations
- Multiple simultaneous engagements.

1.2 Role of Aviation

The salient feature of the AirLand battle concept is maneuver in which Aviation operations play a major role. Successful application of maneuver will include rapid movement of forces, speed, surprise, fire power and deception to obtain a significant advantage at the decisive point in the rear, close-in, or deep battle. References on Aviation Operations Management are [FM17-50], and [FM1-100].

Army aviation is required to provide both a vertical and time dimension to AirLand battle operations. Army aviation can provide rapid movement of forces, add speed, time compression and momentum to U.S. forces military operations. It can significantly enhance key AirLand battle missions. For example, in close combat, Army aviation can provide quick dispersion and massing of forces, surprise and security, and undertake offensive actions rapidly In fire support, Army aviation can detect targets rapidly, provide an area fire weapons system to destroy enemy targets, provide aerial fire support adjustment capability, and rapidly transport weapons systems, supplies, and personnel about the battlefield. In air defense, Army aviation missions include engaging enemy air elements with air-to-air weapons, acquire and engage enemy air defense weapons, provide visual identification of enemy air units to ground air defense units, and rapidly transport air defense personnel and equipment about the battlefield. In combat support, Army aviation missions are to move all types of support across the battlefield using utility, medium and heavy-lift helicopters, and rapidly transport air assault forces and materiel into deep attack on rear areas

The above are examples of Army aviation missions on the AirLand battlefield. The key point is that Army aviation offers resources that can be applied in support of many mission areas Army helicopters, if employed properly, can be a highly effective resource, adding significantly to the effectiveness of conducting combat operations. Helicopter assets must be intensively managed and controlled by Corps and Division commanders for maximum payoff However, Army studies and reviews indicate that the management, control, and overall utilization of helicopters as envisioned in AirLand battle operations is much less than optimum and must be improved.

1.3 Current Methods for Operations Management

The section discusses how the XVIII Corps, 101st Airborne Division, 82nd Airborne Division, and 28th National Guard Division are implementing AirLand battle doctrine in their control and scheduling of helicopter assets.

The XVIII Corps Aviation section's primary activity is to task subordinate units to carry out missions in both peace and war Tasking of aviation assets in peacetime is based on mission requests from Corps units that are received in the aviation office. A manual, ledger system is used to record all of the pertinent mission request information. Once a request is logged in, telephonic procedures are used to task an aviation unit (e.g., 269th battalion) to carry out the mission. The 269th will accept or reject the mission based on factors such as aircraft availability. The Corps aviation section does not specify to the tasked aviation unit such items as the aircraft tail number or how the mission is to be flown. The tasked aviation unit works out the details of the mission directly with the unit requesting support.

While Corps aviation performs the administrative aviation allocation function in peacetime, wartime planning on the use of aviation assets would be accomplished in close coordination with the G-3 (operations/plans) staff section. The result of this coordination would be the Corps' tasking of specific aviation units to fulfill combat/combat support/combat service support missions.

Airspace management/control is a problem area of major concern to the Corps. Mission deconfliction and safety of Air Force high performance aircraft and Army helicopters in the air corridors is mandatory under AirLand battle doctrine. Specific impacts of AirLand battle doctrine (e.g., simultaneous battles, activities beyond visual range) on Army aviation functions/activities have not yet been addressed at the Corps level. A general requirement for aviation logistics activities to have automated support has been identified by the Corps to track repair/spare parts and job orders and to schedule maintenance on aircraft.

A macro view of Corps aircraft readiness (aircraft availability, by type) is the key factor in helicopter planning activities. Mission planning takes place based on these availability numbers and subordinate aviation units are tasked carry out specific missions. Aircraft assets are not centrally controlled/managed in the 18th Corps aviation section in peacetime and no mechanism appears to exist to modify this for wartime operations.

At the 82nd Combat Aviation Brigade (CAB) there is insatiable demand for helicopter support within the division during peacetime. Much of this demand is caused by the requirement to provide 7 day, 24 hour per day support to the three maneuver brigades, as they proceed through the standard mission, support, and training cycles. In addition, many requests for aviation support are also received from the division artillery and division support command. The current scheduling system is semi-automated, the G-3 establishes user priorities and approves unit requests. The aviation brigade attempts to satisfy all missions through aircraft availability rates. Corrective maintenance scheduling, pilot/crew availability/shortages, and individual/unit flight training requirements have been analyzed and a daily availability rate has been established by the division commander.

The current semi-automated support used by the CAP does not provide adequate information to project aircraft availability (e.g., 8 Blackhawks/day) with an acceptable degree of flexibility and confidence. A specific need exists to maximize the utilization of the Blackhawk (UH-60). Currently the 2 Blackhawk companies of 15 aircraft each cannot effectively support the division's integrated training cycle without major demands on aircraft scheduling/allocation personnel, maintenance workloads, and juggling crew training/assignments. Because of peacetime scheduling problems, little emphasis has been placed on how AirLand battle operations missions should be conducted.

The 101st Division also focuses primarily on peacetime operations. Little information was available on carrying out combat aviation operations. Emphasis is on automating information for maintenance/readiness purposes. The 101st Aviation Battalion maintains a data base on 380 aircraft. The primary use of the data base is to assist in scheduling maintenance and to update flight hours for each aircraft. The large number of aircraft in the division allows the CAB staff to allocate missions to the aviation battalions and the battalions either perform the missions or request that the CAB task another unit. The CAB staff either tasks another unit to perform the mission or directs the originally tasked unit to perform the mission in lieu of a lower priority mission.

The 101st CAB does not have the same problem as the 82nd CAB in providing the required support to the division's integrated training cycle. Less than 20 percent (60 of 380 aircraft) are required per day to support training requirements. The division also has not participated fully in field exercises in which the range of AirLand battle operations (i.e., close, rear, deep battles) have been practiced. Where deep battle exercises have been held, detailed planning was conducted weeks in advance. At the 101st CAB level, status of aircraft/crews during exercises is maintained on a large, grease pencil chart. Updates are provided several times daily during the exercise by the aviation battalions.

The 28th National Guard CAB is organized under the Army of Excellence (AOE) structure, the same as the 82nd CAB. UH-1's are assigned to the units instead of Blackhawks. The system to task/respond to missions is totally manual. The G-3 of the division tasks the 28th CAB to carry out missions that subordinate units have requested. There is no apparent priority system to filter the missions. If the helicopter company cannot perform the mission due to helicopter availability, crew status, or other reason, the company informs the Brigade S-3. The tasking system also uses mission request forms similar to those used in the 82nd CAB. Communication of the requests to the 314th is either by unsecured telephone or courier. At the company level, mission assignments of helicopter and crews for exercise activities are made the night before the mission is to be flown. Problems arise if ongoing missions go longer than expected and force changes in helicopters or crews.

The 28th CAB has identified requirements for ADP support for operations scheduling and to track key C^2 data such as pickup times, zones, crew status, maintenance status. The types of missions requested to be flown, concerns (e.g., maintenance), need for responsive planning time frames, and overall lack of an efficient scheduling system are similar, on a smaller scale, to that of the 82nd CAB.

1.4 Deficiencies

The most significant deficiencies of the C^2 system are a listed below and are with respect to the AirLand Doctrine. The listed deficiencies serve as a motivation for AAMPS.

- Lack of sufficient emphasis on Army Aviation C^2 in the conduct of AirLand battle training.
- Emphasis by aviation units on peacetime operations and readiness rather than on training for the execution of AirLand battle operations.
- Response times in planning and controlling helicopter assets that are inadequate in peacetime and are projected to be seriously deficient in wartime.
- Reliance on communications and manual systems that are limited in updating critical mission, helicopter status and other key information required for responsive helicopter management and scheduling.
- Time response required to meet helicopter mission planning cycles.
- Inability to deal with dynamically changing factors such as aircraft status, crew availability/status, maintenance response times, aircraft flying hour requirements, mission changes, and environmental changes, for example.

Army aviation has the mission to support AirLand battle operations by becoming a force multiplied in two dimensions-airspace and time. Army aviation can overcome terrain impediments to support AirLand missions and can react much more rapidly than ground forces to place combat power at decisive points in the battle. However, the complexities of AirLand battle (and Army Aviation implementation of it) dictate that a responsive C^2 system is the key to achieving these goals.

1.5 Problem Statement

Our approach to the Army Aviation management, command, and control was to identify a limited scope problem which captured as much of the essential difficulties as possible while remaining small enough to take to completion within the scope of our effort. We were to identify a line Army Aviation unit to work with, and with their input develop a prototype distributed decision aid. The aid was to be installed and evaluated in the user site, with the active participation of the aviation unit in both garrison and field environments.

The limited scope problem we chose is to do operations management for Blackhawk Aviation Companies, and the unit we are working with is the 82nd Avn Bde. Specifically, we have developed a prototype distributed decision aid for operations management for Companies A and B, 2/82nd Air Assault Battalion. The decision aid is called the Army Aviation Mission Planning System (AAMPS).

The operations management problem was formulated as a resource allocation problem imbedded in a constraint directed reasoning framework The objective function for the resource allocation problem has two layers The first goal was to select a flight schedule which minimized the number of unscheduled missions. Among proposed missions, there exists a quantifiable priority structure. The second goal was to maximize the readiness (availability over time) of the aircraft fleet. A related goal was to minimize the number and total flight hours used to met the mission demand. To be acceptable, the schedule has to be within all of the constraints.

Constraints can be categorized as hard or soft. A soft constraint is one that can be relaxed, up to some limit, in order to produce a schedule which is within all constraints. When a hard constraint is one that cannot be relaxed. Organizational Constraints are examples of soft constraints. Limits on cargo loads is an example of a hybrid constraint. The maximum cargo load for an aircraft starts at the maximum safe load, and is relaxable up to the maximum load allowed by lift capacity of the aircraft. An example of a hard constraint is the physical limitation of aircraft to do only one thing at any instant of time. Many constraints can be thought of as decision variables for the operations management problem, and the relaxation of a constraint can be interpreted as a command decision.

The operations management problem thus becomes the following:

1. Produce a candidate flight schedule, using the two tiered objective function:
   - Minimise the number of unscheduled missions.
   - Maximize the readiness of the aircraft fleet.
2. The candidate schedule must obey all of the constraints. Missions which could not be scheduled must produce a list of active constraints which prohibited their scheduling, and suggestions as to which constraints should be relaxed in order to schedule the mission.
3. The flight scheduling officer can do what if analysis using the suggestions and his/her own insight. This process continues in an iterative fashion until the flight operations officer reaches a balance between scheduling constraints and the mission demand.

The operations management problem is inherently distributed. Mission demand is generated throughout the division and corps, and status information for the aircraft is generated in the aviation companies which execute the missions and maintain the aircraft. These facts motivated the initial operational concept calls for AAMPS, which is a functionally distributed implementation of the decision aid, with nodes at company, battalion and brigade level At the brigade level, the decision aid would perform a mission filtering and unit assignment function. The mission filtering function is basically a capacity analysis and allows the brigade to decide if it can support the mission or not. Typically, the brigade has a planning horizon of six weeks in peacetime, three to five days in wartime

At the battalion level, the decision aid would be used to schedule helicopters by tail number. Typically, the battalion has a planning horizon of two weeks in peacetime, two days in wartime. The battalion level includes inputs from the subordinate maintenance company on aircraft status. At the company level, the decision aid would be used to schedule crews and to receive orders in the form of schedules. The companies typically have a planning horizon of one week in peacetime, one day in wartime. All of the status information on crews and aircraft are generated and enter the system at the company level. The same decision aid would be used at all levels, but for different purposes. Also, there is a single database which is shared among all users. This initial operational concept for the decision aid was developed in conjunction with the 82nd AvnBde and represents only a starting point As stated in the development strategy, we expect it to change as we gain experience with it.

2 Technical Approach

It has been shown that the kind of scheduling activities involved in Aviation Operations Management is a constraint driven process. Our decision aid is predicated on this assumption and is an implementation of the constraint directed reasoning paradigm Constraint directed reasoning allows the user to see explicit relationships between constraints and unsatisfied mission demand The user can then relax the constraint(s) in question if the need (unsatisfied mission demand) so warrants. By making the relationship between constraints and unsatisfied demand explicit, the user can make quantitative judgments about allocating resources.

The scheduling algorithms we used are based on stochastic models for the availability of assets and for potential demand. They are motivated by index rule approach for scheduling many jobs on many parallel independent machines. We cannot use the optimal results for our problem because of the complex maintenance models involved. In the course of our effort, we discovered that the algorithms we developed, motivated by stochastic operations research ideas, were similar in structure to the approach used by the operations officers we spoke with. This is because the availability models we assume are strongly stochastic in nature, and this reflects actual experience.

2.1 Stochastic Modeling

As Pinedo [PINE83] correctly points out, there is a fundamental difference between deterministic and stochastic scheduling problems. Often [FORE78], [PATT82], [GLAZ80], [GLAZ79] stochastic scheduling problems with exponential (in continuous time) or geometric (in discrete time) processing times, have very simple optimal policies minimizing a variety of criteria, such as the expected weighted sum of completion times or the weighted number of late jobs. A well known policy of this type is the so-called "μc rule" [BARA85]. On the other hand, the corresponding deterministic scheduling problems are NP-hard [LENS77A], [KARP72], [GRAH79]. This means that for the latter problems, no polynomial time algorithms are known. This contrast between deterministic and stochastic scheduling problems can be explained on intuitive grounds. when only the distributions of the processing times are known, finding the optimal policies is easier than when the actual realizations of the processing times are known. In the latter case, there is such precise information available to the decision maker that it may not be possible for him to take advantage of it in an efficient way. When the decision maker only knows the distributions of the processing times, there is of course, much less information available to him, which makes it easier to determine the optimal policies.

In helicopter operations management randomness is present in various elements of the problem. Clearly failures of equipment are random and should be properly incorporated in the overall model. We used Markov chain models with a small number of states. Other elements of randomness include attrition, mission execution time, and length of scheduled maintenance. All of these led to an availability model which was strongly stochastic for each aircraft. Since the availability of each aircraft can be modeled as independent random variables, the aggregate availability for a blackhawk company can be modeled accurately. The demand process is also random, and missions can change not only after initial specification but also during execution.

These random phenomena have been incorporated in the scheduling algorithms developed during the course of the effort. It is quite fortunate that typically the latter are much simpler than deterministic algorithms. Since we cannot obtain optimal strategies, due to the problem complexity and to non-quantitative scheduling rules (such as priority commands, military doctrine etc), it is really unwise to include very complex and computationally intensive optimal deterministic scheduling algorithms.

2.2 Dynamic Scheduling

Basic references on sequencing are [BAKE74], [LENS77A], [ELMA73], [ASHO72], [COFF76], [WEIS82]. Abstractly speaking, a sequencing problem is the determination of a sequence of execution times for a set of partially ordered operations. Such partial orderings are typically called precedence relationships. Since it is well known that the great majority of sequencing problems are NP-complete if formulated as optimization problems, it would be futile to attempt the solution of a multiple criteria sequencing problem. Thus we will not attempt to solve a huge multiple criteria sequencing optimization problem. Rather we are looking for systematic ways to evaluate alternative allocation plans, or to assess their performance by a variety of performance criteria. With such a goal in mind, it is indeed advantageous to look into optimization based formulations, since through the appropriate duality relationship these optimization based formulations do provide the tools for systematic evaluation of sequencing strategies.

Clearly the problem is combinatorial in nature. A variety of techniques exist to determine feasible sequences. There exists a variety of tools for comparing pairwise feasible sequences. One of the basic research problems we propose to explore is to generalize some of these methods so that the above requirements, necessitated by realistic considerations for helicopter management, are met. Particularly useful are techniques to compute performance bounds for certain sequence selection strategies.

The scheduling algorithms used are motivated by so called list policies [WHIT81A], [GITT79], [PINE83], [WEBE82]. A list policy arranges all jobs (i.e. mission segments and missions) in a list, say $j_1, j_2, \ldots, j_n$ and at any point in time processes the available job with smallest index. There are basically two types of list policies. The first class, called static list policies, specify the ordering list at the outset of the scheduling process. These policies therefore specify right at the beginning of the schedule which job (or request or mission) will be completed at each point in time. The second class of policies are called dynamic policies. In a dynamic policy, the decision-maker is allowed to determine his actions at any moment in time, while taking into account all the information that has become available up to that moment. Clearly from the discussion so far the Aviation Operations Management problem is dynamic.

For resource allocation problems appearing in Army Aviation operations we have used primarily dynamic list policies. As indicated earlier these can be shown to be optimal, under a variety of cost criteria (therefore simplifying our multiple-criteria approach) and are automatically adaptive, for stochastic scheduling problems. One of the greatest advantages of list policies, is their simplicity and their feasibility for distributed implementation. Dynamic list policies are also called dynamic priority assignment policies. In addition to the well known "μc rule" other list policies in scheduling problems are LEPT (Longest Expected Processing Time) and SEPT (Shortest Expected Processing Time). The latter are also optimal for the scheduling problems they solve, under a variety of performance measures.

Initially we the list policy we used was similar to the SEPT rule, which will tend to minimize the total number of late or unscheduled missions. Feedback from the user made us change to a variant of the LEPT rule, since it turns out that larger missions have an implicit higher priority than smaller missions, even if they have the same explicit priority. Here the size of a mission is measured as the number of aircraft required times the expected flight time for the mission.

The second goal of the two tiered objective function is to maximize the readiness of the aircraft fleet. Readiness is measured as the number of mission capable aircraft (both fully and partially) over some period of time. The Army articulates this goal as minimizing the maximum number of aircraft out of service for a period of time. One example of this is that it is a desirable characteristic for a fleet of aircraft to be banked with respect to flight hours till phase maintenance is due. Phase maintenance is the periodic major overhaul of the aircraft[2] Banked means that the aircraft are evenly separated in flight time so that the maximum number of aircraft in phase maintenance is minimized. The concept of banked hours can be expanded to include all types of scheduled downtime, suggesting a natural ranking of aircraft with respect to the desirability of flying. The ranking is calculated as follows:

1. All of the aircraft currently available are sorted with respect to flight hours left until the scheduled downtime is due

[2]For the Blackhawk, phase maintenance is due every 500 hours. It takes from one to three months to perform in peacetime.

2. The difference between the aircraft's value for flight hours left and the nominal value if all aircraft were evenly separated is calculated. Note that if the actual value is less than the nominal value, the ranking is positive.

3. The above calculation is made for all scheduled downtimes.

4. A weighted average of the calculated values is made. The weights are selected empirically, but the values we use are proportional to the length of the downtime.

5. The aircraft are sorted in ascending value of the ranking calculated above. The more desirable aircraft to fly are those with larger ranking.

The above rankings, both for missions and helicopters, are central to the scheduling algorithms. AAMPS will take the first mission from the mission rankings and try to schedule it to the aircraft at the top of the aircraft list. AAMPS will cycle through the aircraft until either it can schedule the mission or until it runs out of aircraft. If it runs out of aircraft then AAMPS will move the mission to the unscheduled list and update the active constraint list. Otherwise, it will update the status of the aircraft to reflect the scheduled mission. AAMPS will then consider the next mission.

If certain constraints become active, AAMPS will automatically relax these within prespecified bounds. Examples of this include scheduling of scheduled maintenance. This is discussed in the next section, Constraint Directed Reasoning.

2.3 Constraint Directed Reasoning

Scheduling may usefully be modeled as a constraint directed process. We take this point of view and represent scheduling knowledge as constraints. Some of these constraints can be thought of as decision variables, and are under the control of the scheduler. AAMPS quantifies the relationship between the set of constraints and the unscheduled mission demand. Thus the flight operations officer can evaluate *what if* options by manipulating the decision variables and analyzing the resulting schedules. As mentioned above, constraint directed reasoning has three facets.

- Fast, interactive schedule generation with specified values set for decision variables.
- Automatic constraint relaxation for certain variables.
- Schedule suggestions as to how to relax constraints or modify the mission specifications in order to produce a better schedule.

The interactive nature of AAMPS will be discussed below in the section, Implementation. The other items are discussed below. References on Constraint Directed Reasoning and other heuristic approaches are [HAYE83], [DUDA76], [BUCH84], [FOX83].

AAMPS will automatically relax certain constraints subject to prespecified models and parameters. By far the most complex class of constraints handled in this manner are maintenance constraints. The Blackhawk, as are all high performance aircraft, is a maintenance intensive combat vehicle. The Blackhawk requires six different types of periodic maintenance triggered by flight hours, as well as inspections based upon calender days and special inspections. For this example we will consider only maintenance due to flight hours. Each maintenance of this type must occur in a window around the nominal due time. AAMPS monitors the flight time left until each type of maintenance is due, for each instant of time. When checking the maintenance constraints for a particular mission / helicopter pairing, AAMPS will check if the time left till maintenance is due is less than the expected flight time of the mission. If it is, AAMPS will check if it is possible to schedule the appropriate type of maintenance, in such a way as to increase the time until maintenance is due. If it is possible, AAMPS will schedule the maintenance.

AAMPS also produces suggested modifications of schedule constraints in order to improve the flight schedule. These suggestions are generated by a production rule expert system with a limited rule base. The initial rule base includes obvious suggestions like releasing a reserved helicopter, if that would allow the mission to be scheduled. In the future we plan to incorporate rules based upon operational experience with the system. A heuristic approach must be used for the following reasons. If multiple missions are unscheduled, the suggestions would probably conflict and the set of active constraints may change after scheduling an additional mission. Secondly, the interpretation of certain constraint relaxation is difficult to interpret and the explicit calculation of the effect would take too much time. Finally, heuristic have proved valuable for similar problems, and allow AAMPS to incorporate operational experience in a natural way.

2.4 Implementation

AAMPS has been implemented as an application on the ADDCOMPE testbed at Fort Bragg, North Carolina, under the sponsorship of USA CECOM. The computer program is written in the C programming language and runs on a network of SUN[3] workstations. The program takes full advantage of the native windowing and graphics facilities of the SUN, and is meant to be used in an interactive manner. Figures 1 and 2 are screen dumps of the system in use, and they are explained below.

The strength of the program lies in the internal data representation scheme used. A frame based, dynamic representation of scheduling data is used. A detailed description is beyond the scope of this paper, but a discussion of the dynamic treatment of time is essential to understanding how the scheduling algorithms work. AAMPS sets up a linked list of time block data structures. Each time block represents a homogeneous activity, for example participating in a mission or being scheduled to maintenance. Each time block has a start and end time, which AAMPS represents internally as the number of seconds from January 1, 1971. AAMPS must have a time horizon, which it uses to initialize the time block structure. The time horizon is set to ten weeks, but this is arbitrary and could be ten years. The resolution of AAMPS is a second. If an activity is scheduled, a new time block must be allocated and initialized in the appropriate place in the list of time blocks, and all state variables must be updated. Dynamic time structures are used to keep track of aircraft schedules, time varying constraints, and resource utilization.

Figure 1 is a screen dump which includes the AAMPS command window, the timeline display, and the schedule assistance window. The command window is the default display, and from it the user can access the data areas (by way of the icons), save / load data

[3] TM SUN Microsystems

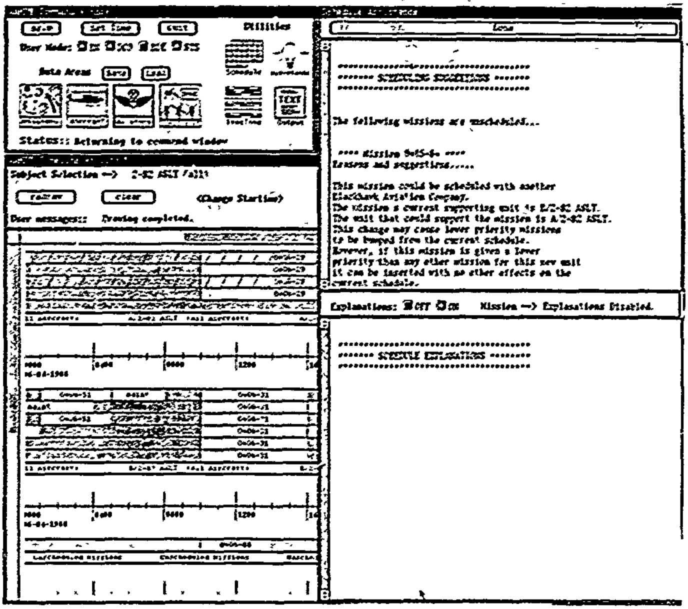


Figure 1: Screen Dump with Schedule Assistance Window Open

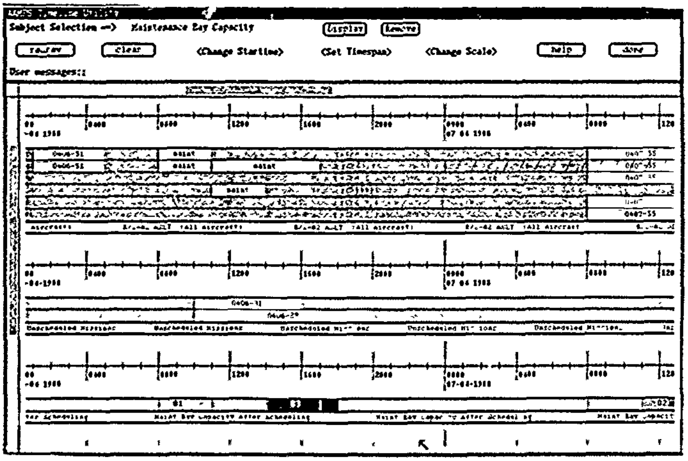


Figure 2: Screen Dump of Timeline Window

from disk, set the user mode, access the AAMPS online, context sensitive help facility, and use any of utilities. The utility programs include generation of a schedule, viewing the schedule graphically, generating operations orders (text icon), and displaying schedule assistance (as is done in the figure).

Figure 2 is a screen dump of the timeline window which reflects the schedule with peacetime duty hours for the maintenance company. Peacetime duty hours are ten hours a day, five days a week. Note that this causes missions to be unscheduled. In figure 1, the maintenance company is assumed on duty 24 hours a day seven days a week. The maintenance bay capacity after scheduling is also displayed using the timeline facility. Display of critical resources and constraint levels is useful because it allows the scheduling officer to quantify performance, and it allows for insight into how to improve the schedule.

3 Results/Observations

The project is still ongoing, and the bulk of the work left is in the area of test and evaluation. Nonetheless, several conclusions can be drawn and observations made at this stage. They are in the following categories: operational / doctrinal relevance, utility of AAMPS, and the research approach taken.

As highlighted by the discussion above responsive C^2 is the key to success on the AirLand battlefield, and responsiveness is the key deficiency in current implementations of Aviation Operations Management. AAMPS was designed with these two assertions as central tenets. The feedback so far from operational users is quite positive, and in the near future we will take AAMPS into a field environment, for a more complete picture of the relevance and limitations of AAMPS with respect to AirLand battle doctrine.

The goal of our effort was to investigate the feasibility of applying the technical methodologies outlined in the paper to Army Aviation C^2 problems. We were to do this by investigating a limited scope problem which was representative, and develop a prototype decision aid. AAMPS is the prototype, and as mentioned above, AAMPS has utility to the operations management problem. AAMPS should be expanded in both scope and functionality before becoming a fielded system. It should expand to include the entire Aviation Brigade structure, with nodes located with all divisional major subordinate commands and the division G-3. The scope should be expanded to include scheduling of crews, brigade administrative functions, and mission planning. The system should be ported to standard military hardware in the future, after the design and configuration becomes stable.

The approach we took to the research project was to identify a line Army Aviation unit, identify appropriate personnel and understand the C^2 problem from their point of view. Then in conjunction with the aviation personnel identify the limited scope problem. Next, in an iterative fashion, develop the functionality and MMI of the system. Finally, test and evaluate the prototype with the aviation unit using real data. The danger here is that the prototype is just an automation of current procedure. We feel that we were able to avoid this trap, and develop a prototype which has utility but still represents a significant innovation.

4. Acknowledgements

The authors would like to acknowledge the contributions of Dr J.S. Baras[a] in the initial stages of this research. Mr V. Khera and Ms I. Yan also made significant contributions.

5. References

LEIN84 B. Leiner and T. Klein, Public Information Briefing Charts on ADDCOMPE Distributed Command and Control Applications, Arlington, Virginia, July 24, 1984.

LEIN83 B M. Leiner, T. Klein and M Frankel, Joint ARMY/DARPA Distributed Communications and Processing Experiment, Volume II, Technical Plan, IPTO-83-II, November 1983.

BAKE74 K.R. Baker, INTRODUCTION TO SEQUENCY AND SCHEDULING, John Wiley and Sons, 1974.

LENS77a J K. Lenstra, SEQUENCING BY ENUMERATIVE METHODS, Mathematical Centre Tracts 69, Mathematisch Centrum, Amsterdam, 1977.

ELMA73 S E. Elmaghraby, ed., SYMPOSIUM ON THE THEORY OF SCHEDULING AND ITS APPLICATIONS, Springer-Verlag, 1973.

ASHO72 S. Ashour, SEQUENCING THEORY, Lecture Notes in Economics and Math Systems 69, Springer-Verlag, 1972.

COFF76 E G. Coffman, Jr, ed, COMPUTER AND JOB-SHOP SCHEDULING THEORY, John Wiley and Sons, 1976.

FORE78 J P. Forestier and P. Varaiya, "Multilayer Control of Large Markov Chains," IEEE TRANS. on AUTO CONTROL, Vol. AC-23, April 1978, pp. 296-305.

PATT82 K R. Pattipati and D.L. Kleinman, "Priority Assignment Using Dynamic Programming for a Class of Queuing Systems", IEEE Trans. on Autom.Control, AC-26, No. 5, pp. 1095-1106, 1181.

GLAZ80 K D Glazebrook, "On Stochastic Scheduling with Precedence Relations and Switching Costs", Journal of Applied Probability, Vol. 17, pp. 1016-1024, 1980.

PINE83 M. Pinedo, "Stochastic Scheduling with Release Dates and Due Dates", Operations Research, Vol 31, No. 3, May-June 1983.

GLAZ79 K.D. Glazebrook, "Scheduling Tasks with Exponential Service Times on Parallel Processors", Journal of Applied Probability, Vol. 16, pp. 685-689, 1979.

BARA85 J.S Baras, A.J Dorsey and A Makowski, "Optimal Control of Two Competing Queues with Linear Costs and Geometric Service Requirements The μc rule is often optimal", Journal of Applied Probability, March 1985.

LENS77b J.K. Lenstra, H.G Rinnooy Kan and P. Brucker, "Complexity of Machine Scheduling Problems", Ann. Discrete Math. 6, pp. 281-300, 1977.

[a] Director, Systems Research Center, University of Maryland, College Park

KARP72 R M. Karp, "Reducibility Among Combinatorial Problems", in Complexity of Computer Computations, pp. 85-103, R.E. Miller and J.W. Thatcher (eds.), Plenum Press, New York, 1972.

GRAH79 R.L. Graham, E L. Lawler, J.K. Lenstra, and A.H G. Rinnooy Kan, "Optimization and Approximation in Deterministic Sequencing and Scheduling: A Survey. Ann. Discrete Math. 5, pp. 287-326, 1979.

WHIT80 P. Whittle, "Multi-Armed Bandits and the Gittins Index", Journal of the Royal Statistical Society, Vol. 42, pp. 143-144, 1980.

GITT79 J. C. Gittins, "Bandit Processes and Dynamic Allocation Indices", Journal of the Royal Statistical Society, Vol. 41, pp 140-177, 1979.

GLAZ80 K. D. Glazebrook, "On Randomized Dynamic Allocation Indices for Sequential Design of Experiments", Journal of the Royal Statistical Society, Vol. 42, pp. 342-346, 1980.

HAYE83 F. Hayes-Roth, D.A. Waterman and D.B. Lenat, Building Expert Systems, Addison-Wesley, 1983.

WEBE82 R.R. Weber, Scheduling Jobs with Stochastic Processing Requirements on Parallel Machines to Minimize Makespan or Flow Time, J. APPL. PROB., 19, pp. 167-182, 1982.

WEIS82 G. Weiss, Multiserver Stochastic Scheduling, in Deterministic and Stochastic Scheduling, pp. 157-179, M A.H. Dempster et al. (eds), D. Reidel, Dordrecht, Holland, 1982.

DUDA76 R.O. Duda, P.E. Hart, and N.J. Nilsson, "Subjective Bayesian Methods for Rule-Based Inference Systems", SRI Technical Note 124, Jan. 1976.

BUCH84 B G. Buchanan and E H. Shortliffe, Rule-Based Expert Systems, Addison-Wesley, 1984.

FOX83 M. Fox, Constraint Directed Search: A Case Study of Job Shop Scheduling, Ph D. Thesis, Carnegie-Mellon University, 1983

FM17-50 Attack Helicopter Operations, Field Manual No 17-50, Headquarters, Dept. of the Army, Washington, DC, July 1977

FM100-5 Operations, Field Manual No. 100-5, Headquarters, Dept. of the Army, Washington, DC, May 1986

FM1-100 Combat Aviation Operations, Field Manual No. 1-100, Headquarters, Dept. of the Army, Washington, DC, September 1984.

ORGANIZING EW INFORMATION TO SUPPORT NAVAL TACTICS

John T. Egan

Naval Research Laboratory, Washington, DC 20375-5000

Donald S. Lindberg

American Technology Corp., Fairfax, VA 22031

ABSTRACT

This study examines the problem of organizing EW information to support naval tactics. EW information is fused from intelligence, security (cryptologic) and organic sources, and is needed to successfully conduct broad EW tactics (EMCON, OPSEC, C3CM, etc.) in support of force objectives. The problem is presented from a system perspective where it is partitioned and discussed within a coherent framework. This decomposition results in three different assessments dealing with the nature and flow of the information, the proper way to organize it and the extraction of relevant and timely information to support tactical actions. A general theory of combat tactics is presented leading to development of models which guide the organization of information. A general EW fusion system is postulated which should be capable of analyzing those conditions for the successful commitment of EW tactics and countermeasures. An example is given which illustrates the problem and illuminates the conditions for a technical solution.

INTRODUCTION

The ability to acquire and apply information concerning enemy combat forces is a central problem of command and control. Because tactical EW resources collectively comprise a primary means of threat observation, this report examines the problem of EW information handling as it supports the employment of tactical actions in combat. The problem consists of how the EW information should be handled to reflect the dynamics of the operational world in a manner that is relevant for planning, for recognizing the conditions for tactical commitment, and for providing the conditions that permit control of the tactics execution. The focus of this study is to expose and address key issues affecting this problem so that technical solutions may evolve.

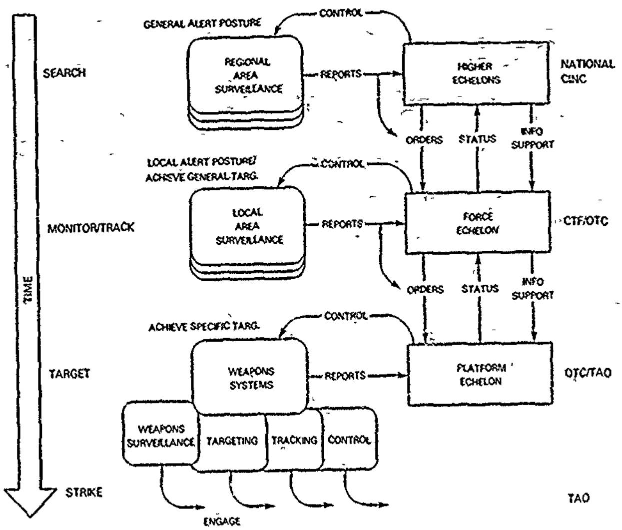
One way of explaining the problem is to do so in terms of the flow, organization and relevance of the information. Consider a simple model of a general force system (fig. 1). At each echelon, the tactician is required to deal with the flow of two fundamentally different kinds of information. information flowing from above and information flowing from within. The former are reports from sources outside the force, and the latter are reports from sources controlled by the force. The former tend to be more strategic or technical and broad ranging. The latter are more detailed and voluminous. Normally the principal tactician does not control the sources from above, but exercises full control over those from within since the information is provided by means organic to his echelon.

As one proceeds downward along the battle time line information must be organized to confront more definitive threat possibilities, leading ultimately to engagement. This means that at the lower echelons the tactician must be presented with tailored products relevant to his needs and of appropriate quality (accuracy, timeliness, coverage). How well the structure handles the flow and organization of its information and determines relevance is a measure of its effectiveness. It must be able to coordinate disparate sources of information to focus upon the same objects. It must be able to manage information so that its organization reflects the nature, status and trends of on going operations. Finally it must provide information of the right type, of the right amount, at the right time, to the right users to determine windows of tactical opportunity and vulnerability. The result will be the selection of tactics and countermeasures optimized for success and impact against a threat in almost any tactical situation.

THE NATURE OF EW INFORMATION

There are essentially three basic categories of tactical information that are available to a tactician at the force level and in some cases the platform level - intelligence, security (cryptologic) and organic. EW information is a product drawn from these three types as well as from other classes of information which forms a bounded set of information needed to formulate tactical EW plans. As such, EW information contains data both from above and within. The kinds of EW tactics supported by this information are broad in scope and are intended to impact the thinking of the enemy commander. They include tactics to:

- gain information (directed observation)
- withhold information (EMCON, OPSEC)
- provide misinformation (OPDEC, Countertargeting)
- disrupt information (C3CM, Offensive ECM)
- protect information (Own-C3 protect)



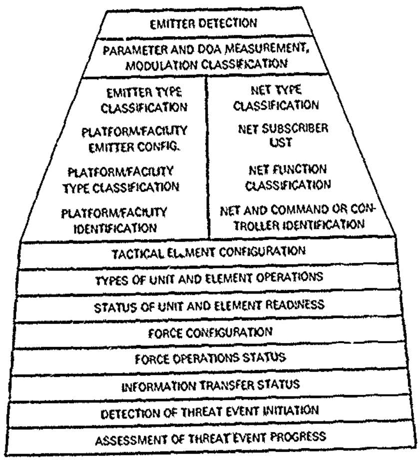
GENERAL FORCE SYSTEM

Figure 1

and have been previously discussed by Layman [1]. Terminal defense jamming and other isolated EW actions are not included.

These tactics share other characteristics as well, they require persistence in their execution and consistency in the operational patterns that they expose to a hostile force. This implies a need for further information to evaluate the effectiveness of the tactic and to replan, reposition and refocus effort where required.

The structure of EW information is shown in figure 2. At the top of the figure, data in the form of emitter detections, direction of arrival and modulation parameters usually furnish the first subjects for assessments. Upon initial assessment these fall into either of two categories ELINT and COMINT where further analysis will lead to the identification of platforms and facilities, in the case of ELINT or the identification of nets and commands in the case of COMINT. As the fusion process continues, and further technical and combat information is brought to bear, the quality of information reaches a point where it is possible to gauge the progress of an ongoing threat tactic. It should be emphasized, however, that the figure represents an ideal situation where all of the data is available and that there is enough time for assessments. In the vast majority of cases, the information will be incomplete.



EW INFORMATION STRUCTURE

Figure 2

EW INFORMATION FLOW

The problem of EW information flow is viewed as one of obtaining the necessary information to fill in as much of the EW information structure as possible and to do so in a timely manner. This will involve not only the coordination of information from above and within but will also require interoperability between systems and analytical centers.

To be purposeful, the flow must be coordinated to support the creation of a description of the tactical EW situation. Figure 3 depicts the situation necessary for the capture of EW information that will support the analysis of tactics. The combat information arrives at the force level where it is distributed to the appropriate analytic centers. There are three basic types of analysis centers: centers which provide initial assessments on deployment (physical); those which provide assessments on activity (operational); and those which provide analysis of the structural (organizational) aspects of the evolving threat. The products provided fall into these categories and form what are termed "Tactical Indicators".

The analysis tasks performed by these centers consist of associating combat information reports to each other and/or entities of record, or new entities (an entity may be a unit, group, system, net, etc.). Deriving assessments of current location and tracks; and to detect anomalies. Deriving assessment of the current composition and organization of tactical units. To process all data related to tactical command and multi element system organization and composition; and to detect anomalies. Deriving assessments of the activity of current operations of tactical units, commands, nets and weapons systems and determine the functions performed and the status and objectives of operations; and to detect anomalies. The results are stored in the distributed analysis files.

If EW tactics goals are to be achieved, all of the information required for a tactical problem must be identified and brought into a common focus for tactical analysis and decision making. For this reason we postulate the requirement for an EW data fusion system.

Basic purposes of the EW data fusion system are, first, to ensure the flow of relevant information required. Second, to create a description of the tactical EW situation. Third, to track the flow of EW events in that situation. Fourth to correlate that flow of events with a pattern of operations in the known enemy doctrine and gauge the nature of the tactics, the organization, the status of progress and the objective. Fifth to associate this pattern with a counter measures plan conceived to oppose the hostile objective and to control the flow of events of that plan.

The EW data fusion system must provide a means to task the three types of analytical centers

EW INFORMATION FLOW

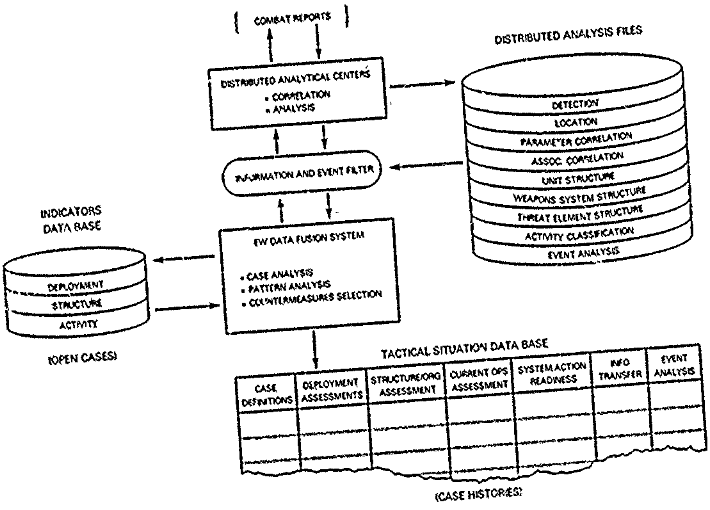


Figure 3

specifically for EW support, information about events and for the specified contents of their analysis files. The information that flows through these three channels would reside in the indicators data base organized by "case".

A case would be a problem for analysis of evidence about a tactical entity to attain a specific set of assessments. For example, a "case" could initially be to evaluate data concerning detection of platforms new to the scene and to classify same. The analytic objective then could be upgraded to determine the tactical organization to which a detected platform belongs and the type of activity in progress. And so on. Information would be saved in those indicator categories with relevance to solving the required analytic problem. Sources insofar as possible would be kept advised of that problem.

The Indicator Data Base is that set of first level inferences relating to "cases", from which second and higher level inferences may be drawn regarding the analysis of tactics. Files and codes must exist in the source data base for the case definitions and recording the possible first level inferences so automated techniques of query and aggregation may be employed. Reporting centers must be required to report in the vocabulary of these codes. In situations of uncertainty about the implication of data, it would be assigned to more than one case, or to more than one analytic possibility within a case.

Two fundamental reasons are advanced for recommending that the primary EW analytic data be organized this way. The first related to the basic differences among the three categories of data that would flow into the EW Data Fusion System source data base. Each of the categories are generated by a fundamentally different observation perspective regarding what is happening. There is essentially no correlation between them of the parameters of related observable data about the underlying tactical phenomena. They are independent and analysis of one category will tend to provide validation of the results of analysis from the other categories. For example, intercept of the modulation parameters of a missile control signal could be an alert of a missile firing about to occur. Track data of platforms in the area can independently confirm if a launch is reasonable in terms of platforms in the area and if a missile is physically present. Activity data will reveal independently the possibility that the essential pre-launch coordination has occurred. If activity data, structural data and track data do not reasonably relate, the presumption of the analysis is not correct.

The second reason for organizing input data in this way is that analysis must be purposeful i.e., case oriented. EW tactics must be committed according to a plan of action against a finite entity, hip-shooting will simply draw the enemy and the isolated actions that ensue, will provide ineffective results. A main purpose of the EW data fusion system is to perform the analysis that will allow the plan to be selected and put into effect. This is the ultimate analytic problem, and all prior analytic tasks must have laid the groundwork for that.

In essence then a case represents a "snapshot" of a tactical situation and a series of snapshots will contain tactical patterns. Open cases will reside in the Tactical Indications Data Base; closed cases will be stored in the Tactical Situation Data Base.

As this information accumulates in the Tactical Situation Data Base, it will form time-history patterns exposing the nature of change of tactical functions and indications of achievements. The nature of change and tactics yield provides a context for assessing the significance of the present status, projecting the probability of forms of activity for the future. The problem however is that these patterns are implicitly represented in the stored information. A means must be found for extracting this information and converting it to an explicit format where it can be useful.

EW INFORMATION ORGANIZATION-EXPLICIT PATTERNS

The effective organization of the information is dependent upon a proper perspective of the threat. The operations of the threat may be viewed as a continuum along the battle time line shown in the first figure. It deploys, organizes, surveys, signals, postures, targets and strikes. All of these activities are connected and would form a continuous explicit pattern if they could be seen. On the other hand tactical EW information is fragmentary. It deals with indicators -- bits and pieces of evidence. Different observers see different parts of the component activities.

In order to maintain this perspective, it is submitted that there are only a few basic tactics components and all combat tactics are made up of them. They are:

Searching and Analyzing
Monitoring and Tracking
Targeting
Strike

Upon analysis and discussion of these generic components it is found that they apply equally as well to offensive and defensive postures. In addition, there are a small number of major decisions that must be made as the battle progresses along the time line. They are. (1) decisions to be alert and maintain a monitoring and tracking posture, (2) decisions to accept the possibility of combat and to prepare for targeting; and (3) decisions to strike and engage in combat. The major thrust of EW information and collection should be to obtain evidence for making any of these decisions, or determining if they have been made by the enemy.

The dynamics of the situation is captured in a simple model shown in figure 4. The model simply tells us that as the threat progresses along the battle time line, he will provide us with the opportunity to observe and understand his activities. With the proviso, that we can place his individual actions within the context of an

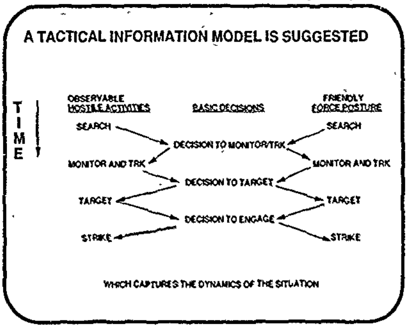


Figure 4

historical pattern of activities. If our own tactical information is arranged properly, we should be able to follow his progression step-by-step and predict his next moves, applying appropriate countermeasures to delay or thwart his intentions. The model also tells us that unless the hostile tactician reacts irrationally and, let's say, jumps directly from search into strike (as he might do if complete surprise was desired) then he will transition sequentially and predictably until he either engages or the tactic is broken off.

What then is the nature of the pattern that will allow us to follow enemy activities and to determine whether it is proper to apply a countermeasure? Figure 5 shows a generic pattern and a model of how it might work. It states that a combat operation progresses by successfully achieving a series of objectives. In the surveillance phase, for example, the objective might be to find the general location of the battle group. Once this is done the next objective might be to identify major combatants and so on. Each objective is achieved by the successful completion of individual steps or functions that lead to the achievement of the objective. Each function generates an activity which can be observed. The successful completion of a function is termed an event. An event is a measured change in the tactical situation. It could be signified by the cessation of a current activity and/or the beginning of a new one. The functions affiliated with an objective form a group. It is possible to affiliate a status indicator with each group which tells us how far along the enemy has progressed toward the achievement of the group's objective. The figure also shows that each function in the pattern also has an appropriate countermeasure.

In addition to the serial aspect, combat operations also exhibit cyclical patterns. This occurs when a decision has been made not to progress into the next phase of operations, but to either sustain the current phase or to revert back to a previous one.

The patterns we have described represent hostile activities explicitly. Unfortunately the tactical information embedded in various reports, do not directly lend themselves to this form of representation. It was shown that they must first undergo assessment and then form implicit patterns in the Tactical Situation Data Base. A data fusion technique is needed to extract the relevant implicit patterns embedded in the Tactical Situation Data Base and convert them into a dynamically changing explicit pattern.

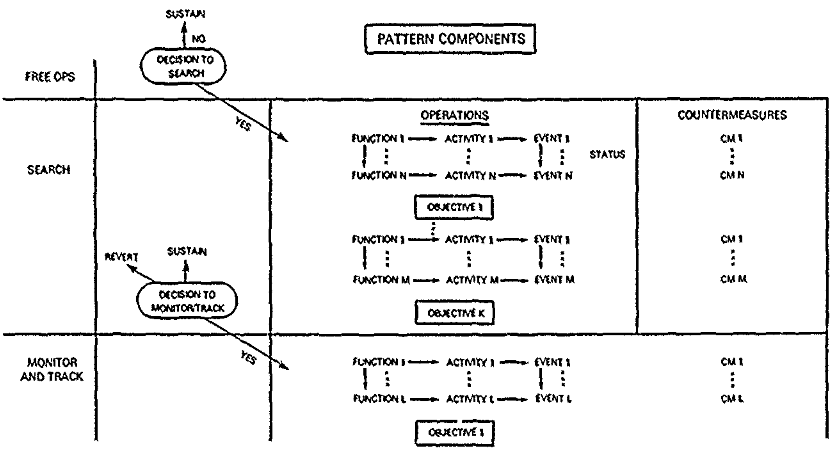


Figure 5

EW INFORMATION RELEVANCE

There are many different forms of patterns depending on the level of the threat condition. During a confrontational phase, of special interest are the few main patterns of functional activity, organization and compositions, deployment and stationing that expose the combat situation i.e., the status of attainment of the principal, combat related objectives. In a general situation involving forces and commands, there are six of these basic patterns. These are (1) the organization, resource utilization, deployment and activity to gain knowledge to enable targeting; (2) the deployment of suitable targeting units into targeting positions and dissemination of instructions for targeting to a suitable organization; (3) the accomplishment of targeting and dissemination of target assignments and coordinates by suitable targeting and command direction units; (4) movement of suitably configured units into firing strike positions; (5) the command interaction necessary to establish authority to fire/strike; and (6) employment of weapons.

Since these patterns lie buried within the Tactical Situation Data Base, it is desirable to construct modules that can operate upon this file and extract the six patterns for further processing.

According to figure 6, there are six such modules labeled "Combat Assessment Modules". The following are definitions of those modules:

a. Surveillance/Targeting Readiness Module: Organization of the recent history of evidence on surveillance and monitoring activity to enable that activity to be assessed, and in particular the attainment of a state of information readiness to position for targeting and engagement when desired, to be detected.

b. Command Direction/Engagement Posture Module. Organization of the recent history of evidence of dissemination of intelligence to combat forces, command direction of those forces, and intercommand liaison among forces and between forces and headquarters, to enable the nature of that collective activity to be assessed, and in

EXTRACTION OF RELEVANT INFORMATION

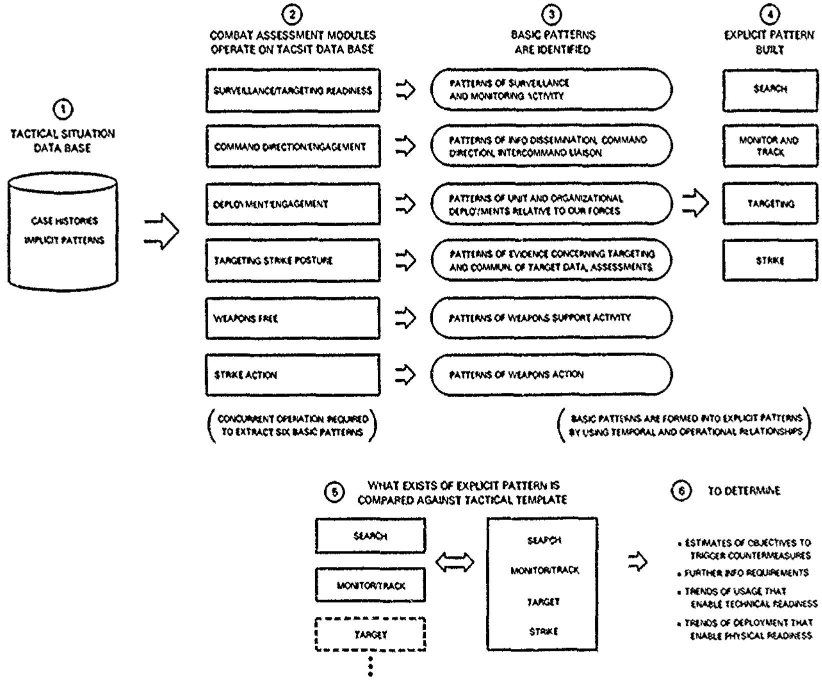


Figure 6

particular the existence of a state of requirement to target and engage, to be detected.

c. Deployment/Engagement Posture Module. Organization of the recent history of evidence concerning deployment of units and organizations, relative to our forces, to enable deployment posture to be assessed, and in particular the existence of a deployment posture meeting the requirements of targeting, or of strike for the weapons involved, to be detected.

d. Targeting/Strike Posture Module. Organization of the recent history of evidence concerning targeting and communication of target data and target assignments, to enable assessment of that activity, and in particular the achievement of an information state of readiness to strike, to be detected.

e. Weapons Free Module. Organization of the recent evidence on intercommand liaison and the overall context of activity to enable assessment of that activity, and in particular the estimation of the intent to release weapons.

f. Strike Action Module. Organization of the recent history of evidence on weapons action and weapons-support action (ECM and other combat direct support) to enable assessment of that activity and in particular the determination of the status, thrust and targets of each component of strike, to be accomplished.

Next we seek a strategy for grouping these extracted, basic patterns to form an explicit pattern that will create the best possible chance that the true situation will be apparent from the pattern that results. Figure 7 shows two ways of relating the information embedded in the six basic tactics.

The temporal relation states information collection and reporting activities will always precede commands or orders. Events will follow commands. The relation also indicates that commands are given by appropriate command elements and that tactical events are carried out by tactical elements. Conversely if we know the type of element involved, we may be able to guess both the orders given and the event that will ensue.

The operational relation allows us to guess with some efficiency what function is being prosecuted if we know what activities are in effect and what event has occurred. It also allows us to guess at the objective if we know the function and to guess the status of the functional group if we know what events have been completed.

With these rules of thumb (as well as human assistance), it may be possible to fuse the basic patterns into a dynamic explicit pattern which may be the best information we have to go on. The final step is to compose the evolving pattern against a template pattern in the known doctrine of the enemy. Once we know where we are in the tactics trajectory, we can estimate objectives, determine further information requirements, trends of usage and deployment that enable readiness.

AN EXAMPLE

Consider the example shown in figure 8. The question arises as to what is actually conveyed by the reports. Do they signify a transition from surveillance into targeting? Or do they merely indicate that a targeting resource is being used to determine new locations for continued surveillance. Since the messages are out of time sequence, how do they relate with respect to progression of tactics?

To some degree all hostile activity can be observed, but many observations are not definitive of the underlying capabilities, deployment or objectives, and in general no single observation is decisive in terms of analysis of what is really happening, and so by itself does not enable a coherent CM act. Observations generally create a fog of knowledge.

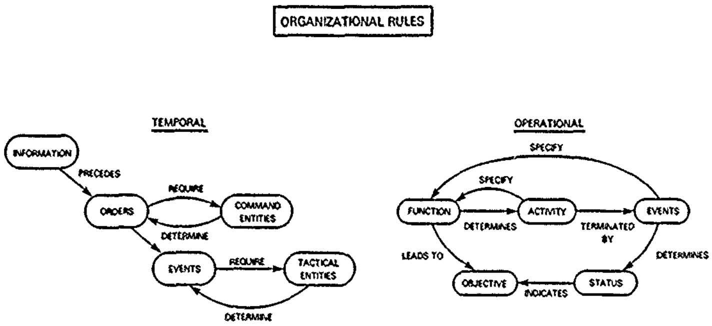


Figure 7

EXAMPLE

CONTEXT: THE BLUE FORCE HAS BEEN UNDER INTERMITTENT SURVEILLANCE FOR PAST 48 HOURS

THE FOLLOWING MESSAGES HAVE BEEN RECEIVED.

OTC NET ACTIVE, CODED COMMS FROM OTC TO CRUISERS XY AND YZ AND DESTROYERS ABC AND DEF. CODE FORMAT 123 IS USED TIME 1435Z
TARGETING RADAR ACTIVE, SINGLE SWEEP, HELO BORNE AT 1410Z, POSIT 10W 48N.
MOVEMENT OF TWO CRUISERS AND TWO DESTROYERS TOWARD BLUE FORCE AT 1500Z.

Figure 8

INDICATORS

ASSUMPTION: HELO IS BEING USED TO TARGET

ASSESSMENT	INDICATOR TYPE
• BLUE FORCE IS WITHIN RANGE OF HELO • HELO OPERATING COVERTLY	DEPLOYMENT
• HELO IS TARGETING CAPABLE SURFACE ACTION GROUP IS ASUW CAPABLE	ORGANIZATIONAL
• OTC NET NOT NORMALLY USED FOR TARGETING • CODED FORMAT 123 NOT USED FOR TARGETING	ACTIVITY
• POSITION AND TRACK DATA TRANSMITTED • TEMPO OF OPS IS LOW	

CONCLUSION REJECT TARGETING HYPOTHESIS
ASSUME ACTION SUPPORTS ATTEMPT TO MAINTAIN CONTACT

Figure 9

This fact was noted by Clausewitz [2] who stated:

"Many intelligence reports in war are contradictory; even more are false, and most are uncertain. The tactician must possess a standard of judgment, which he can gain only from knowledge of men and affairs and common sense."

Most reports that can be generated in a combat situation are essentially momentary snapshots and are very limited in scope. If we are to make the best sense possible of such reports, they must be grouped as they relate to functions in progress, and the functions, as they relate to overall tactics. Individual reports must be viewed in the context established by both groups. Having formed these groups, the information they contain needs to be interpreted to determine those elements of information needed for our own purposes. Accomplishing this requires knowledge of the basic hostile information-building process and a process for systematically analyzing an organized data base formed by these groups.

Let's show how this can be done. Starting with the case of the helo, we first create a case with the helo as the entity (Fig. 9). An assumption is made that the helo is being used to target the friendly force. Upon receipt of deployment, organizational and activity indicators from the analytical centers, assessments are made.

The deployment and organization indicators support the hypothesis that targeting of force elements is under way. The activity indicators however do not support this premise. The OTC net is not normally used for targeting operations and coded format 123 is used primarily to transmit position and track data rather than stationing or fire orders. Further, the quality of information does not suggest the high tempo of operations that would normally accompany a prelude to combat.

Conclusion: We reject the targeting hypothesis. It is assumed that the action supports an attempt to maintain contact. Case closed!

Figure 10 demonstrates how the contents of the Tactical Situation Data Base might look for the three reports after case assessments have been completed. Earlier cases are not shown nor will the data be actually encoded in the manner shown. The picture shows that there are two cases labeled "SAG-3" (which is a surface action group consisting of 2 cruisers and 2 destroyers) and that HELO-1 is grouped with them. The implicit patterns are embedded sequentially in the data fields. The event analysis history file, for example, suggests a pattern which shows that events have progressed from determining the general location of the blue force, possibly using long range sensors, to established contact with the main body.

The information quality suggests that while operations are being conducted under a cloak of security, the tempo of operations, the frequency of data refresh and the operational readiness do not suggest that an engagement is imminent.

Finally, in figure 11 the combat assessment modules operate concurrently upon the Tactical Situation Data Base to extract the basic patterns. Applying the temporal and operational rules on them allows us to create a functional entry or record into the growing explicit pattern. This pattern is then compared against a doctrinal template which is essentially a list of expected progressions. Projections, assessments, countermeasure selections, etc. ensue.

CONCLUSION

There are three objectives that this report has tried to achieve. The first of these is to take a very complex information management problem and to articulate it within the framework of a systems concept. When the problem was decomposed into the flow, organizational and relevance aspects, distinct technical as well as operational challenges emerged and were exposed. It is hoped that this will be useful in organizing efforts to attack the problem in a coherent manner.

TACTICAL SITUATION DATA BASE ENTRY
CASE HISTORY FILES

ASSESSMENTS

CASE DEF	DEPLOYMENT	STRUCT/ORG	CURRENT OPS	READINESS	INFO TRANSFER	EVENT ANALYSIS
SAG-3 ANALYTIC	TRANSITING OPSEC NO VULN	ASUW/STK 2 CG 2 DD	SEARCHING	SURV SENSORS SURV CTRL CKTS OTC NET	ENCRYPTED CMDS	GENERAL LOCATION DETERMINED
HELO-1 DETECTION	AIRBORNE COVERT	ASUW/STK HORMONE	TRACKING	TARGETING SENSOR	POS TRKS CODE 123	BG LOCATION/ORG DETERMINED
SAG-3 ANALYTIC	APPROACH OPSEC BG VULN	ASUW/STK CG XY CG YX DD ABC DD CBA	TRACKING REPORTING	OTC NET	ENCRYPTED CMDS	CONTACT ESTABLISHED WITH MAIN FORCE

INFORMATION QUALITY

OPERATIONAL TEMPO	:	LOW
SECURITY	:	STANDARD
FREQUENCY OF INFO UPDATE	:	LOW
OPERATIONAL READINESS	:	MEDIUM

NOTE: THE QUALITY OF INFORMATION GREATLY INFLUENCES THE INTERPRETATION OF THE DATA

Figure 10

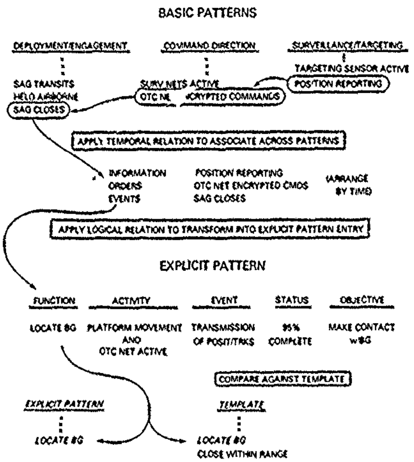


Figure 11

The second objective was to define the requirements for an EW data fusion system that would function at the force level in a proactive (continuous) manner and which would possess the generality to analyze any tactic countermeasure application. Force level EW is a relatively new Navy concept which is still evolving. There is currently no system either existing or proposed which attempts to fuse EW information to allow the mind of the commander to focus clearly on the tactical situation.

Finally we have presented the elements of a general theory of combat tactics. This has been done to provide greater insight into the complexities of Naval warfare and because if a generalized system for analysis of tactics and countermeasures against those tactics is to be described, the generalized principles of tactics themselves must be understood.

With a systems concept in place, we are ready to pursue technical solutions. Knowing of course that there may be well-presented boundaries to what technology can ultimately provide without human partnership. In the words of Clausewitz [3]:

". . . Things are perceived, of course partly by the naked eye and partly by the mind, which fills the gaps with guesswork based on learning and experience, and thus constructs a whole out of the fragments that the eye can see; but if the whole is to be vividly present to the mind, imprinted like a picture, like a map upon the brain, without fading or blurring in detail, it can only be achieved by the mental gift that we call imagination."

At the minimum, this work will attempt to provide the best available information so that imagination can be applied earlier and more effectively.

REFERENCES

1. Layman, G.E., C3CM: A Warfare Strategy, Naval War College Review, March-April 1985, p. 31-42

2. Clausewitz, On War, M. Howard and P. Paret Eds. and Trans. Princeton, N.J. Princeton Univ. Press, 1984) p. 101

3. Ibid, p. 109

ACKNOWLEDGEMENTS

We would like to thank Ms. Susan Conter and Miss Lisa Mason for their expert typing of the manuscript and their patience while this manuscript underwent revision.

Section IV

Decision Support Systems and Behavioral Aspects of C^3

Working Group Chairman:

Dr. Stanley M. Halpin
Army Research Institute

TEAM COORDINATION IN DISTRIBUTED COMMAND AND CONTROL

Daniel R. Sullivan
David L. Kleinman

Cyberlab, Department of Electrical and Systems Engineering
The University of Connecticut, U-157, Storrs, Connecticut

ABSTRACT

The scheduling of jobs by a team of distributed decisionmakers (DMs) is an important subtask of tactical decisionmaking in Command and Control. To investigate this problem an experiment was developed where three geographically seperated DMs process tasks soas to maximize a predefined objective.

The problem consists of a team of three DMs who share a pool of distributed renewable resources. These resources are used process a variety of tasks which arrive stochastically and have a fixed opportunity window or deadline. Also, there is a varying penalty for each task which is not processed before the deadline.

The goal of this research effort is to develop a normative descriptive model which will aid in the understanding of how DMs coordinate and schedule jobs under varying input conditions. Some preliminary results for the above experiment will be discussed. Also, a normative model is discussed for solving the distributed scheduling problem.

1. INTRODUCTION

How do teams adapt their decision strategies to task environments? How should a team organize and coordinate to achieve its best performance? How do teams divide responsibility amongst members? Do teams choose a leader when the situation warrants a centralized command? What are the issues that make a team decision difficult?

The study presented in this paper was motivated by the above problems in distributed tactical decisionmaking (DTDM) which arise in Naval Battle Group Command and Control (C^2). The focus of the study is team task scheduling and resource allocation in a distributed environment. To investigate this problem a test-bed was developed where three person teams schedule tasks using a pool of limited renewable resources in order to maximize a predefined goal.

The objective of this research is to develop a normative descriptive model of how teams solve a distributed scheduling problem. Experimental research has been conducted so that deviations in decision behavior from those predicted by normative models can be investigated The framework utilized in the modeling effort is a Discrete Optimization technique which was augmented slightly to encompass the distributed nature of the problem. This scheduling method matches jobs to decisionmakers (DMs) in an optimal fashion. Also, this method models the transferring of resources between DMs.

The paper is organized as follows. First an intuitive explanation of the team scheduling problem is discussed. Next, the experimental design process is outlined along with the important dependent and independent variables. This is followed by an overview of some interesting experimental results that confirm and disclaim the orinal hypotheses. The last two sections contain an explanation of the normative model and a comparison of the model results with human team data.

2. PROBLEM DEFINITION

An experimental test-bed has been developed to aid in the understanding of how human DMs solve a Team Distributed Scheduling (TDS) problem. The important issues of task scheduling in C^2 systems are incorporated in an experimental paradigm where input conditions can be varied and dependent variables measured.

The TDS paradigm can be thought of in the following manner. The three DMs represent commanders of a battle group who are geographically separated. The battle group is confronted with three types of hostile threats. Each threat type can be processed by any of one, two, or all three commanders depending on the scenario (functional overlap is explained thoroughly in experimental design section). The battle group employs a fixed number of resources for its defense. The resources can be exchanged among DMs, so that at any time a particular resource can be under the command of any of the three commanders.

Threats arrive stochastically (poisson arrivals), have strict deadlines, and require varying numbers of resources to complete. The attributes of each threat are as follows:

i) penalty - the amount subtracted from the final strength should this threat not be attacked within its opportunity window ("penetration").

ii) resources required - number of resources required to process this threat.

iii) identification number - used for communication between team members.

Therefore, the task of the commanders is to assign resources to incoming threats and to share the limited number of resources so as to maximize final survival strength. It should be noted that a single commander can only process one threat at a time. This constraint is an effort to simulate the limited capacity of a commander to oversee many events at one time. The survival strength is diminished each time a task "penetrates", i.e. the strict deadline is not met. The amount of penalty associated with threats is proportional to the number of resources required, but the relationship is not linear. Also, when a commander decides to relocate/transfer some of the resources a significant delay occurs before the resources arrive at the new commanders post.

The commanders are permitted to communicate to each other through several pre-formed messages (i.e request resources, request an attack on a given threat). However, since communication uses the valuable time of the commanders, there are occasions when communication is conciously omitted and potential conflict occurs (i.e. two commanders send out resources to process the same threat). Therefore, time is wasted and no extra reward is gained.

3.0 EXPERIMENTAL DESIGN

Since the TDS study was a model driven study (i e. several modeling alternatives were investigated before the experiment was designed although the normative model was not completed). The model was used to select many of the fixed parameters prior to the actual running of experiments. Also, several important features of the TDS experiment were determined using results of the DREAM and DDD experiments.

As noted, the goal of this experiment was to investigate how teams coordinate among team members, and how their strategies change to achieve optimal performance. A task scheduling problem was selected since it deals with many of the important facets of Naval Battle Group Command and Control. Also, a scheduling problem is a simplification of the resource allocation problem encountered in the DREAM experiment.

The team consists of three decisionmakers (DMs) who are of equal rank. This parallel structure was adopted since it is simple and allows the experimental objective to be met. Also, the three person parallel structure is a logical intermediate step between the DREAM experiment (2 person parallel structure) and future research that will involve a more complicated hierarchical team organization.

3.1 INDEPENDENT VARIABLES

Several independent variables were chosen based on their ability to affect a team coordination and strategy selection. The four input parameters which were expected to cause significant deviations in strategy and coordination were functional overlap, balance of load among team members, resource scarcity, and tempo or arrival rate of tasks.

Functional overlap, in this context, is defined as the number of DMs capable of processing a single task type. This independent variable was also implemented in the DDD and DREAM experiments. In addition, this variable was investigated by Boettcher and Levis (1981) and their work indicates the effectiveness of a team depends on the degree of functional allocation and the type of functional allocation. Here we vary the degree of functional allocation over the following three levels:

i) Sole responsibility by a single DM for a task type.

ii) Shared responsibility between two DMs for a single task type.

iii) Shared responsibility among three DMs for a single task type.

Table 1 displays how overlap of responsibility will be implemented.

DM	NO OVERLAP (NO)			PARTIAL OVERLAP(PO)			FULL OVERLAP(FO)		
	Task Type			Task Type			Task Type		
	A	B	C	A	B	C	A	B	C
1	X			X	X		X	X	X
2		X			X	X	X	X	X
3			X	X		X	X	X	X

Functional Overlap
table 1

The second independent variable is balance of workload among team members. In this case the overall arrival rate of tasks will remain constant while the arival probability of a single task type will be increased. Therefore, one DM will be busy while the other two are idle. This quantity was expected to cause teams to alter their communication rates to and from the overburdoned DM.

The third independent variable is the level of resource constraint or the number of resources owned by a team. The two levels of resource scarcity are 5 resources for the team and 7 resources owned by the team. The criterion used to select these quantities were:

i) to maintain a three person team in most of the scenario's, i e. to try to assure that all 3 team members have something to do at all times.

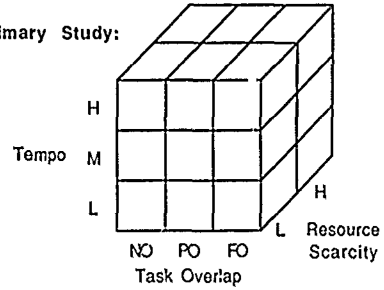
ii) to require a significant number of resource transfers in most of the scenario's.

The effects of varying this parameter were expected to be similar to varying tempo. However, varying resource scarcity in the DDD experiment resulted in larger increases in workload than those found by varying tempo. Also, this variable increases the difficulty of the problem significantly as shown in the normative model section.

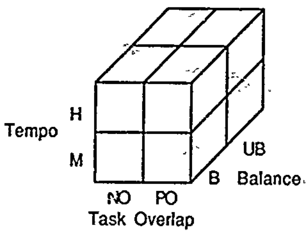
The last quantity which was varied is the load on the team as a whole. The mechanism used to vary load was tempo or arival rate of threats This method of simulating increased workload was also implemented in the DDD, DREAM, and DIS experiments. The findings of Payne, Johnson and Bettman (acceleration and filtration) were expected to be observed over three levels of load, but not neccessarily equally among all team members.

The experimental conditions tested in the TDS experiment are shown in figure 1 In the primary study data was collected for 18 conditions involving different levels of tempo, resource scarcity, and functional overlap Similarly, the secondary experiment studies the effect of imbalance of task load over different levels of tempo and functional overlap The data collected in the secondary study has yet to be be fully analyzed Three teams of three participated in this experiment and each team completed two repetitions of each of the experimental conditions shown in figure 1. Also, the duration of the scenarios was 13 minutes.

Primary Study:



Secondary Study:



Experimental Conditions
figure 1

3.2 DEPENDENT VARIABLES

The dependent variables are grouped into four functional categories: Performance, Strategy, Coordination, and Workload. The first category contains two quantities that measure the performance of a team. Final strength is the variable that teams strived to optimize. This quantity is a measure of loss over a scenario or more quantitatively the sum of the values of targets that penetrated. The second measure of performance is slack time which is defined as the time available, minus the time required to process, at the time of attack. This dependent variable provides an indication of how quickly teams were able to perform the required scheduling.

The next category of independent variables measure changes in team strategy. These variables measure how teams adapted to the experimental conditions in their attempts to achieve optimal results. The following four dependent variables were collected to compare with normative model results and thereby determine human cognitive limits and biases:

i) resource transfers
ii) number of tasks attacked by each DM
iii) number of each task type (A,B,C) attacked
iv) number of tasks attacked requiring 1,2,3 resources.

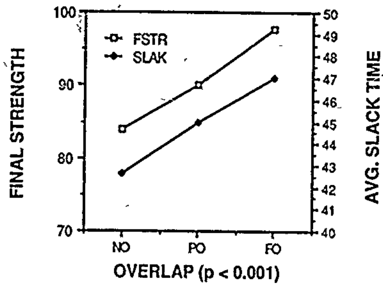
The third category of dependent variables indicate how teams coordinated during a scenario. First, resource requests give an indication of team members need for explicit communication despite centralized perfect information. Similarly, the amount of real time planning required was measured by the number of action requests ("can you please attack") and the number of action transfers ("I plan to attack"). The last dependent variable in this category measures the lack of coordination over the experimental conditions. The number of wasted attacks marks instances when two DMs attacked the same target, thereby wasting valuable time, while gaining no extra reward.

The last dependent variable category is team subjective workload assessment (SWAT). Although this variable does not fit into any of the above groups, it is a valuable measure that quantifies how teams perceived their workload over the experimental conditions.

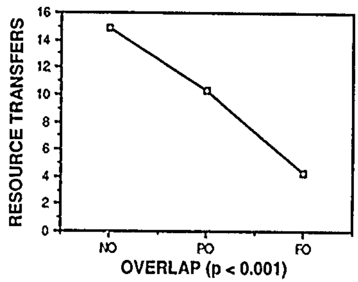
4 EXPERIMENTAL RESULTS

The experimental results continue to be analyzed at this writing, however, one independent variable forced some interesting effects which are discussed in this section. The degree of functional overlap which is defined as the distribution of task responsibility caused teams to change their strategies and alter coordination requirements over all tempos and resource scarcity levels.

An hypotheses generated prior to the experiment was that performance would increase as overlap increases, but with a concomitant need for increased coordination. This hypothesis is based on the results of Serfaty 1985. As seen in figure 2, both of the dependent variables depicting performance rise significantly ($p < 0.001$) as overlap is increased. This increase in performance is a result of the change in strategy shown in figure 3. Here teams were able to decrease the number of resource transfers required and therfore increase their timeliness and subsequent final strength.

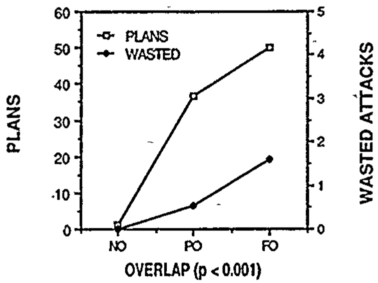


Effect of Overlap on Performance
figure 2



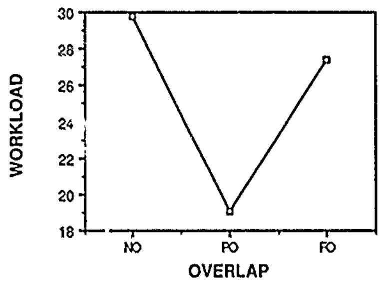
Effect of Overlap on Strategy
figure 3

Also as shown in figure 4, when overlap increases the scheduling problem requires more coordination among team members. In fact, the team cannot fully coordinate as displayed by the increase in wasted attacks. The dependent variable "plans" is defined as the sum of action requests and action transfers which were defined in section 3 2. The nonlinear increase in "plans" suggests that teams began to trade-off online communication for highly structured preplanned strategies.



Effect of Overlap on Coordination
figure 4

The graph of perceived workload (figure 5) over the levels of overlap displays an interesting result which can be explained by the elements which comprise workload. The high subjective workload in the "no overlap" condition can be attributed to frustration or psychological stress due to the lack of alternatives available to improve performance. In contrast, the high subjective workload in the "full overlap" condition is a result of the increased scheduling difficulty and subsequent coordination requirements. The partially overlapped organization provides the best trade-off between these two elements of workload.



Effect of Overlap on Workload
figure 5

6.0 NORMATIVE MODEL

The objective of this normative model is to provide a framework with which human biases and cognitive limits can be judged. Therefore, the model should lead to a normative-descriptive model that predicts human performance. The general idea of the model is to transform the scheduling problem into a weighted assignment problem and solve the assignment problem by a conventional algorithm (Hungarian Method). A simple example is used to explain the transformation from scheduling problem to weighted bipartite matching.

In this example we assume there is a single DM who owns two resources and is presently idle. At some instant in time we take a snapshot of the dynamic scenario and attempt to schedule the tasks in the picture. First, the tasks are grouped into the following three sets:

$0 <$ slack time < 30 ====> SET 1
$30 <$ slack time < 60 ====> SET 2
$60 <$ slack time < 90 ====> SET 3

Since the processing time associated with each task is constant ($T_r = 30$), the DM can schedule the tasks according to the above three sets. The fact that a DM can only be attacking one task at a time constrains the solution set. For instance, the DM can select at most one task from SET 1 because when the attack is completed 30 seconds have elapsed and the slack time of tasks in SET 1 will be negative. Similarly, the DM can select at most two tasks from SET 2 and three tasks from SET 3.

To implement the DM's task selection options, we assume the DM will attack three tasks in the next 90 seconds (1 task every T_r seconds). For the DMs first attack, any tasks in SETs 1,2,3 can be selected. Planning ahead to the next attack, the DM can only choose tasks from SETs 2 and 3. Similarly, the DM can only select tasks in SET 3 for the third attack. Figure 6 displays an example of the available matchings to a single DM who owns 2 resources. The numbers displayed along the arcs in figure 6 represent the reward gained upon attacking the associated task. Also, the task type and resources required to prosecute are located within the circles that represent tasks. The bold arcs show the optimal matching for this example.

In the second example (figure 7), there are two DMs and three types of tasks displayed. The first DM owns 2 resources and is responsible for task types A and B, while the second DM owns three resources and is responsible for task types B and C. This example is analagous to the partial overlap condition where the third DM has been omitted. The solution is not as trivial as in the first example due to the overlap of the two DMs. For instance, if the DMs do not plan ahead at least two stages a suboptimal result will occur (i.e. DMs will attack both tasks in SET 1). However, the matching algorithm is solved in a similar manner. Similarly, this modeling approach can be extended to three DMs in any resource state and overlap condition.

6.2.3 Resource Transfers

Throughout the scenario the scheduling problem is re-solved each time a DM becomes idle and tasks are available. Each time the problem is resolved for all the resource states where resource states are defined as ($R_{DM1}, R_{DM2}, R_{DM3}$). The two team resource levels are 5 and 7. Therefore, the following resource states are available depending on the scenario (table 2):

When forming the matching graphs for resource states other than the present state, the time to effect the neccessary resource transfers must be included. This resource transfer time is comprised of:

i) the time until a DM has free resources to transfer
ii) the actual resource transfer delay.

Figure 8 shows two matching problems where the first includes no resource transfer time and the second has a nonzero delay. As seen in figure 8 tasks may change SETs when solving the matching problem for resource states other than the present resource state (i.e. in alternative resource state task 1 is no longer available, task 3 moves from SET 2 to SET 1, and task 6 moves from SET 3 to SET 2). Therefore, the slack times are largest when solving for the present resource state. To transfer to another resource state diminishes the time available for all the viewable tasks.

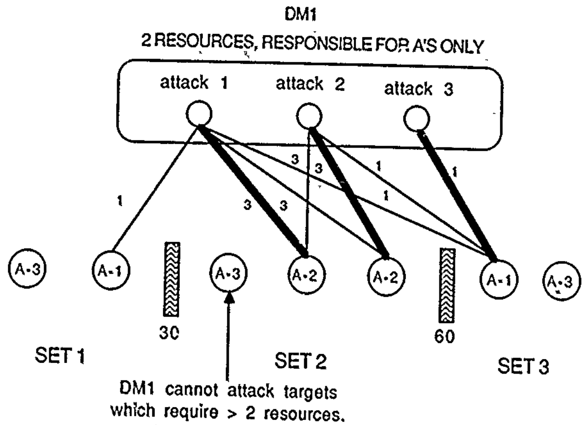


figure 6 : Matching Graph (1 DM)

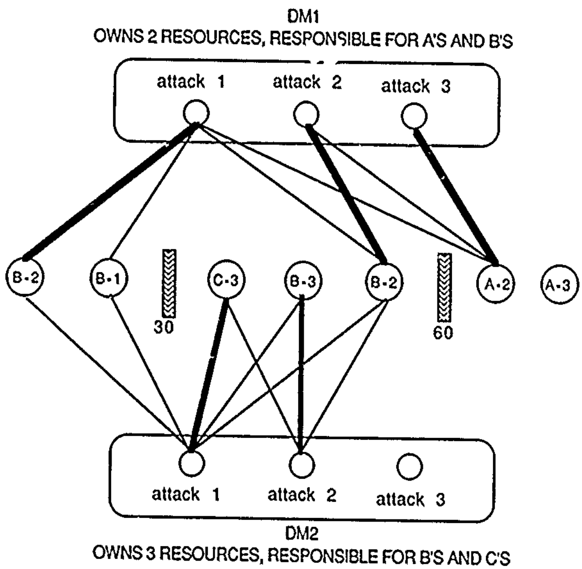
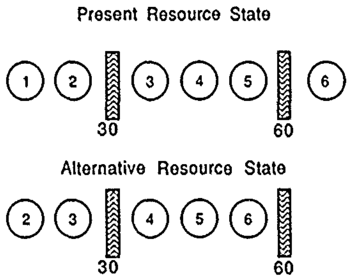


figure 7 : Matching Graph (2 DMs)

	5 RESOURCES			7 RESOURCES		
	R_{DM1}	R_{DM2}	R_{DM3}	R_{DM1}	R_{DM2}	R_{DM3}
1	0	2	3	2	2	3
2	0	3	2	2	3	2
3	2	0	3	3	2	2
4	3	0	2	3	3	1
5	2	3	0	3	1	3
6	3	2	0	1	3	3
7	1	1	3			
8	1	3	1			
9	3	1	1			
10	1	1	2			
11	1	2	1			
12	2	1	1			

Resource States
table 2

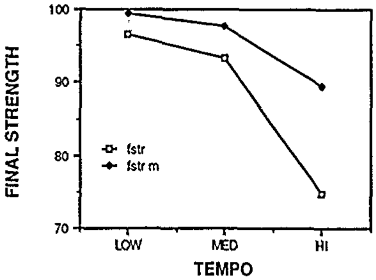


Resource Transfer Time
figure 8

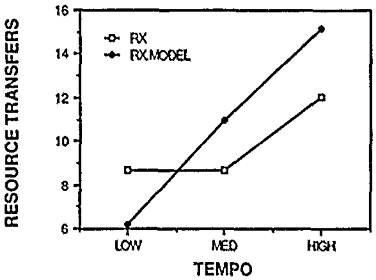
6. MODEL/DATA COMPARISON

In the first stage of the normative descriptive modeling approach, the normative model predictions are compared to the experimental results The basic hypothesis of this approach is that motivated expert decisionmakers strive for optimality, but are constrained from achieving it by inherent human perceptual limitations and cognitive biases.

An hypothesis that was generated prior to the experimental phase was that an increase in tempo would cause an increase in resource transfers and a decrease in team performance. This hypothesis is supported by the normative model in figures 9 and 10. However, the human data does not clearly follow this trend due to biases and cognitive limits. In the low tempo scenario's, it appears that humans attempted to evenly distribute load among all team members and therefore transferred more resources than necessary. Also, it is apparent that humans reduced the complexity of the problem in medium and high tempo scenarios by ruling out resource transfers This is analagous to "filtration" in Payne 1986 where humans adapt to a strategy that reduces the decision space at the expense of performance.



Normative Model Performance vs. Human Data
figure 9



Normative Model Strategy vs. Human Data
figure 10

6. CONCLUSIONS AND FUTURE WORK

Several conclusions can be drawn from the material presented in previous sections. First, the teams were able to adapt and coordinate well as more overlap of responsibility was delegated. However, an optimal amount of overlap exists with respect to workload. Also, on the basis of Serfaty's results (1985) it is believed that an optimal amount of overlap also exist for performance when the problem is sufficiently difficult

Next, teams adapted their strategies when overloaded to reduce their decision space. The strategy change presented here deals with insufficient resource transfers when the problem became difficult. In addition to this strategy change, it is believed that humans selected a different mix of tasks than the normative model and therefore could not perform as well These and other human biases and cognitive constraints must be quantified so that descriptive factors can be included in the model.

An additional conclusion drawn from the results of this experiment is that a leader is not required if the problem is sufficienlty simple. Therefore, more difficult scenarios with increased need for coordination, must be implemented to realize the benefits of a hierarchical team structure.

REFERENCES

Boettcher, K.L., and A.H. Levis, 1982, "Modeling the Interacting Decision-Maker with Bounded Rationality," IEEE Transactions on Systems, Man and Cybernetics, Vol. SMC-12, No.3, pp.334-344.

Johnson, E.J., and J.W. Payne, 1985, "Effort and Accuracy in Choice," Management Science, vol. 31, pp.395-414.

Kohn, C., 1988, "Team Distributed Resource Allocation and Management," Masters Thesis, The University of Connecticut.

Papadimitriou, C.H., and K. Steiglitz, 1982, Combinatorial Optimization: Algorithms and Complexity, Prentice-Hall, Inc., NJ.

Payne, J.W., J.R. Bettman, and E.J. Johnson, 1986, "Adaptive Strategy Selection in Decision Making," Report No. 86-1, Duke University, N.C.

Serfaty, D., 1985, "An Experimental Plan for Distributed Tactical Decisionmaking," Proceedings of the 8th MIT/ONR Workshop on C^3 Systems, Cambridge, MA., pp.65-68.

Serfaty, D., 1988, "Team Decisionmaking," PhD Dissertation to be published, The University of Connecticut.

Allocation and Distribution of 155mm Howitzer Fire

Ann E.M. Brodeen
Wendy A. Winner

Director
US Army Ballistic Research Laboratory
ATTN: SLCBR-SE-P
Aberdeen Proving Ground, MD 21005-5066
Comm (301)-278-6659/6655
AV 298-6659/6655

ABSTRACT

The U.S. Army Ballistic Research Laboratory (BRL), Aberdeen Proving Ground, MD, as well as other agencies, has been investigating the problem associated with allocating and distributing friendly fire based on the importance of an enemy target and its function in a particular tactical situation. As an alternative to standard parametric procedures, the BRL is applying recently published classification tree methodology which extends previous developments in this area.[1] Unlike other classification tree techniques, Breiman et al.'s methods concurrently handle non-standard data structures, a mixture of data types, non-homogeneous variable relationships, and different degrees of influence of the variables. These characteristics are common to the data collected by the BRL on Fire Direction Officers' decisions on 155 mm howitzer targets. An overview of Breiman et al.'s research in the context of this particular problem is presented.

I. INTRODUCTION

The Probability and Statistics Branch of the US Army Ballistic Research Laboratory's (BRL) System Engineering and Concepts Analysis Division has been in the process of analyzing two data sets with rather unusual properties. Both data sets are characterized by a mixture of data types, nonhomogeneous variable relationships, and different degrees of influence of the variables. Several well known and often used statistical approaches have been applied with the goals of uncovering the relationships among the variables and providing accurate predictions. These approaches have included multiple regression analysis, the Mann-Whitney test, Kruskal-Wallis analysis of variance by ranks, and cluster analysis. However, none of these methods concurrently handle the nonstandard data structures, mixture of data types, nonhomogenous variable relationships, and different degrees of influence of the variables. Even the combined results of these procedures have not provided an effective means of analysis, and it is expected that some information has been lost.

As an alternative to these standard parametric procedures, the BRL is investigating employing a recently published classification tree methodology to these data sets.[1] Similar to other published classification tree methodologies, Breiman et al.'s methodology provides predictions by constructing binary trees. However, unlike other analytical techniques, Breiman et al.'s classification tree structured methods concurrently handle nonstandard data structures, a mixture of data types, nonhomogeneous variable relationships, and different degrees of influence of the variables.

This paper will briefly describe the standard parametric and nonparametric procedures that were applied to the data sets, and their associated deficiencies. Following this discussion, Breiman et al.'s classification tree structured methods and the advantage of this type of analysis will be briefly described. Since work is currently being undertaken to apply this classification tree structure methodology to the two available data sets, the methodology will be applied to a less complex data set. A subsequent report, however, will be published to describe the results of this research.

II. BACKGROUND

A. Data Sets

In December 1985, the BRL conducted a controlled laboratory experiment, the Firepower Control Experiment, at the joint US Army Human Engineering Laboratory and BRL Command Post Exercise Research Facility.[2] As part of this statistically designed experiment, information was collected on Fire Direction Officers' (FDOs') decisions on a variety of targets being forwarded to 155mm howitzer units.* FDOs' decisions include selecting the type, volume, and method of firing ammunition on enemy targets by a specific 155mm

---

[1] Breiman, L., Friedman, J., Olshen, R., and Stone, C., Classification and Regression Trees, Belmont, California, Wadsworth, Inc., 1984.

[2] Winner, Wendy A., Brodeen, Ann E.M., and Smith, Jill H., *Test Design and Analysis. Firepower Control Experiment Part 12 of 12*, U.S. Army Ballistic Research Laboratory Memorandum Report, BRL-MR-3612, June 1987

* Tactical Fire Direction and gunnery instructors from the US Army Field Artillery School, Fort Sill, OK, participated as FDOs.

howitzer firing configuration, i.e., the allocation and distribution of friendly fire. This data set comprises 3,219 tactical fire control decisions collected for different FDOs, target types/subtypes, target sizes, types of fire mission control (i.e., "method of engagement") and initial ammunition basic loads.

As part of the BRL's research in tactical computer science, several unclassified scenarios between friendly and enemy forces in the Fulda Gap have been developed under a BRL contract with LB&M Associates, Inc., Lawton, OK. Embedded within these scenarios are decisions on allocating and distributing 155mm howitzer fire on independent targets observed in one-hour periods. To date, information associated with 522 tactical fire control decisions has been extracted from a portion of these scenarios.

Figure 1 summarizes the type of information available for the decisions in these data sets. A combination of categorical and numerical variables describes the principal factors (FDO through ammunition available) thought to influence the FDO's decision process as well as the actual decision (allocation method through type of second munition fired). Based on the results of previous data analyses, it is anticipated that these variables have different degrees of influence and exhibit nonhomogeneity.

B. Parametric and Nonparametric Procedures Applied

1. Multiple Regression Analysis

Multiple regression analysis is an analytical methodology that usually has one of the following primary goals: 1) predict the value of the dependent variable for new values of the independent variables, 2) screen variables to detect each variable's degree of importance in explaining the variation in response, 3) specify the functional form of the model, or 4) provide estimates of each coefficient's magnitude and sign.[3] By applying multiple regression analysis to the data from the Firepower Control Experiment, it was hoped that a regression equation could be derived to suitably predict the allocation method. Using a combination of indicator factors for the categorical variables (e.g., FDO and target type/subtype) and untransformed values for the numerical variables (e.g., ammunition load expressed as a percentage of a basic load, target size, and the method of engagement), stepwise and "best subset" regressions were run to predict the response factor (e.g., the allocation method).

Stepwise regression was run to insert factors into the regression equation based on their partial correlation coefficient with the response factor.[4] At each step, the partial F criterion of each regressor already in the equation was compared to the appropriate tabled F value. The regressor was either retained in the equation or rejected based on whether the test was significant or not. Stepping continued until none of the regressors could be removed, and none of the other potential regressors could be inserted due to the value of their partial correlation coefficient. "Best subset" regression was then run on the stepwise regressor variables to determine the best overall subset out of all possible regressions according to the maximum R^2 criterion.

As a consequence of performing a least squares fit of the data, fitted equations were obtained for the allocation method. However, based on the proportion of variance accounted for by the regressors in the regression equations, none of the factors clearly influenced the allocation method. This suggests that other factors not taken into account may influence FDOs' decisions on an allocation method.

2. Mann-Whitney Test

One of the objectives of the experiment was to test whether the amount of available ammunition affected the number of rounds the FDO elected to fire on a target. Prior to comparing all FDOs within a given ammunition basic load or comparing an individual FDO across the three ammunition basic loads, it was desirable to examine whether or not it would be necessary to distinguish between the adjust fire (AF) and fire-for-effect (FFE) methods of engagement. Since the distribution of total rounds fired against a target is not known for the two employed methods of engaging a target, the nonparametric Mann-Whitney test was used to test whether the two independent random samples could have been drawn from two populations having similar distribution functions. Based on the results of the Mann-Whitney test, the samples associated with the two methods of engagement could not be grouped together for other statistical tests.

3. Kruskal-Wallis Test

Similar to the Mann-Whitney test, the nonparametric Kruskal-Wallis one-factor analysis of variance by ranks procedure was used to examine, *first*, the mean number of rounds fired within each of the three different ammunition basic loads by each FDO, and, *second*, the mean number of rounds fired by each of the

[3] Myers, Raymond H., *Classical and Modern Regression with Applications*, Boston, Massachusetts: Duxbury Press, 1986.

[4] Draper, Norman R., and Smith, Harry, *Applied Regression Analysis*, New York, New York: John Wiley & Sons, Inc., 1981.

three FDOs within a given ammunition basic load.[5] It was concluded from the test that there were significant differences within an ammunition basic load in the mean number of rounds fired by each FDO against an individual target. In addition, test results showed that only one of the FDOs tended to fire on average more rounds against a target under at least one of the ammunition basic loads than under at least one of the other basic loads. For the random samples resulting in rejection of the null hypothesis, viz., no difference in the mean rounds fired against a single target, additional pairwise Kruskal-Wallis tests were performed.

4. Cluster Analysis

Cluster analysis was employed to categorize targets according to their importance based on their contribution to an enemy force in a particular tactical situation, i.e., their target value.[6,7] There are several ways to measure the value of the target. For example, one way could be to use several variables to measure the description, location, and activity of the target. A description of the target might include its type/subtype, size, and degree of protection. The location of the target might consider the actual grid location of the target, the altitude of the target, and the distance between the target and specific friendly units. The activity of the target might take into account its speed and direction of movement.

Cluster analysis provided a multivariate statistical method to examine the interrelationships between the target description, the FDOs, and the initial ammunition load expressed as a percentage of a basic load. Target value was based on the mean number of rounds expended against an individual target. Targets were categorized into three target value clusters, i.e., "low", "fair", or "high", based on the minimization of the Euclidean distance between each target and the mean of the targets in the cluster.

C. Deficiencies Among the Analyses

Despite the fact that each of these statistical procedures is well known and used, they have several shortcomings with regard to the problems inherent to the Firepower Control Experiment data set. For instance, these methods do not concurrently handle nonstandard data structures, mixtures of data types, nonhomogeneous variable relationships, and different degrees of influence of the variables. Subsequently, it is expected some information has been lost.

Thus, the combined results of these procedures do not provide an effective means of analyzing the experimental results concerning the allocation and distribution of 155mm howitzer fire for enemy targets. For instance, cluster analysis provides a coarse evaluation of a target's value based on the initial ammunition load, its type/subtype, and FDO. The "best subset" multiple regression equations provide only weak relationships between the FDO, allocation method, target type, target size, method of engagement, and initial ammunition load. Thus, the question remains, "Is this a result of variables measured in the experiment or a consequence that these procedures could only be focused on limited subsets of the data collected?" Subsequently, a search for a different means of analyzing this data has been undertaken.

III. CLASSIFICATION TREE METHODOLOGY

A. Background

Trees, whether known as decision trees, binary trees, or by some other name, have been used by data analysts as an informative nonparametric tool for investigating various types of data sets. Tree classification methods use the data to form prediction rules for a response variable based on the values of independent variables. Specifically, measurements are made on some object, and a prediction rule is then used to decide to what class the object belongs. This methodology is so simple that it is often passed over in favor of other methods which are thought to be more accurate, such as discriminant analysis.

Recent developments in the area of structured classification trees, which have been published by Breiman et al., are aimed at strengthening and extending the original tree methodology. Their advancements have been incorporated into a statistical software package known as CART™ (Classification and Regression Trees). Given complex data sets with many independent variables, the developers of CART believe that the structured trees produced by CART can have error rates significantly lower than those produced by the usual parametric techniques. These procedures are robust, i.e., they minimize the effects that data outliers might produce.

The advancements made in the area of structured tree methodology are significant enough to warrant investigation and application to the problems of allocating and distributing 155mm howitzer fire.

B. Overview of the CART Methodology

1. Definitions

Many of the statistical techniques presently available are designed for small data sets having a standard data structure. By a standard data structure we mean that

[5] Conover, W.J., *Practical Nonparametric Statistics*, New York, New York: John Wiley & Sons, Inc., 1976.

[6] Romesburg, H. Charles, *Cluster Analysis for Researchers*, London. Lifetime Learning Publications, 1984.

[7] FC6-20-2, "Targeting and Target Value Analysis," coordinating draft, Fort Sill, OK. US Army Field Artillery School, October 1984.

there are no missing values among the measurements made on an object, or so few they may be estimated prior to analyzing the data. In addition, the variables all have to be of the same type, i.e., all numerical or all categorical. The underlying assumption of the data is that the driving phenomenon is homogeneous, i.e., the *same relationship* holds over the entire set of measurements made on the object in question.

The data available to study the problem of allocating and distributing friendly fire on enemy targets does not meet the above criteria. In both data sets, values for several of the measurements used to describe an enemy target may be missing or must be assumed not available for any number of reasons. The variable list comprising a target's description (location, activity, description, etc.) is a mixture of numerical and categorical variable types. Finally, we cannot realistically expect the same relationships to hold amongst the wide range of measurements made on a target.

2. Constructing a Classification Tree

To construct a structured tree, four elements are needed: 1) a set of binary questions of the form: Is $x \in A$?, $A \subset X$, where x is the measurement vector defining the measurements $(x_1, x_2, ...)$ made on a case, and X is defined as the measurement space containing all possible measurements, 2) a goodness of split criterion that can numerically evaluate any split of any node of the tree, 3) a rule which dictates when to continue splitting the node or to declare it a terminal node, and 4) a rule for assigning every terminal node to a class. The set of binary questions generates a set of splits of every node. Those cases answering "yes" go to a left descendant node, while those answering "no" go to a right descendant node.

3. Features and Advantages

Breiman et al.'s methodology for classification trees appears to be a powerful and flexible analytical tool. Some of its major features and advantages over other methods will be very briefly outlined.

One of the more important aspects of the CART methodology is its ability to automatically handle missing values while minimizing the loss of information. This is achieved via the concept of surrogate splitting.

To understand surrogate splitting, two splits are said to be associated at a node if either of two conditions exists. If most of the cases are sent to the left or to the right by one split, and the other split also sends most of the cases in the same direction, the two splits are said to be strongly associated. The splits are also associated when one split sends most of the cases to the left (right) while the other split sends most of the cases to the right (left). The missing value algorithm then proceeds as follows. The CART methodology is designed to initially search through all possible splits on a given node and select the best split. For example, suppose the best initial split is: Is $x(5) > 34.1$?. All other variables except $x(5)$ will then be searched until the split on each variable which is most closely associated with the split on $x(5)$ is found. This series of splits might result in a list such as the following:

$x(2) > 26.2$ is the most closely associated with $x(5) > 34.1$,

$x(11) > 50.6$ is the second most closely associated with $x(5) > 34.1$,

and so forth. These splits are the surrogate splits for the initial split: Is $x(5) > 34.1$?.

If a case has a missing value of $x(5)$ so that the best split is not defined for that case, CART then looks at all nonmissing variables in that case and finds the one having the highest measure of predictive association with the best split. In this example, CART would first look at the most closely associated surrogate split. For example, if the value of $x(2)$ is not missing, then the case would go left if $x(2) > 26.2$ and right otherwise.

This procedure is analogous to the one used to estimate the missing values in a linear model (viz., regression on the nonmissing value most highly correlated with the missing value). However, the CART missing value algorithm is more robust. The cases with missing values in the selected splitting variable do not determine which direction the other cases will take. Given further splitting, cases sent in the wrong direction due to the missing value algorithm may still be classified correctly.

Since variables do not act alone when predicting a classification, it is natural to question which variables played the role of predictors. In the construction of a tree there may be instances in which some of the variables are never used to split any node; however, this does not necessarily mean these variables lack any predictive information. Therefore, each variable is assigned a measure of importance which may be helpful to the analyst in uncovering variables otherwise glossed over when looking at only the splits from the final selected tree. One note should be made. Like many variable ranking procedures, this one is a bit subjective and the exact numerical values should not be interpreted precisely.

Other features which do not require such an in-depth discussion are the following: 1) ability to handle both numerical and categorical variables in a natural and simple fashion, 2) application to any type of data structure through the formulation of an appropriate set of binary questions, 3) a variable selection process closely resembling a stepwise procedure since a search is made at each intermediate node for the most significant split, and 4) in the overall measurement space X, the trees suggest the property of robustness, while within the learning set the method is not appreciably affected by several misclassified points.

C. Digit Recognition Example Using the CART Methodology

The following digit recognition example was constructed by the authors of CART and illustrates the various parts** of the classification portion of the methodology.

Most of us are familiar with electronic calculators which ordinarily represent the digits 1, ..., 9, and 0 using seven horizontal and vertical lights in specific on-off combinations. If the lights are numbered as shown in Figure 2, then i denotes the ith digit, $i = 1, 2, ..., 9$, and 0, and the measurement vector $(x_{i1}, ..., x_{i7})$ is a seven-dimensional vector of zeros and ones. Let $x_{im} = 1$ if the light in the mth position is "on" for the ith digit, otherwise $x_{im} = 0$. Table 1 presents the possible values of x_{im}.

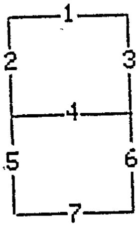


Figure 2. Horizontal and Vertical Lights.

Set the number of classes $C = \{1, ..., 10\}$ and let the measurement space X contain all possible 7-tuples of zeros and ones.

Suppose the data for this problem are generated from a faulty calculator for which it is known that each of the seven lights has the probability of 0.1 of not functioning properly. The data consist of outcomes from the random vector $(X_1, \ldots, X_7, Y)$ where Y is the class label and assumes the values 1, .., 10 with equal probability and, as noted previously, the $X_1, ..., X_7$ are zero-one variables. Given Y, the $X_1, ..., X_7$ are independently equal to the value corresponding to Y in Table 1 with probability 0.9 and are in error with probability 0.1.

Table 1. Possible Values of x_{im}.

Digit	x_1	x_2	x_3	x_4	x_5	x_6	x_7	y
1	0	0	1	0	0	1	0	1
2	1	0	1	1	1	0	1	2
3	1	0	1	1	0	1	1	3
4	0	1	1	1	0	1	0	4
5	1	1	0	1	0	1	1	5
6	1	1	0	1	1	1	1	6
7	1	0	1	0	0	1	0	7
8	1	1	1	1	1	1	1	8
9	1	1	1	1	0	1	1	9
0	1	1	1	0	1	1	1	10

The learning sample L is comprised of two hundred samples which are generated using the above distribution. Each sample in L is of the general form $(x_1, .., x_7, j)$ where $j \in C$ is the class label and the measurement vector $x_1, ..., x_7$ consists of zeros and ones.

As mentioned in Section III.B.2., to apply the CART structured classification construction on L, four things must be specified: 1) the set of questions, 2) a rule for selecting the best split, 3) a criterion for choosing the right-sized tree, 4) a rule for assigning every terminal node to a class. Here the question set consisted of the seven questions: Is $x_m = 0$?, where $m = 1, ..., 7$. The Gini index of diversity rule was used to select the best split. The concept of this splitting criterion depends on a node impurity measure. Given a node n with estimated class probabilities $p(j \mid n), j = 1, ..., J$, and the probability that given a randomly selected case of unknown class falls into node n it is classified as class i, define a measure $i(n)$ of the impurity of the given node n as a nonnegative function ϕ of the $p(1 \mid n), ..., p(J \mid n)$. Subsequently, the Gini index of diversity takes the form: $i(n) = \sum_{j \neq i} p(j \mid n) p(i \mid n)$. This node impurity is largest when all classes are equally mixed together in the node and smallest when the node contains only one class. A search is made for the split that most reduces the node, and consequently tree, impurity. The V-fold cross-validation method was used to "prune" to the right-sized tree. Here the original learning sample L was divided by random selection into V subsets $L_v, v = 1, ..., V$, of nearly equal size. The vth learning sample is: $L^{(v)} = L - L_v, v = 1, ..., V$, where $L^{(v)}$ contains approximately $(V-1)/V$ of the total data cases. Assume that a classifier can be constructed using any learning sample. Then, for every v, apply the classification procedure and let $d^{(v)}(x)$ be the resulting classifier. Since none of the cases in L_v was used to construct $d^{(v)}$, a sample estimate of the overall tree misclassification rate may be calculated, and a classifier is now constructed using the entire original learning

** It should be pointed out here that while this is the same example as outlined by the authors in their textbook, the output they produced for the purpose of illustration was not generated by the learning sample data presented in the text. Padraic Neville, who has been assisting the authors with the software management, has stated that the original data used to run this example was accidentally lost; however, the data in the text nearly depicts the original data. Therefore, the final structured tree presented in this paper will differ from that presented in the text.

sample L. A suggested assignment rule is to classify a terminal node n as that class for which $N_j(n)$ is largest, where $N_j(n)$ is the number of class j observations in n.

The resulting classification tree is shown in Figure 3.[†] The question leading to a split is indicated directly underneath each intermediate node. If the question is answered affirmatively, the split is to the left; if it is answered negatively, the split is to the right. Note that there are 11 terminal nodes, each corresponding to at least one class with class 3 having a second terminal node. Generally speaking, a one-to-one correspondence occurs by accident since any number of terminal nodes may correspond to a particular class, or some classes may have no corresponding terminal nodes. It also should be noted that the terminal node class is determined by the most predominant digit that falls into the terminal node.

The overall probability $R^*(l)$ of misclassifying a new sample given the constructed classifier (and the above fixed learning sample) was estimated as 0.31. Two other estimates of $R^*(l)$ were also computed: 1) the cross-validation estimate, and 2) the resubstitution estimate. Since the learning sample L must be used in actual problems to construct both the classifier and to estimate $R^*(l)$, these estimates are referred to as internal estimates. The cross-validation estimate was estimated as 0.32 - satisfactorily close to $R^*(l)$. The resubstitution estimate was also calculated to be 0.32. This particular estimate identifies the proportion of cases from the learning sample which is misclassified once the set is run through the constructed classifier. Using the V-fold cross-validation method explained earlier, such estimators come satisfactorily close to $R^*(l)$.

[†] The notation used here to describe the classification tree differs from that of the text.

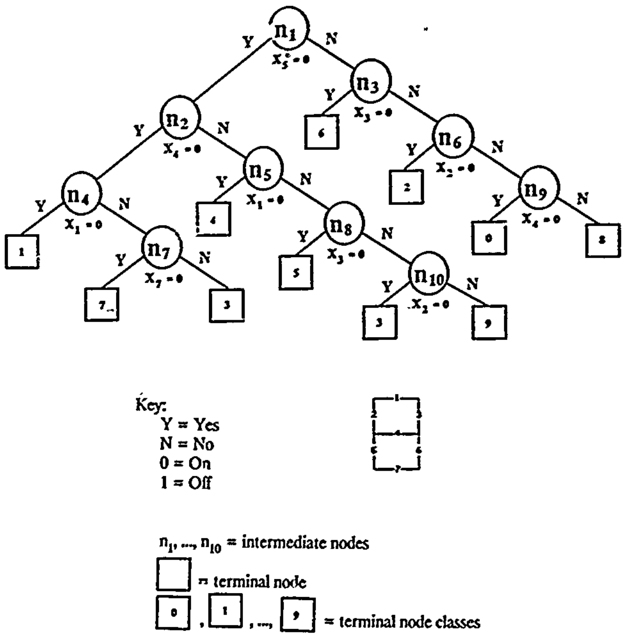


Figure 3. Digit Recognition Classification Tree.

IV. CONCLUSIONS

The classification tree structured methodology developed by Breiman et al. currently seems to be a viable approach to analyzing the available data sets. Although the regression tree portion of Breiman et al.'s methodology has not been examined in detail, it also may be another means of analyzing this data. In the case of the data from the Firepower Control Experiment, it should be interesting to compare the results of the multiple regression analysis, Mann-Whitney test, Kruskal-Wallis tests, and cluster analysis to the CART results. Work is currently being undertaken to apply the classification tree structure methodology developed by Breiman et al. to the two available data sets. A subsequent report will describe the results of these investigations.

V. ACKNOWLEDGEMENT

The authors wish to express their appreciation to Mr. Eric G. Heilman of the US Army Ballistic Research Laboratory for presenting our paper at this year's Symposium.

REFERENCES

[1] Breiman, L., Friedman, J., Olshen, R., and Stone, C., *Classification and Regression Trees*, Belmont, California: Wadsworth, Inc., 1984.

[2] Winner, Wendy A., Brodeen, Ann E.M., and Smith, Jill H., *Test Design and Analysis: Firepower Control Experiment Part 12 of 12*, U.S. Army Ballistic Research Laboratory Memorandum Report, BRL-MR-3612, June 1987.

[3] Myers, Raymond H., *Classical and Modern Regression with Applications*, Boston, Massachusetts: Duxbury Press, 1986.

[4] Draper, Norman R., and Smith, Harry, *Applied Regression Analysis*, New York, New York: John Wiley & Sons, Inc., 1981.

[5] Conover, W.J., *Practical Nonparametric Statistics*, New York, New York: John Wiley & Sons, Inc., 1976.

[6] Romesburg, H. Charles, *Cluster Analysis for Researchers*, London: Lifetime Learning Publications, 1984.

[7] FC6-20-2, "Targeting and Target Value Analysis," coordinating draft, Fort Sill, OK: US Army Field Artillery School, October 1984.

HUMAN DECISIONS IN AN SDI COMMAND AND CONTROL SYSTEM: A PETRI NET MODEL*

Jean MacMillan
Joseph G. Wohl

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

ABSTRACT

This study identifies some of the ways in which human beings can make a contribution to the effectiveness of a Strategic Defense Initiative (SDI) system. The study concentrates on key human roles in the SDI command center during the peacetime and transition phases of system operations, focusing on the critical task of situation assessment. The study identifies the functions that support situation assessment decision making, and analyzes the flow of information among functions leading up to a decision.

This network of interdependent functions was captured in a Stochastic, Timed, Attributed Petri Net (STAPN) model of the critical information flows supporting situation assessment in the command center. The function of the model was to allow sensitivity analysis of command center reaction times. The model produces estimates of the time required to reach a decision after an event has occurred under different assumptions about the organization of the command center, the architecture of the command and control system, and the delegation of authority.

INTRODUCTION

The Strategic Defense Initiative (SDI) program is a technologically complex and ambitious undertaking. Identifying the appropriate role for human decision making in the resulting system architecture is a challenging task. Although there has been a consensus from the beginning of the program that human involvement in strategic defense decisions is essential, there are significant practical limits on the role that can be played by humans. The issue is how to define and support the role of human decision making, given the characteristics and limitations of human information processing capabilities. If meaningful human roles are to be established in an SDI system, early incorporation of human factors analysis in system design is indispensable.

The purpose of this study was to identify ways in which humans can best make a contribution to the effectiveness of an SDI system, based on analysis of potential human roles and activities in system operations. Our goal was to identify requirements and objectives for human performance, and to trace the flow of information supporting human functions within the system. This study focuses on human roles in the SDI Command Center during the peacetime and transition phases of system operations. The command center is the place where the most critical human activities will take place in an SDI system. The peacetime and transition phase was selected for study because, even though peacetime activities will form the vast majority of all SDI activities, very little analysis has been devoted to this phase.

The key roles for humans in the command center involve authority, responsibility, accountability, and coordination. Human beings must exercise authority over the system. They must take responsibility for system status and readiness, and for the validity of system inputs. They must also be accountable to higher authority for all actions taken. Finally, they must coordinate with each other in order to carry out the system objectives.

Our task was to analyze peacetime and transition functions for selected SDI subsystems. We have chosen to focus our analysis on situation assessment functions in the command center. Situation assessment involves the weighting and fusing of many different types of information, including operational information about the functioning of the SDI system and intelligence information about the state of the world, into an overall picture. Situation assessment draws on the unique capabilities of human beings to merge and interpret information, and provides a fertile area for investigating effective human roles in SDI.

The analysis of control functions in complex systems, and the allocation of system functions to humans and machines, have been topics of considerable interest in recent years. A methodology for functional allocation has been proposed for nuclear power plants (Pulliam et al., 1983) and generalized for application to aerospace systems (Pulliam and Price, 1985). The first stage in this methodology is the identification of those functions for which automation is mandatory, and those functions that *must* be performed by humans. A series of considerations and suggestions is then given for allocating those functions that fall into the grey area between mandatory automation and mandatory human responsibility.

Our analysis concentrates on the first step of this allocation process by identifying the key decisions that must be made by human beings. We then consider the support requirements for these decisions. What must the decision maker know in order to make the decision, and what

* This work was sponsored by ESD/AT through AAMRL/HED.

are the sources of this information? The analysis drew on published materials about SDI command and control functions (SDI BM/C3 Working Group for Standards, July 1987), as well as SDI architecture reports prepared by contractors. (A list of SDI architecture reports reviewed for the project is included in the references section.) We also analyzed existing system architectures for systems with some similarity to SDI, especially NORAD architectures. (NORAD reports reviewed are listed in the references section.)

SITUATION ASSESSMENT FUNCTIONS

Situation assessment is the process of integrating information from multiple sources to form a coherent picture. The SDI command center must assess both its own internal situation and the external world situation, and combine them into a total situation assessment that supports critical decisions.

The data for internal situation assessment come from monitoring the SDI system. This monitoring produces detailed data on system functioning over time, and is the first step in identifying any problems in sensor capability Based on this health and status data, the command center must assess current and projected sensor capabilities, identify adjustments and fixes needed, estimate the time needed to make these changes, and make a set of critical decisions.

The data for external situation assessment come from a variety of intelligence sources as well as from higher authority (e.g., DEFCON level). External situation assessment is the process of fusing data from multiple sources into an overall picture of the current world situation. As part of this process, indicators and warnings are generated that reflect world tension levels and enemy activities. Another activity of external situation assessment is the estimation of possible time lines for enemy actions. Based on intelligence data, what is the earliest time that an attack might occur?

Assessments of the internal and external situation must be combined with information about system readiness requirements to make an overall assessment of the situation. Any problems with sensor capability must be evaluated in the context of world events. Even minor problems with the surveillance system may be of great concern if DEFCON level is high, and certain critical indicators and warnings are present. The time needed to adjust or fix sensors must be evaluated in comparison to the estimated time of earliest enemy actions. For example, sensor problems that can be adjusted or repaired before the earliest estimated time of an enemy attack are not as serious as problems that cannot be fixed within the estimated time available. Based on the total internal and external situation, the command center must evaluate the adjustments that can be achieved within the estimated time available and to determine the resulting projected defense capabilities.

Five critical decisions that must be made by humans in the command center during peacetime and transition were identified, based on the total situation assessment:

Adjust sensor sensitivity thresholds. The sensitivity thresholds of the sensors may be changed so that they react to lower or higher levels of infrared radiation. Higher sensitivity levels are associated with a higher probability of detecting targets, but at the cost of a higher probability of false alarms. Lower sensitivity levels are associated with a lower probability of false alarms, but a lower probability of detection.

Initiate surveillance sensor cool down. Sensors designed to detect and track targets during their midcourse phase must be able to sense the infrared radiation of a target against a cold background. These sensors must be cooled down to temperatures below the temperature of their satellite environment through stored cryogen, refrigeration, or some other cooling method. This cooling process takes time, so that sensor cool down must be initiated before sensor function is needed. However, cooling capacity is limited, so cooling down the sensors shortens their useful lifetime This may represent an important "false alarm" penalty.

Enable weapons (as authorized by higher authority) Fire control systems must be enabled before they are needed, and this process will not be instantaneous (Note that the sensors guiding the weapons must be cooled down in the same way as the surveillance sensors.)

Recommend a change in DEFCON level. Although the SDI Command Center will not have the authority to change the DEFCON level, it may make recommendations to higher authority about such changes and provide supporting information.

Recommend or select Rules of Engagement (ROEs) options Rules of engagement specify actions to be taken in response to different situations. Based on their situation assessment, the SDI command center may decide to recommend a change to the ROE option currently in effect ROE changes may alternatively be selected within the command center, depending on operating doctrine yet to be determined.

Of all the activities to be carried out in the command center, these five require drawing conclusions about the meaning of information and making decisions based on those conclusions. Total situation assessment requires the complex integration and interpretation of data from multiple sources, much of the data will not be in precisely defined quantitative form. This is an area in which human performance is still consistently superior to machine performance. Finally and most importantly, the decisions based on that assessment must be made by human decision makers because humans must take final responsibility for the system's actions. Thus, there is an essential human role in those command center functions.

FLOW OF INFORMATION AMONG FUNCTIONS

Figure 1 shows the flow of information among the functions that support situation assessment Some major themes emerge from the analysis of information flow in

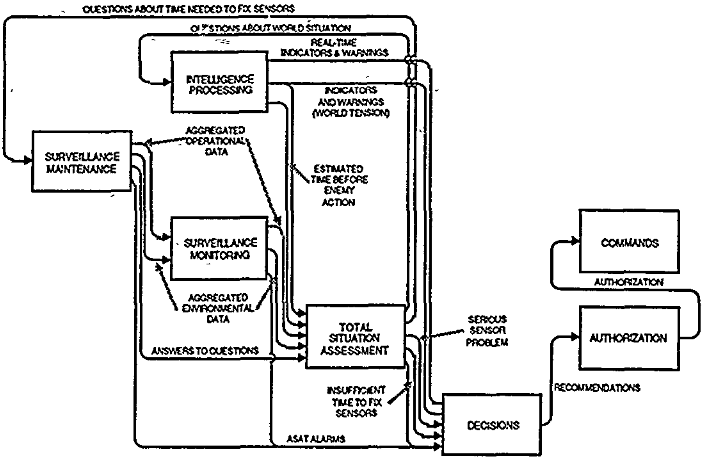


Figure 1. Overview of Information Flow in Situation Assessment

situation assessment. The first is the idea that data are converted into information through successive analysis and aggregation. Data on systems operation are processed at successive levels of detail, starting with the detailed raw data about the individual operations of every sensor in the system, and leading up to a global overview of system status based on aggregated data.

The functions involved in this process progress from maintaining the system, at the lowest level of aggregation, to monitoring system operations, at a higher level of aggregation, to assessing system status, at the highest level of aggregation. The assumption is made that data are analyzed, aggregated and passed upward at each successive level For example, a person at the lowest level might see extensive data on an individual sensor, while a person at the highest level would see aggregated data for all sensors in the system.

Evidence of enemy ASAT action might be detected at any point in this aggregation process. An enemy action might be noticed at the lowest level for an individual sensor, or it might not be detected until the data were aggregated and a global pattern could be seen. At the point that an ASAT action is noticed, we have assumed that it would be reported directly to the decision function at the command center.

External situation assessment takes place under the intelligence processing component, which produces real-time indicators and warnings that are passed directly to the decision component, and more traditional indicators and warnings that are passed to both the situation assessment component and the decision component.

Information about both the internal and external situation come together in the total situation assessment component. Questions about the world situation are referred back to intelligence processing, and questions about the time required to fix any sensor problems are referred back to surveillance maintenance. The total situation assessment component of the model passes on any sensor problem judged to be serious (when evaluated in the context of the world situation) to the decision component. If there is insufficient time to fix the sensors before the earliest estimated enemy action, this also generates a message to the decision component.

Another theme of information flow is the presence of an authorization loop for critical decisions. Once a decision has been made, it is passed to higher authority as a recommendation. If authorization is received, then a command is issued. Authority structure has not been defined officially for the SDI command center. However, we assume that at least some of the decisions made in the command center would require authorization by higher authority before action could be taken.

MODEL OF SITUATION ASSESSMENT

Events in the real world are linked to decisions in the command center through the flow of information. The architecture of the command and control system and the organization of the command center will determine the path of this information flow. Although the system architecture and the organization for the SDI command center have not yet been precisely defined, we were able to trace the flow of information among command system functions by making a set of reasonable assumptions about system architecture and organization. This paper analyzes the effects of varying some of these assumptions.

The purpose of the situation assessment model is to support sensitivity analysis of the time required to make decisions under different assumptions about the architecture and organization of the command center. The point of the model is not to provide absolute answers, but to allow meaningful comparisons. It provides a framework for asking questions about the effects of changes in such command center design variables as centralization of functions, lines of communication, distribution of expertise, and structure of authority.

The time required to make a decision following an event was selected as the central variable for the sensitivity analysis because of the critical time requirements for SDI decisions. Minutes and seconds, rather than hours, will be the meaningful units for analyzing SDI command center reaction times. Obviously, the accuracy of decisions will also be extremely important, and a tradeoff can be expected between decision accuracy and reaction time. This issue is not dealt with specifically in our analysis because we lacked the detailed information needed for a meaningful analysis of decision accuracy, but it is an essential area for further work as the SDI command and control system is more precisely defined.

Each of the boxes in Figure 1 has been expanded into a Petri net formulation. However, the resulting model is too detailed and complex for full discussion here. Examples of how command center activities may be modelled using Petri nets are discussed below.

Figure 2 shows an example of intelligence processing. As soon as intelligence data are available, analysis can begin This analysis produces indicators and warnings, as well as a data base that can be used to make estimates of the time available before the earliest possible enemy action. Indicators and warnings are transmitted to the situation asssessment and decision components. Estimates of the time available are transmitted to situation assessment. The time delays associated with this part of the model include the time needed to analyze the raw data, the time needed to estimate the time available, and the time needed for transmission.

Figure 3 shows an information processing activity that is typical of the surveillance maintenance and monitoring functions. When data on surveillance system operations are received, they are processed. If any evidence of ASAT actions is detected, a message is immediately sent to the decision component. If no ASAT actions are detected, the data are aggregated and passed on to the next level of analysis.

Note that a probability or priority must be assigned to the two arcs that emerge from the "detection of ASAT action" place. What is the probability of detecting an ASAT action? This probability should depend on whether

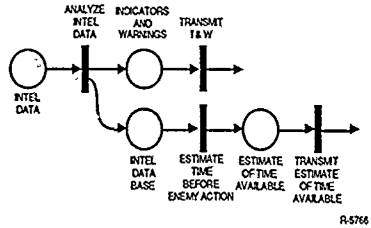


Figure 2. Example of Intelligence Processing Activities

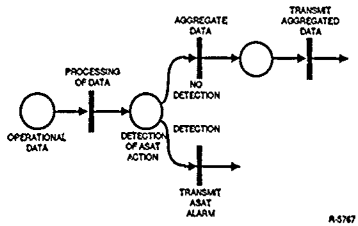


Figure 3. Example of Information Processing During Surveillance Maintenance and Monitoring

or not such an action has actually occurred. The full model allows for the assignment of different values to this probability, depending on whether or not there has been an enemy action. If an action has actually occurred, then the probability of detecting the action is the probability of a true alarm. If no action has occurred, then it is the probability of a false alarm. The time delays associated with this part of the model include the time needed to process the data, the time needed to aggregate the data, and the transmission time.

Figure 4 gives an example of how the assessment of sensor capability in the context of the world situation is represented by the Petri net model. Two kinds of information are needed for the assessment: aggregated operational data, and data on world tension levels from indicators and warnings. If either type of information is missing, then the assessment cannot take place. Sensor capability problems may be assessed as either "serious" or "not serious." Serious problems are passed on for more processing; nonserious problems are not. A probability or priority governs which of these two paths is taken by a token leaving the "seriousness of problem" place. Time delays for this portion of the model include the time before tokens become available at both of the initial places, the time needed to evaluate the capability problems, and the time needed to transmit the serious problems.

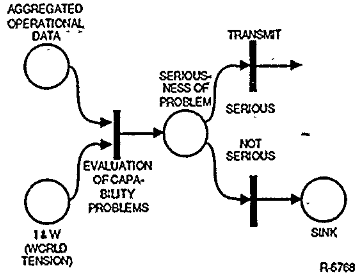


Figure 4. Example of Sensor Capability Assessment.

Figure 5 shows how a decision can be represented by the model. The decision in the diagram maps three initial conditions--high world tension, serious sensor capability problems, and insufficient time to fix those problems--into four different actions: recommend DEFCON change, recommend cool down of sensors, recommend change in sensor thresholds, and recommend enabling of weapons.

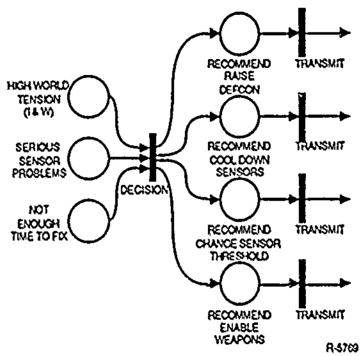


Figure 5. Example of Decision Process

Note that when decisions are reserved to a higher authority, the output of the decision process at the command center is a decision to forward one set of recommendations versus alternative sets. The Petri net, as it is pictured, represents a decision rule stating that, if all three initial conditions occur, then all four of the actions should be taken. However, if any one of the initial conditions is missing, e.g., there are serious sensor problems in a situation of high world tension but they can be fixed within the time available, then none of the four subsequent actions should be taken. The time delay here includes the time that elapses before there are available tokens in all three of the initial condition places, the time needed to make the decision, and the time needed to transmit the four different recommendations for action.

Figure 6 shows a representation of the authorization and command process. When a recommendation for action is received, some time elapses before it is authorized, and the authorization is transmitted back to the originator of the recommendation. A command cannot be issued until the authorization is received. The higher authority authorizing the command is shown as being separate from the command center that issues the command. This is a plausible assumption, but not a necessary one. If no separate authorization of commands is required, then the delays associated with "authorize" and "transmit" in the model would be set at zero.

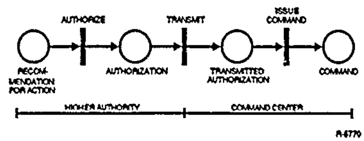


Figure 6. Example of Authorization and Command Process

Figures 2 through 6 have illustrated the ways in which command center activities and information flows are represented in the complete Petri net model. The full model adds some additional detail and links all of the activities together so that delays can be measured from the time that data are received by the system (indicated by an initial available token in the place representing that type of data) to the time that a command is issued (indicated by the creation of a token in the place representing that command).

SENSITIVITY ANALYSIS

The purpose of the model is to analyze the elapsed time between events and decisions under different assumptions about command center architecture and organization. In order to perform this sensitivity analysis, we must assign numerical time values to each of the transitions in the model that represents a process or activity that will consume time. Ideally, these time parameters should be based on empirical data about the time actually taken to complete those tasks. Because the design of the SDI command center is still in a preliminary stage, however, such data are not available. Instead, we have made a set of arbitrary assumptions about possible times that seemed reasonable, and then varied them to test the sensitivity of the overall response time to the processing times for individual components.

The purpose of the model is to allow the comparison of reaction times under different command center organiza tion and system architectures. Reaction time, in this context, refers to the time that elapses between the time that raw data are received by the system, and the time that a command is issued based on those data. For example, in the case of an ASAT action, how much time elapses between the receipt of data from which this attack could be detected and the issuing of a command to cool down the sensors based on the detection of the attack? Of course, some time will elapse between the time that an event occurs and the time that the raw data reflecting that event are received by the system, but this is outside the scope of our model.

The first step in the sensitivity analysis is to establish a set of reaction times that are of interest. We want to examine the links between commands based on critical decisions and the events that generate them. Our approach was to develop a set of "scenarios" of possible events and identify the critical decisions that might be triggered by those scenarios.

The four scenarios created for the analysis and the critical decisions triggered by each are shown in Table 1. The first scenario is simply the detection of a real-time indicator and warning. This is assumed to trigger four critical decisions: cooling down the sensors, changing sensor sensitivity thresholds, recommending a DEFCON change, and enabling the weapon system.

The second scenario involves sensor capability problems. indicators and warnings show a high state of world tension, the DEFCON level is high, and a serious sensor capability problem has emerged. The decisions triggered by this scenario are cooling down of the sensors, changing thresholds, and recommending a shift in ROE option. Scenario 3 adds a time "window" constraint to scenario 2. there is not enough time to fix the sensor problem before the estimated time of the earliest enemy attack. This is assumed to trigger the additional decisions to recommend a DEFCON change, and to enable the weapon system.

The fourth scenario deals with the detection of ASAT actions. Three possible detection points are included: early detection, as raw data for individual sensors are processed during surveillance maintenance, later, as more aggregated data are processed during surveillance monitoring, and still later, during situation assessment, when global operational and enviromental data are analyzed along with intelligence data. The decisions triggered by the detection of an ASAT attack are asssumed to be the same as those triggered by real-time indicators and warnings: cooling down the sensors, changing the sensor thresholds, recommending a DEFCON change, and enabling the weapon system.

The model may now be used to estimate the total aggregate time that would elapse between initial data receipt and the issuing of different types of commands for

TABLE 1. LINKS BETWEEN SCENARIOS AND CRITICAL DECISIONS

SCENARIOS	CRITICAL DECISIONS: Cool down sensors	Change sensor threshold	Recommend DEFCON change	Recommend ROE shift	Enable weapons
1. A real-time I&W occurs.	X	X	X		X
2. I&W show high world tension, DEFCON is high, and a serious sensor capability problem exists.	X	X		X	
3. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists, and there is not enough time to fix it before the estimated time of earliest enemy action.			X		X
4. An ASAT action occurs, and is detected:					
• early, during maintenance;	X	X	X		X
• later, during monitoring;	X	X	X		X
• not until assessment, when operational, environmental, and intelligence data are combined.	X	X	X		X

the scenarios shown in Table 1. (Note: As computed by the model, the total aggregate time between a given triggering event and the issuing of appropriate commands depends directly on the assumed time delay for each individual activity leading to those commands. Thus, the arbitrariness of the individual time delay parameters is reflected in all response time predictions.)

The Petri net model was implemented on a Macintosh computer using ALPHATECH's Micro-Modeler software. This software supports the construction of Stochastic, Timed, Attributed Petri Net (STAPN) models, the assignment of time parameters to transitions in the model and the assignment of priorities or probabilities to transitions. After the model is construcuted, it may be executed, and performance measures such as response times and delays may be computed.

The model with its present assumptions provides a tool for analyzing how some of the factors in command and control system architecture and the command center organization might affect reaction times. Variables that could be manipulated in organizational and system design include the centralization of functions (geographically or organizationally), the coordination between the command center decision functions and the intelligence processing and surveillance maintenance functions, and the authority structure for approving command decisions. The remainder of this section explores the effects of shifts in these factors on reaction times.

TABLE 2. ELAPSED TIME IN MINUTES BEFORE COMMAND IS ISSUED UNDER TWO DIFFERENT ASSUMPTIONS ABOUT TRANSMITTAL AND COMMUNICATION TIMES

	CRITICAL DECISIONS				
SCENARIOS	Cool down sensors	Change sensor threshold	Recommend DEFCON change	Recommend ROE shift	Enable weapons
1. A real-time I&W occurs	4.0 **7.5**	4.0 **7.5**	7.5 **11.0**		4.0 **7.5**
2. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists.	41.5 **48.5**	39.0 **51.0**		43.0 **50.0**	
3. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists, and there is not enough time to fix it before the estimated time of earliest enemy action.			68.5 **79.0**		69.0 **79.5**
4. An ASAT action occurs, and is detected:					
• early, during maintenance;	13.0 **16.5**	13.0 **16.5**	16.5 **20.0**		13.0 **16.5**
• later, during monitoring;	24.25 **29.5**	24.25 **29.5**	27.75 **33.0**		24.25 **29.5**
• not until assessment, when operational, environmental, and intelligence data are combined.	35.5 **42.5**	35.5 **42.5**	39.0 **46.0**		35.5 **42.5**

Note:
Numbers in regular type are based on a transmittal time of 0.25 minutes.
Numbers in bold type are based on a transmittal time of 2 minutes.

The Centralization of Functions

Transmittal and communications times are pervasive throughout the model. This includes the time needed to pass information from the intelligence processing function to the decision function, the time needed to transmit aggregated data from the surveillance functions to the situation assessment function, the time needed to transmit questions back to surveillance and intelligence processing, the time needed to transmit requests for authorization to higher authority, and the time needed for authorization to be transmitted back to the command function.

A number of factors could increase or decrease transmittal and communication times. The design of the communications system is critical, as is the organizational design of the command center. If functions are performed by the same individual, or by individuals within the same part of the organization or located in the same area, then communication time may be much less than if the same functions are performed by a group of dispersed individuals. For purposes of the sensitivity analysis, two transmittal times were tested: 0.25 minutes and 2 minutes. Table 2 shows the effect on reaction times if the time for transmittal and communication is increased from 0.25 minutes to 2 minutes throughout the model.

Obviously, an increase in transmittal time will increase reaction times. The more functions involved in a scenario, the larger this effect will be. The increase in reaction time for scenario 1, based on real-time indicators and warnings, is 3.5 minutes (calculated as the difference between the reaction times under the two transmittal time assumptions). The increase for scenario 2, based on sensor capability asssessment, is 7 minutes. For scenario 3, which involves analysis of the time available before enemy actions, the increase is 10.5 minutes. For scenario 4, the increase in reaction time for ASAT actions detected early is 3.5 minutes. As more processing is required, the increased communication time will have more of an effect. If an ASAT action is not detected until situation assessment, the increase in reaction time due to slower communication is 7 minutes.

Coordination with Intelligence Processing

The reaction times shown in Table 2 assume that the processing of intelligence data and the generation of indicators and warnings takes place in parallel with the processing of data about surveillance system operations and local satellite environments. If intelligence data and operational data are received at the same time, then indicators and warnings will have been prepared and passed to the total situation assessment function by the time that aggregated operational data are received by that function. In other words, the time spent generating indicators and warnings of world tension under intelligence processing is not part of the critical path that drives reaction times for scenarios 2 and 3 in Table 2.

This assumption of parallel processing may not be realistic. For example, a structure in which intelligence processing is driven by questions generated during situation assessment might be more similar to current practices. Under this assumption, the generation of indicators and warnings from intelligence data would not start until sensor capability problems had been identified and questions had arisen about the seriousness of those problems in the context of world tension. If the model is altered to reflect this structure, then the time needed to generate indicators and warnings will become part of the critical path that determines reaction times and will substantially increase them.

Table 3 shows reaction times (in bold type) for scenarios 2 and 3 (which depend on the generation of indicators and warnings) under the revised assumption about intelligence processing. If the generation of indicators and warnings is reactive, rather than proactive, then the reaction time under both scenarios 2 and 3 increases by 15.5 minutes. That is, waiting until sensor problems arise to begin analysing intelligence data to evaluate their significance leads to a substantial increase in reaction time. Continual, on-going analysis and evaluation of intelligence data, i.e., proactive fusion, supports much faster reactions.

Coordination Between Surveillance and Command Functions

The reaction times shown in Tables 2 and 3 assume that questions about the time needed to make sensor adjustments or fix sensor problems are generated during total situation asessment, and referred back to sensor maintenance experts in the surveillance maintenance function. These experts then transmit answers to the questions back to situation assessment. This process is part of the critical path for decisions in scenario 3.

The assumption that a round of questions and answers must occur between the surveillance maintenance function and the situation assessment function may be incorrect for several reasons. First, the individuals responsible for situation assessment may have enough knowledge of the sensor system to know how long repairs and adjustments will take, without having to consult sensor maintenance staff. Also, the sensor maintenance staff may anticipate the needs of situation assessment, and may pass along information about the time needed to make adjustments at the same time that they transmit information about problems in sensor capability.

In either case, the result will be faster reaction times for those actions triggered by the assessment that not enough time is available, as shown in Table 4. Reaction time is decreased by 5.25 minutes under the assumption that estimates of the time needed for sensor repairs can be made by the individuals performing situation assessment, without referring back to surveillance maintenance.

TABLE 3. ELAPSED TIME UNTIL COMMAND IS ISSUED UNDER TWO DIFFERENT ASSUMPTIONS ABOUT COORDINATION WITH INTELLIGENCE PROCESSING

	CRITICAL DECISIONS				
SCENARIOS	Cool down sensors	Change sensor threshold	Recommend DEFCON change	Recommend ROE shift	Enable weapons
2. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists.	41.5 **57.0**	39.0 **54.5**		43.0 **58.5**	
3. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists, and there is not enough time to fix it before the estimated time of earliest enemy action.			68.5 **84.0**		69.0 **84.5**

Note:
Numbers in regular type are based on the assumption that intelligence processing takes place in parallel with processing of operational data.
Numbers in bold type are based on the assumption that intelligence processing does not begin until questions are generated from the processing of operational data

TABLE 4. ELAPSED TIME BEFORE COMMAND IS ISSUED UNDER TWO DIFFERENT ASSUMPTIONS ABOUT COORDINATION BETWEEN SURVEILLANCE AND COMMAND FUNCTIONS

	CRITICAL DECISIONS				
SCENARIOS	Cool down sensors	Change sensor threshold	Recommend DEFCON change	Recommend ROE shift	Enable weapons
3. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists, and there is not enough time to fix it before the estimated time of earliest enemy action.			68.5 **63.25**		69.0 **63.75**

Note:
Numbers in regular type are based on the assumption that the total situation assessment function must refer questions about the time needed to make sensor adjustment back to the surveillance maintenance function.
Numbers in bold type are based on the assumption that the total situation assessment function does not refer to questions about the time needed to make sensor adjustments back to the surveillance maintenance function

Authority Structure

The structure of the model (Figure 6) provides an authorization loop for all decisions made in the SDI command center. For each command, a message must be transmitted to higher authority, an authorization decision must be made, and the authorization must be transmitted back to the command center before the command can be issued.

Obviously this process consumes time, and it is in the critical path for all decisions. However, these assumptions are dependent on the not-yet-determined authority structure of the SDI command center, and reasonable alternatives may be generated. For example, perhaps decisions that involve the surveillance system, such as sensor cool down and threshold changes, need not be approved by higher authority.

Table 5 shows the decrease in reaction time from removing the authorization loop for sensor decisions. Authorization time is on the critical path for sensor decisions, so the removal of the time needed for transmission and authorization has a direct effect on reaction times. Removing the need for authorization cuts 1.5 minutes for decisions based on real-time indicators and warnings and ASAT alarms, and 5.5 minutes for decisions based on sensor capabilities.

CONCLUSIONS

The sensitivity analysis based on the model identifies a series of issues that must be addressed in evaluating designs for an SDI command center. The model provides a structure for analyzing the links between functions and the implications of those links for the overall reaction time of the system. Any design for an SDI command center must take into account the relationship between information flow and reaction speed.

The centralization of functions and the speed of communications within the command center will have a pervasive effect on the speed with which decisions can be made and commands can be issued. Situation assessment in the comand center will require the fusion of information from multiple sources, and the speed of the slowest link will determine the speed of the overall assessment.

TABLE 5. ELAPSED TIME BEFORE SENSOR COMMANDS ARE ISSUED, WITH AND WITHOUT THE NEED FOR APPROVAL BY HIGHER AUTHORITY

CRITICAL DECISIONS

Cool SCENARIOS sensors	Change down threshold	sensor	
1. A real-time I&W occurs 2.5 2.5	4.0	4.0	
2. I&W show high world tension, DEFCON is high, a serious sensor capability problem exists.	41.5 **36.0**	39.0 **33.5**	
4. An ASAT action occurs, and is detected:			
• early, during maintenance;	13.0 **11.5**	13.0 **11.5**	
• later, during monitoring;	24.25 **22.75**	24.25 **22.75**	
• not until assessment, when operational, environmental, and intelligence data are combined.	35.5 **34.0**	35.5 **34.0**	

Note:
Numbers in regular type are based on the assumption that sensor commands must be approved by higher authority.
Numbers in bold type are based on the assumption that sensor commands do not require approval by higher authority

The extent to which questions and answers pass back and forth between functions is another major factor in reaction time. Interchanges take time, and if situation assessment can be done without the need for referring questions back to intelligence processing or surveillance maintenance, this will speed reaction time considerably. The elimination of such interchanges will require either that the staff involved in situation assessment be extremely knowledgable about the intelligence situation and the functioning of the surveillance system, or that staff in those areas anticipate the needs of the situation assessment staff and pass on exactly the information that is needed.

The integration and coordination of intelligence analysis with situation assessment is a key factor in reaction times. If intelligence analysis is done only on request, after serious sensor capability problems have been identified, this will slow reaction times by a substantial amount. Continual, on-going intelligence analysis that anticipates the needs of situation assessment in the SDI command center will lead to faster reaction times.

Authority structure is another important element of reaction time. Authorization loops are part of the critical path for all of the decisions in the model. A reduction in the time needed to transmit intended commands to higher authority and to receive authorization to proceed will reduce command center reaction times by an equal amount.

This sensitivity analysis is intended only as an illustration of the kinds of questions that must be evaluated in assessing command center designs. The basic idea of the STAPN model is to establish linkages among functions and to identify critical paths of information flow leading up to decisions. As plans for an SDI command and control system are further developed, it should become possible to assign more realistic values to time delays in the system and reach more exact conclusions about reaction times under different designs. The STAPN model is also capable of generating probability distributions of reaction times and computing means and variances.

REFERENCES

BBN Laboratories Inc., *An Air Force Advanced Engineering Program for SDI BM/C3 Human Factors: Final Report*, October, 1986.

Peterson, J.L., *Petri Net Theory and the Modelling of Systems*, Prentice-Hall: Englewood Cliffs, New Jersey, 1981.

Pulliam, R. and H.E. Price, *Automation and the Allocation of Functions Between Human and Automatic Control: General Method*, Air Force Aerospace Medical Research Laboratory, February, 1985.

Pulliam, R., H.E. Price, J. Bongara, C.R. Sawyer and R.A. Kisner, *A Methodology for Allocating Nuclear Power Plant Control Functions to Human or Automatic Control*, United Carbide Corporation, August, 1983.

SDI BM/C3 Working Group for Standards, *Functional Decomposition*, July, 1987.

Wohl, J.G. and R. Tenney, *Integrated Analysis Techniques for Command, Control, and Communications Systems, Volume I: Methodology*, Technical Report TR-293, Alphatech, Inc., November, 1986.

SDI Architecture Reports Reviewed

BM/C3 for SDS: Requirements, Concept Definition, and Technology Issues (U), Report No.TB-C-088714, Teledyne Brown Engineering Co., Huntsville, AL.

System Architecture and Key Trade-off Studies: BM/C3 Workshop (U), Report No. TB-C-088711, Teledyne Brown Engineering Co., Huntsville, AL.

SDI BM/C3 Architecture Study: Final Technical Reports (U), IBM Corporation:
- Vol I: Executive Summary
- Vol II: Entry-level BM/C3 Architecture
- Vol III: BOOXT Terminal BM/B3 Architecture
- Appendix C: Communications Sub-architecture

SDI/BM/C3 Architecture Study Phase II (U), Ford Aerospace Corporation:
- Vol I, Executive Summary
- Vol II: Appendix E, Entry-level Architecture
- Vol III: Responsive Architecture and Appendices B and G

Zero One Study Report Draft (U), SRS Corporation, 20 March, 1987:
- Vol I: Report
- Appendix D: BM/C3

NORAD Reports Reviewed

Air Force Aerospace Armstrong Medical Research Laboratory, *Human Factors Engineering in the NORAD Command Post*, April, 1981.

Guarriello, R.V., et al., *Integrated Analysis Technique Development and Applications of IDEF0 to NORAD Cheyenne Mountain Complex*, System Development Corporation, November, 1981.

Nelson, M.A., *Design Considerations for NORAD C3 Displays*, System Research Laboratores, Inc., Technical Report No. SRL-HF 83-100, September, 1983.

System Development Corporation, *NCMC Functional Analysis with Conclusions and Recommendations: Executive Summary*, January, 1983.

BAYESIAN NETWORKS FOR COMMAND AND CONTROL

Stuart L. Crawford, R.A. Riemenschneider, John Sweet, & A. Joseph Rockmore

Advanced Decision Systems
1500 Plymouth Street
Mountain View, CA 94043-1230

ABSTRACT

This paper describes an approach to the development of an *attack assessment* decision aid for command and control, based upon the formalism of the *Bayesian network*. The advantages of Bayesian networks as a representation scheme particularly appropriate to the modelling of inherently uncertain domains is discussed, as are two mechanisms for updating probabilistic beliefs as new evidence is acquired. In addition, a powerful approach to *knowledge acquisition* for such networks is described, and a novel means of computing and updating *high-level hypotheses* of interest is introduced. The paper ends with a brief description of a sophisticated Bayesian network development environment in which such features have been successfully implemented.

INTRODUCTION

For command and control (C^2), facilitating the battle commander's ability to make high quality and timely decisions is of paramount importance. In order to assist him in his decision making processes, decision aids will play an increasingly important role, due to the increases forseen in complexity, data rates, and time allotted to decision-making.

A command and control decision aid (DA) is one that assists a human decision-maker in making decisions related to the exercise of authority over system operations and to the exercise of direction over system activities. A knowledge-based decision aid has, as its basic functionality, a process that incorporates knowledge of the domain, and employs that knowledge in one or more of the techniques that has become collectively known as artificial intelligence. The value of knowledge-based DA's is that they:

- handle uncertainty, ambiguity, and inaccuracy,
- provide heuristics and inference, as well as algorithms,
- capture the expertise and experience of commanders working in the domain,
- allow capabilities for the DA to grow gracefully over time as new knowledge is acquired, and
- reduce the complexity of problems that the human must handle.

ATTACK ASSESSMENT

A class of C^2 problem particularly well suited to decision aiding is that of *situation assessment*. A formal definition of situation assessment is: Given a pre-existing representation of what it means to be a "situation", a current model of friendly assessment capabilities (sensory systems), along with measurements from multiple sensors of several types, form an explanatory model of the world with the following features:

- The most important aspects of the situation receive the most attention.
- Uncertainty and conflicts are both minimized and clearly represented.
- The model of the world has strong predicitive power.

A situation assessment problem of particular interest is that of *attack assessment*, characterized by both *uncertain* relationships among entities in the domain, and *uncertain* input evidence[1]. A *solution* to the attack assessment problem must provide global assessments of the current situation[2], as well as separate estimates of attack *source*, *type*, and *objective*. Furthermore, a tool for solving the attack assessment problem should allow the user to *incorporate* non-sensed[3] information, to *examine* attack assessment hypotheses, to *understand* and *influence* the decision-making process, and to *activate* other decision-making processes.

There are several fundamental issues that must be addressed when designing a solution:

1. How to *represent* uncertain information, both internally and to the user.
2. How to perform *inference* with uncertain information.
3. How to define and evaluate *hypotheses* about the current situation.
4. How to acquire domain knowledge.
5. How to facilitate human-computer interaction.

The solution described in this paper addresses these issues as follows:

1. Use Bayesian networks as a knowledge *representation* scheme
2. Use the Shachter and Pearl algorithms to perform *inference*
3. Use the IHM algorithm to define and evaluate *hypotheses*
4. Use "backwards thinking" for knowledge acquisition
5. Use a sophisticated user interface development system ("Shark") to facilitate human-computer interaction

The approaches enumerated above are discussed in some detail in the sections to follow.

BAYESIAN NETWORKS

Bayesian networks were selected as a paradigm for knowledge representation and inference for both theoretical and practical reasons. The theoretical shortcomings of the uncertainty mechanisms used in first generation expert systems, typified by MYCIN's rule-based scheme in which "certainty factors" attached to production rules of the form $A \rightarrow B$ were used to adjust the certainty of B when the certainty of A changed, have become well understood [2, 5]: given any modular, incremental updating procedure, *qualitatively* anomalous adjustments are likely unless the structure and content of the knowledge base is severly restricted. Strictly Bayesian methods were believed to be too difficult to engineer and computationally intractable. Bayesian networks address both these apparent difficulties by graphically encoding conditional independencies. As a result, the expert need only supply a small number of conditional probabilities, more comparable in number to the certainty factors in a rule based system than the full joint distribution, that are directly meaningful. The benefits of Bayesian networks, for both representation and inference, are described more fully in the sections immediately following.

BAYESIAN NETWORK REPRESENTATION

One of the most appealing properties of the Bayesian network as a knowledge representation methodology is the ease with which such networks allow domain experts and knowledge engineers alike to separate the *qualitative* structure of a given domain from the *quantitative* detail. The qualitative and quantitative aspects of knowledge representation in Bayesian networks are described below.

Qualitative Representation: For many, the quantitative, algebraic constructs of probability theory are unintuitive and difficult to interpret, despite the fact that most of the *concepts* of probability theory are both familiar and intuitive, as will be shown below. Consider, for example, the formal, algebraic definition of *probabilistic independence*:

$$P(A,B) = P(A)P(B) \tag{1}$$

Expression 1 indicates that two random events (A and B) are probabilistically independent if their joint distribution can be computed as the product of the individual distributions. Although this expression is rather uninterpretable to the lay reader, it is *incorrect* to conclude that the concept of probabilistic independence is unintuitive. On the contrary, nearly everyone maintains an internal "model" of what it means for events to be probabilistically independent. For example, whereas one might be reluctant to express the individual probabilities of two events such as "winning the lottery" and "getting a flat tire tomorrow", one would typically express no hesitation in expressing the belief that the two events are *independent*. In addition, it is certainly true that one's belief in this independence is *not* a result of computing the product of the probabilities of the two events and evaluating the equality of the computed product and the joint probability of the two events! It is clear that the intuitive, qualitative concept of probabilistic independence is, in the minds of many, quite distinct from the quantitative, algebraic expression of that concept.

This phenomenon may be simply demonstrated for two other probabilistic concepts: *probabilistic dependence* and *conditional independence*. The algebraic representation of probabilistic dependence is:

$$P(A \mid B) = \frac{P(A,B)}{P(B)} \tag{2}$$

Now, expression 2 is rather unintuitive despite the fact the the concept of probabilistic dependence is quite familiar. For example, although one might be reluctant to express the probabilities of two events such as "winning the lottery" and "buying a lottery ticket", one would have little hesitation asserting that "winning the lottery" *depended* upon first buying a ticket. Finally, consider the algebraic definition of conditional independence:

$$P(A \cap B \mid C) = P(A \mid C)P(B \mid C) \tag{3}$$

Although mystifying to many, expression 3 defines a common and intuitive probabilistic scenario. Imagine a situation in which you are waiting at a bus stop, and wondering when the next bus will arrive[4] Now, information about the departure time of the last bus is certainly relevant to the arrival time of the next bus and, in fact, it is clear that the two events are dependent. However, when information about the *location* of the next bus is acquired, information about the departure time of the last bus is *no longer valuable*. The events "arrival time of the next bus" and "departure time of the last bus" are *conditionally independent* given observation of the event "location of the next bus".

When considering how to best represent uncertain events for eventual implementation as a DA knowledge base, the observation that *qualitative* interpretations of probabilistic concepts are, to many, completely distinct from *quantitative* representations demands a representation scheme that facilitates the *qualitative* representation of these probabilistic concepts. Bayesian networks[3] support this need remarkably well.

Bayesian networks are directed acyclic graphs of nodes and links, in which *state nodes* represent the properties of an entity in the uncertain situation, *evidence nodes* represent accumulated evidence bearing upon those entities, and links represent probabilistic dependencies—often causal, sometimes merely evidential—between the linked nodes. The Bayesian network representation acts as a bridge between the qualitative description of an uncertain situation and its eventual quantitative specification. This fundamental distinction allows for the rapid knowledge engineering of domain expertise.

Figure 1 illustrates the concept of probabilistic independence, as represented by a Bayesian network. Note how the independence of the events "win lottery" and "flat tire" is indicated by the *absence* of a link between the nodes. Figure 2 illustrates the concept of probabilistic dependence, as represented by a Bayesian network. Note how the dependence of the events "win lottery" and "buy ticket" is indicated by the *presence* of a link between the nodes. The link *to* "win lottery" *from* "buy ticket" indicates that winning the lottery is probabilistically dependant upob first buying a ticket. Finally, Figure 3 illustrates the concept of conditional independence, as represented by a Bayesian network. Note how the conditional independence of the events "arrival time of next bus" and "departure time of last bus" is indicated by the *absence* of a link between those nodes. A situation of general probabilistic dependence amongst the three nodes would *include* a link between "arrival time of next bus" and "departure time of last bus", and the qualitative information about conditional independence has allowed the network to be represented more parsimoniously.

Figure 1: Probabilistic Independence

Figure 2: Probabilistic Dependence

Figure 4 illustrates a simple Bayesian network for an attack assessment problem. Note that, in this figure, state nodes are represented as ellipses and evidence nodes are represented as rectangles. Note also that the qualitative structure *alone* reveals much about the nature of the process being modelled.

Quantitative Representation: Once the qualitative representation has been described, then the *value sets* associated with a node and the corresponding conditional probability matrices associated with sets of nodes are formalized. In an attack assessment DA, for example, a node *Extent*, representing the extent of the attack, with value set $\{extent_j : j \in J\}$ might have a node *Motive*, representing the motive for the attack, with value set $\{motive_i : i \in I\}$ among its predecessors. A link from the *motive* node to the *extent* node represents the fact that the motive for the attack is among the causes of the extent of the attack. In this case, this conditional probabilities

$$P(Extent = extent_j \mid Motive = motive_i)$$

would be assessed. The *means* by which such assessments are made is discussed in the section to follow.

Knowledge Acquisition: Knowledge-based systems, by their very nature, demand a rich substratum of *domain knowledge* in order to be effective. In standard rule-based systems, the domain knowledge is generally represented as "IF...THEN..." style *production rules*, and the process of acquiring domain knowledge from subject matter experts and implementing this knowledge in the form of rules is known as *knowledge engineering*. Unfortunately, the knowledge acquisition process is difficult and time consuming, since many domain experts find it difficult to articulate their knowledge in the form of rules. In fact, it is widely acknowledged [1] that knowledge acquisition represents a major knowledge engineering bottleneck.

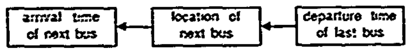


Figure 3: Conditional Independence

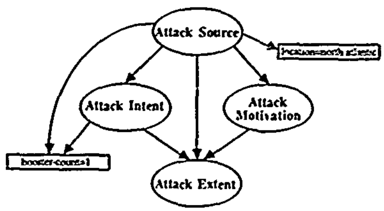


Figure 4: A Simple Network for Attack Assessment

Unlike traditional rule-based systems, the attack assessment DA does *not* use rules for knowledge representation, but instead uses the Bayesian network formalism. The use of Bayesian networks helps to ameliorate some of the problems associated with knowledge acquisition in two ways.

1 the domain expert is permitted to first describe the strictly *qualitative* relationships amongst the entities involved in the scenario being engineered, and

2 when the time comes for the elicitation of quantitative detail, the elicitation process is facilitated by use of a process known as "*backward thinking for knowledge acquisition*" [9].

Figure 5: Diseases Cause Symptoms

Figure 6: The Probability of Interest

The heart of the "thinking backward" approach to acquiring the quantitative detail of a Bayesian network lies in the idea of eliciting probabilities from the domain expert in the *causal direction*. Although, strictly speaking, the links in a Bayesian network indicate only *probabilistic dependence*, it is often useful, particularly for knowledge acquisition, to make use of a causal interpretation where appropriate.

As an example, consider the simple network shown if Figure 5. Probabilistically, one can infer from this figure that the observation of symptoms (evidence node) is dependent upon the presence or absence of disease (state node). In addition however, one can interpret Figure 5 as an indication the disease *causes* the symptom — a perfectly reasonable interpretation. If the simple network shown in Figure 5 made up part of the knowledge base for a simple medical diagnosis decision aid, one would require that the decision aid be able to ascertain how belief in the proposition that a patient *has* the disease changes with observance of *symptoms* of disease and, as such, the probability of interest is *Probability*(*disease* | *symptom*), as shown in Figure 6.

In terms of knowledge acquisition, the domain expert would likely have considerable difficulty expressing *Probability*(*disease* | *symptom*), but would experience much less difficulty expressing *Probability*(*symptom* | *disease*). Since diseases clearly *cause* symptoms, and not the other way around, when one elicits *Probability*(*symptom* | *disease*) from the domain expert, the probabilities are being elicited in the *causal direction*. Furthermore, since the knowledge that is obtained is the "reverse" of the knowledge that is eventually desired[5], the elicitation process is known as "backward thinking". The method by which *Probability*(*disease* | *symptom*) is computed from *Probability*(*symptom* | *disease*) is Bayesian network *inference*, described below.

Summary: Once node value sets and conditional probabilities have been quantified, the Bayesian network becomes a formal representation of the joint probability distribution of the situation being modelled. Any given network is, however, only a particular representation of the joint probability distribution. For any given distribution, there are a number of different Bayesian network representations, each conforming to the axioms of the probability calculus. Although each representation is equivalent in a probabilistic sense, the representations differ in terms of explanatory power and (potential) computational efficiency. The selection of a parsimonious, easily understood representation is thus a crucial process that is, in fact, greatly facilitated by the ability to separate qualitative structure from quantitative detail, as described above.

BAYESIAN NETWORK INFERENCE

Since, clearly, one's belief in the certainty of a given situation changes with the acquisition of evidence, a mechanism must be made available by which the impact of newly acquired evidence can be revealed by the Bayesian network representation of the situation. Two such mechanisms, each of which makes use of the same Bayesian network representation scheme, have been implemented. Although both algorithms operate within the same basic updating loop,

i acquire evidence

ii create evidence nodes

iii attach evidence nodes to the network

iv compute *a posteriori* probabilities for all state nodes

v compute *hypotheses* about situation

vi go to *i*

each algorithm computes *a posteriori* probabilities for the state nodes in a different manner. The two algorithms, referred to subsequently as *Shachter's Algorithm* [8] and *Pearl's Algorithm* [7], are described below.

Shachter's Algorithm: Typically, one wishes to compute the degree of belief in the values of a *single* state node, conditioned upon the acquisition of some set of evidence nodes. Probabilistically, this represents the computation of an *a posteriori* probability for the state node, as derived from the original joint probability distribution represented by the network. In effect, this requires that *all* state nodes except for the one selected must be removed from the network, a procedure accomplished via the manipulation of the graph structure itself, by means of the processes of *node removal* and *link reversal*. Figure 7 shows a simple example of the kind of graph manipulation that the computation of *a posteriori* probabilities would require.

A state node may be removed only when it has no successors in the network, and such a node is called a *barren* node. When a node must be removed, and it *has* successors, then links must be reversed so as to create a network in which the node becomes barren. Link reversal corresponds probabilistically to the application of Bayes' rule and any link may be reversed providing that no cycles are created by its reversal. In terms of manipulations to the graph structure itself, link reversal amounts to ensuring that each node associated with the reversal acquires additional links emanating from the other's predecessors. In short, the removal of a non-barren node is accomplished by simply transforming the node into a barren node via the sequential reversal of all links between the node and its successors. Figure 8 illustrates the process of link reversal, and figure 9 illustrates how the *a posteriori* probabilities can be computed via node removal and link reversal.

When evidence is acquired, the appropriate inference making procedure takes place in two steps. Evidence is first added to the Bayesian network as *evidence nodes*, by means of a rule based process in which evidence is recognized and attached to the appropriate nodes in the network along with the associated conditional probability matrix. Once evidence has been attached, link reversal constitutes the process by which one infers the *a posteriori* distribution of the state nodes conditioned upon the evidence.

Pearl's Inference Technique: Unlike the approach described above, this inference mechanism does not involve manipulation of the graph structure itself. Propagation of evidence occurs not by link reversal, but instead by the transmission of beliefs within the framework of an object-oriented, message passing inferential metaphor. In this approach, two processes are involved at a node: the *fusion* of the beliefs associated with the predecessor nodes (causal or anticipatory support) with the beliefs associated with the successor nodes (diagnostic or retrospective support), and the subsequent *transmission* of newly fused information to both successors and predecessors. In this inference scheme, each node acts as an independent processor and so fusion and propagation can take place asynchronously, thus offering considerable potential for the development of a distributed, parallel implementation of this inference mechanism. One pays for this luxury, however, with the coin of representational restriction, since only trees and singly connected graphs are condusive to this approach to inference, unlike the Shachter mechanism which imposes no such restrictions. Figure 10 illustrates the propagation scheme[6]. Two leaves acquire evidence attachments and, subsequent to fusion, new information is propagated

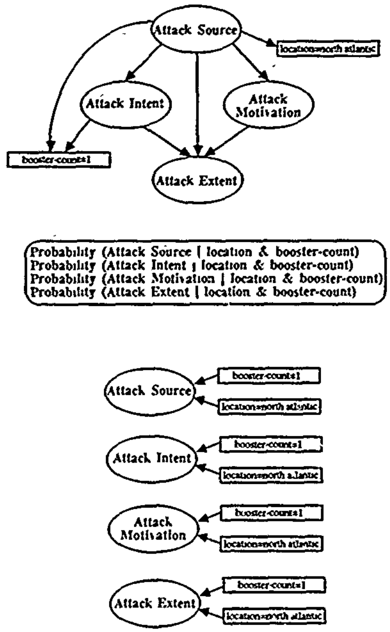


Figure 7: Deriving *a posteriori* Probabilities

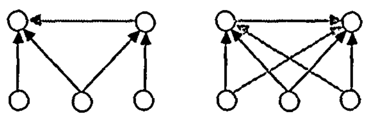


Figure 8: The Link Reversal Process

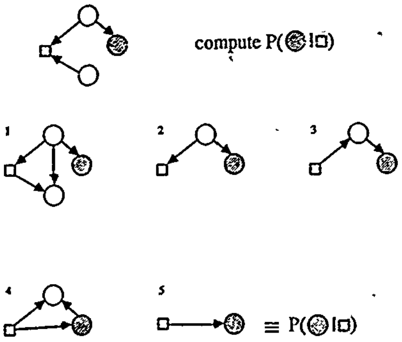


Figure 9: The *a posteriori* Computation Process

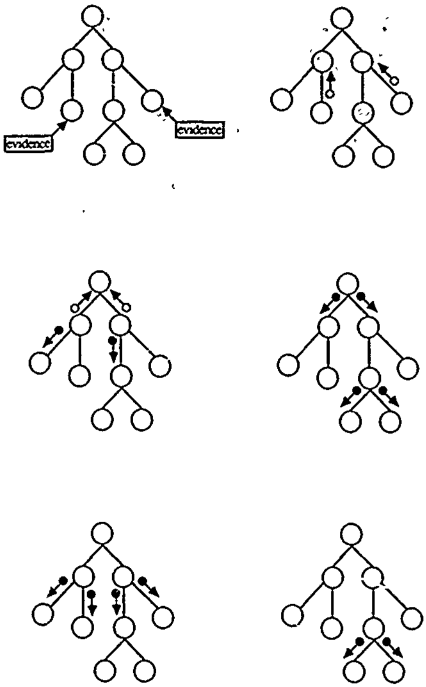


Figure 10: Fusion and Transmission of Evidence

upwards (white arrows) and downwards (black arrows) in *parallel*. When no new information remains to be fused, the graph is in equilibrium and propagation halts.

The availability of two distinct inference procedures, both making use of the same underlying representation scheme, offers a considerable advantage: when the particular topology of the network is appropriate, Pearl's approach to inference[7] can be used whereas, when the network extends beyond the complexity of a singly connected graph, Shachter's more general approach can be utilized.

THE IHM ALGORITHM

One problem in applying the Bayesian network representation to situation assessment has to do with interpretation of the results of inference. For simplicity, assume that our task is to classify the situation assessed as being either *normal* or *abnormal*. Since we have only limited and uncertain indications, this classification must be probabilistic; we must assess the probability p_n that the situation is normal (or, equivalently, the probability $p_a = 1 - p_n$ that the situation is abnormal). Typically, this assessment will depend on the posterior values at many—often *every*—state node in the network. If the posterior distributions on the state nodes generally tend to indicate that the situation is normal, but single posterior value at a single state node is anomalous, we might very well fail to notice this one number among the dozens (or even hundreds) that are computed—even though, given the other values, this single number is a very strong indication that the situation is abnormal. Evidently, some support for a global assessment of the posterior values is required.

The way to address this problem within the framework set forth in the previous section is:

- create a new state node *Situation* with values *normal* and *abnormal*,
- add a link from every state node in the network to *Situation*,
- assess $P(normal \mid x)$ for every combination x of values of X^s, and
- compute a posterior value for $P(normal)$ via Shachter's algorithm.[9]

There are, however, several disadvantages to this approach. Most important among them is that calculating a posterior for *normal* will probably be rather expensive: compared to computing posteriors for one of the maximal nodes in the network, every barren node removal will require reversal of an additional link to *Situation*, and the array of conditionals associated with that link is very large compared to any of the others in the network. Also, Shachter's algorithm does not take advantage of the fact that

$$P(normal\,|\,\mathrm{x}) = 0$$

for the overwhelming majority of values x.

Fortunately, it turns out that computation of p_n is quite simple given rather standard Bayesian methods that nonetheless take advantage of the conditional independencies in the network. Consider the function N from values x to $[0,1]$ defined by:

$$N(\mathrm{x}) = P(normal\,|\,\mathrm{x})$$

The best estimate for the value of N—that is, the best estimate of p_n—given the evidence $E = e$ and the joint probability distribution represented by the network is simply the conditional expectation of N given the evidence.

$$\langle N\,|\,e\rangle \equiv \sum_{\mathrm{x}} N(\mathrm{x})\,P(\mathrm{x}\,|\,e)$$

By Bayes' theorem,

$$\langle N\,|\,e\rangle = \frac{\sum_{\mathrm{x}} N(\mathrm{x})\,P(e\,|\,\mathrm{x})\,P(\mathrm{x})}{P(e)}$$

where

$$P(e) = \sum_{\mathrm{x}} P(e\,|\,\mathrm{x})\,P(\mathrm{x})$$

Note that the quantities $P(e\,|\,\mathrm{x})$ are the the conditional probabilities used in attaching the evidence to the net, that the quantities $P(\mathrm{x})$ can be computed at "network definition"-time, as

$$P(\mathrm{x}) = \prod_i P(x_i\,|\,x_{i+1}, x_{i+2}, \ldots, x_n)$$

and that an addend in the numerator of the formula for $\langle N\,|\,e\rangle$ makes a contribution to the sum only if $N(\mathrm{x}) \neq 0$ making compact and efficient representation of the "definition" of N straightforward.

These formulæ easily generalize to multiple pieces of evidence. Moreover, the complexity of the calculation can be made linear in the size of the network and number of pieces of evidence. Consider the case where there are two pieces of evidence, $E_1 = e_1$ and $E_2 = e_2$. We have

$$\begin{aligned} P(e_1, e_2, \mathrm{x}) &= P(e_1\,|\,e_2, \mathrm{x})\,P(e_2\,|\,\mathrm{x})\,P(\mathrm{x}) \\ &= P(e_1\,|\,\mathrm{x})\,P(e_2\,|\,\mathrm{x})\,P(\mathrm{x}) \end{aligned}$$

where the last equality is due to the fact that the evidence is probabilistically independent given the value of the nodes in the net (*i.e.*, evidence nodes are always final). Therefore,

$$\langle N\,|\,e_1, e_2\rangle = \frac{\sum_{\mathrm{x}} N(\mathrm{x})\,P(e_1\,|\,\mathrm{x})\,P(e_2\,|\,\mathrm{x})\,P(\mathrm{x})}{P(e_1, e_2)}$$

where

$$P(e_1, e_2) = \sum_{\mathrm{x}} P(e_1\,|\,\mathrm{x})\,P(e_2\,|\,\mathrm{x})\,P(\mathrm{x})$$

Generally, for evidence $E_j = e_j$ $(1 \leq j \leq m)$,

$$\langle N\,|\,\mathrm{e}\rangle = \frac{\sum_{\mathrm{x}} N(\mathrm{x}) \cdot \left(\prod_j P(e_j\,|\,\mathrm{x})\right) \cdot P(\mathrm{x})}{P(\mathrm{e})}$$

where

$$P(\mathrm{e}) = \sum_{\mathrm{x}} \left(\prod_j P(e_j\,|\,\mathrm{x})\right) \cdot P(\mathrm{x})$$

Although further optimizations are certainly possible[10], performance of the "unoptimized" version has proven adequate for applications to date. In practice, it has been found that the "brute force" method of computing hypotheses (creating a *hypothesis node* that depends upon other state nodes), described at the beginning of this section is useful for rapid prototyping, and a transition to the more efficient IHM algorithm described above then takes place when a completed DA is ready to be fielded.

A NETWORK DEVELOPMENT ENVIRONMENT

In order to make full use of the Bayesian network approach described in this paper, a network development *environment* must be made available to the DA developer. The authors have developed a sophisticated mouse and window based environment that allows for the rapid prototyping of Bayesian networks. This environment runs on a SUN-3[11] workstation, and is written completely in Common LISP. This Bayesian network development environment allows the DA developer to build networks, define and attach evidence to networks, compute *a posteriori* probabilities, and define and evaluate hypotheses. Graph drawing features of the development environment allow probability distributions to be conveniently examined, and also enable probability distributions to be *entered* graphically. Since networks may be completely defined without the need for programming, the development environment facilitates close interaction between DA developer and domain expert, thus minimizing the knowledge engineering "bottleneck".

CONCLUSIONS

Inherently uncertain domains demand a unique approach to both knowledge representation and inference for situation assessment. The Bayesian network representation has proven to be remarkably successful for ameliorating the problems associated with knowledge acquisition in an uncertain domain and, similarly, the Bayesian network inference schemes have been found to deal with uncertainty in an intuitive and consistent manner. Finally, a rich set of Bayesian network development tools has been found to be essential for effective knowledge engineering.

FOOTNOTES

[1] The term *evidence* is henceforth used to describe feature-value pairs abstracted from raw sensor data.

[2] Such global assessments are referred to as *hypotheses* in the remainder of this paper.

[3] That is, evidence acquired from sources other than sensors.

[4] This example is drawn from one given in [7].

[5] Remember, *Probability*(*disease* | *symptom*) is the probability of interest.

[6] This figure is adapted from one first appearing in [7].

[7] Computationally more efficient than Shachter's approach.

[8] To keep probabilistic formulas a bit more compact, let

$$\mathbf{x} \equiv x_1, x_2, \ldots, x_n$$

and

$$\mathbf{X} \equiv X_1, X_2, \ldots, X_n$$

Then the expression

$$P(normal \mid \mathbf{x})$$

is standard shorthand for

$$P(Situation = normal \mid X_1 = x_1, X_2 = x_2, \ldots, X_n = x_n)$$

[9] Pearl's algorithm cannot be applied to the modified network since there will be multiple paths from any non-final node in the original network to the node *Situation*.

[10] For instance, if

$$P(e \mid \mathbf{x}) = P(e \mid x_1)$$

then

$$\begin{aligned} P(e) &= \sum_{\mathbf{x}} P(e \mid \mathbf{x})\, P(\mathbf{x}) \\ &= \sum_{\mathbf{x}} P(e \mid x_1) \cdot \left(\sum_{x_2, x_3, \ldots, x_n} P(\mathbf{x}) \right) \end{aligned}$$

[11] SUN-3 is a trademark of Sun Microsystems.

REFERENCES

[1] E. A. Feigenbaum, Expert Systems in the 1980's, in *State of the Art Report on Machine Intelligence*, (A. Bond. Ed.), Maidenhead: Pergamon-Infotech, 1981.

[2] D. E. Heckerman and E. J. Horvitz, "The Myth of Modularity in Rule-Based Systems", *Proceedings of the Second Workshop on Uncertainty in Artificial Intelligence*, 1986.

[3] R. A. Howard and J. E. Matheson, "Influence Diagrams", in [4].

[4] R. A. Howard and J. E. Matheson (eds.), *The Principles and Applications of Decision Analysis*, Strategic Decisions Group, Menlo Park, CA, 1984.

[5] R. W. Johnson, "Independence and Bayesian Updating Methods", in [6].

[6] L. N. Kanal and J. F. Lemmer (eds.), *Uncertainty in Artificial Intelligence*, North-Holland, Amsterdam, 1986.

[7] J. Pearl, "Fusion, Propagation, and Structuring in Belief Networks", *Artificial Intelligence* 29 (1986), pp. 241-288.

[8] R. D. Shachter, "Intelligent Probabilistic Inference", in [6].

[9] R. D. Shachter and D. E. Heckerman, "Thinking Backward for Knowledge Acquisition", *AI Magazine*, Fall, 1987.

PERSONALIZED DECISION ANALYSIS (PDA) MODELS AS VEHICLES FOR TACTICAL MI KNOWLEDGE

Dr. Rex V. Brown

Decision Science Consortium, Inc.

MILITARY TASK TO BE ADDRESSED

The motivating problem this paper addresses is how to extract and use knowledge within the military intelligence domain which is relevant to tactical command and control (say at the Army Division level), and is available in advance of combat. It may be resident in a variety of partially overlapping sources, including military professionals, substantive and methodological experts, and documented studies.

The knowledge is to be put at the disposal of tactical commanders and their staffs, who are to use it on the battlefield--either in the form of an on-line computerized decision aid, or in the form of training acquired prior to the battle.

NATURE OF KNOWLEDGE TO BE CAPTURED

The type of tactical MI knowledge we are concerned with is distinctive, in that it involves situations, threats, intelligence and options which it is not feasible to sharply and comprehensively envisage in advance. (By contrast, the success stories of AI expert systems have tended to be for relatively well-structured problems, such as medical diagnosis, oil exploration and engineering design.) Other distinctive features which may also influence the appropriate mode of elicitation, or the format of representation, include: instability (i.e., the substance of military practice evolves), mode of learning (from books and exercises, rather than experience of the "real thing"); and institutional (we are not interested in any individual's knowledge per se).

The knowledge may be at different levels of generality: from that of a field manual, referring, say, to any US-Soviet conflict in Central Europe; to the intelligence preparation of a specific European battlefield, to judgments developed in the course of a particular engagement.

THE PDA APPROACH

The approach adopted is to impose the logical discipline of personalized decision analysis (PDA) on the form of knowledge elicited, so that it is encoded as a coherent system of normative decision and/or inference models incorporating quantitative measures of probability and utility. The PDA prediction approach to KE is practically and conceptually well advanced, and indeed mature software is available for each variant.

The two most familiar PDA modeling paradigms are choice evaluation through subjective expert utility, and inference through Bayesian updating (Barclay, et al., 1977). The knowledge is representing in one of two ways. *Structure* (for example, what enemy intent gives rise to what intelligence indicators) is reflected in the form and labeling of the model. *Content* (such as the extent to which an indicator supports a hypothesis of intent) is reflected in probability and other numbers.

PROGRESS MADE

With the sponsorship of the Army Research Institute, Fort Huachuca field office, DSC has been developing knowledge elicitation (KE) techniques within the PDA paradigm (as well as others from different traditions), and trying them out experimentally in Army contexts, using Army subjects. The focus has been on two common knowledge elicitation tasks, discussed below.

PREDICTING ENEMY ACTION

The knowledge a G-2 uses in assessing enemy course of action or intent is of two kinds, causal and diagnostic, both of which can be represented as probabilistic linkages. Personalist belief nets (typically represented as influence diagrams) capture causal judgments, of the "it all depends" type (e.g. location of enemy attack depends on his mission). Within this framework, predictions of enemy intent (e.g. mission) will often be anchored to a second-guessing of the enemy's decision process, for example in the form of a multiattribute utility model. Updating probabilistic assessments in the light of diagnostic intelligence developments can be captured in Bayesian updating analysis.

Figures 1, 2, and 3 illustrate each of the above techniques with a worked example, purporting to address the same KE task, viz: during IPB (Intelligence Preparation of the Battlefield), for a given division and in a delineated sector of the Fulda Gap region in western Germany after the first few days of a Soviet offensive, how probable is it that the enemy will attack in the North?

FIGURE 1 CONDITIONED ASSESSMENT MODEL

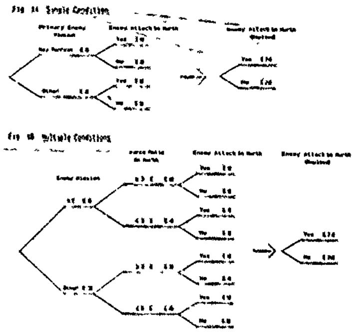


FIGURE 2 BAYESIAN UPDATING

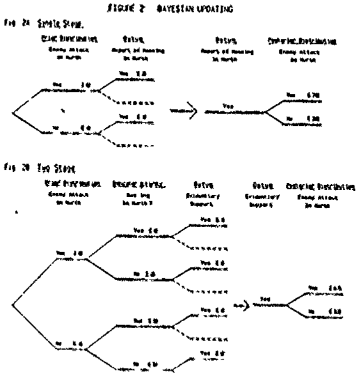
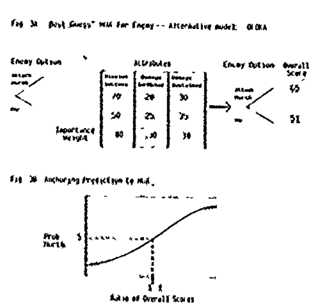


FIGURE 3 MAU TO PREDICT ENEMY CHOICE



The conditioned assessment model shown in Figure 1 has a simple but powerful logic for exploiting the case where the expert's (or other source's) knowledge bears most naturally on one or more causes of the event of interest. When the relevant conditional and unconditional assessments are assigned (using well-established probability elicitation techniques), the required probabilities are straightforwardly calculated. Multiple models can be constructed for the same required assessment, using different conditioning variables, singly or (as in Figure 1b) in combination. Influence diagrams can be used to help organize the linkages, when they become at all complex.

The judgment and data embodied in such PDA models can be logically decomposed into separable sets of inputs, supplied by the same or different sources. In particular, model structure and default content numbers can be developed ahead of battle, using the necessary amounts of time and elicitation expertise which are not available in the heat of battle. For example, in the conditioned assessment example, a menu of typically appropriate model structures would be developed ahead of time (e.g., mission and/or force ratios as determinants of location of attack); as well as any judgment or data on the corresponding conditional probabilities (e.g., on how likely the enemy is to attack where he has a force-ratio advantage of less than 3:1).

Much of the practical value of storing knowledge in this form will depend on being able to complete the model during combat, when the assessment is needed. This requires a user-friendly input-output interface and adequately trained staff. This is a powerful reason to keep the structure simple.

Figure 2a shows the simplest form of a Bayesian updating model, based on the well-known prior-posterior paradigm, where the required assessment is periodically updated as new data is received The BAUDI program developed for ARI is designed to be used in real-time to handle reports of any degree of complexity, provided only that there is someone to assess its diagnosticity, in appropriate conditional probability terms. In this form, all the KE is done during battle, and there is little scope for storing knowledge of enduring interest.

However, a two-stage model of the type shown in Figure 2b permits partitioning out knowledge to be stored ahead of time. Specifically, a required assessment of a given kind (e.g., enemy attack in a given location) can be related (as in the relevant field manuals) to a broadly defined indicator event (e.g., massing of forces) which is of general applicability. Diagnostic generalizations can be stored as conditional probabilities (default likelihood functions), which can be overridden when necessary.

On the battlefield, incoming intelligence only needs to be characterized in terms of how much evidentiary support it provides for the indicator event (typically a scale, rather than yes-no as in this simple example). Conditional probabilities (likelihoods) of evidentiary support given indicator activity could also be given default values in advance. The prior probability (e.g., of enemy attack in this location) would be supplied, on the spot, either by direct judgment, or as output from another exercise, such as conditioned assessment.

The third PDA approach, illustrated in Figure 3, is to capture knowledge which relates to projecting how the enemy will make the choice in question. The possible ways of doing this are as varied as there are PDA choice models, but again there are default models which can be developed to draw on. Multiattribute Utility Analysis (MUA) models are promising, with standard attributes, such as those shown here.

A simple and commonly used additive model will usually work well. Using an expert's "best guess" at how the enemy will weight each attribute, and a score for each of the options in question, an overall score for each such option can be calculated. (Default importance weights would be stored in advance.) These comparative scores need to be converted into probabilities that the enemy will actually adopt each option (since we cannot be sure he will act as we project). This could be done by direct judgment on the battlefield, or via a pre-stored function, such as that shown in Figure 3b. A plausible property for such a function is that if the calculated scores are equal, so are the probabilities (but this does not logically have to be the case).

In general, a given assessment problem can be formulated in different ways corresponding to different "experts" or alternative ways of looking at an assessment by the same expert. These can be pooled or reconciled, and progressively added to or refined, as more knowledge becomes available (Brown and Lindley, 1986).

Figure 4 shows the elements of an integrated KE strategy using the three forms of PDA, and indicating some of the main aspects of supplying structure and content in each case.

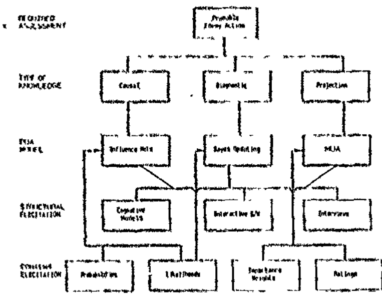


FIGURE 4. ELEMENTS OF PDA KNOWLEDGE ELICITATION PROCESS

SIMULATING THE "UNPREDICTABLE" FLOW OF BATTLE

Step-through simulation is a computer-aided PDA-inspired technique for generating probabilistically realistic sequences through complex ill-defined futures, typified by military combat. It does so by having red, blue, white and combat resolution experts conduct interactive thought experiments, where they specify possible developments in sequence. Probability distributions are assessed, as needed, and sampled, Monte Carlo style.

Unlike conventional Monte Carlo simulation, step-through simulation avoids having to specify in advance, even implicitly, all contingencies and their conditional probabilities. Unlike conventional war-gaming, it limits cognitive myopia and preconceptions, and exercises the full range of possibilities in a controlled fashion.

The resulting sequences are treated as a probability sample of all possible futures, reflecting the knowledge of the participating experts. The sequences can be used to: evaluate the probable consequences of alternative friendly courses of action; generate representative scenarios for various conventional purposes; provide pre-battle training to participants in the simulation; and contribute to a store of surrogate "case studies" from which generalized knowledge can be synthesized.

CONCLUSION

The thrust of PDA models generally is to *enhance* expert knowledge, rather than *emulate* it, (which is basically what typical AI expert systems set out to do).

PDA models assure that conclusions, whether of decision or choice, are logically coherent and can be reviewed and refined, piece by piece, from a multitude of sources (human and other). They can, moreover, be made to conform, at least in essence, to any sound line of reasoning or prescription (and any logical inconsistency can be detected and corrected). For this reason it is a promising candidate for computerizing military field manuals or doctrine, with the prospect of making them richer and more finely articulated, in the form, say, of expert systems for field work stations.

Such expert systems will usually require progressive fine-tuning, in terms of probability, value and importance numbers, right up to the point of use on the battlefield. Until a substantial amount of user-engineering has been done, this will represent a non-trivial burden of time and skill on line and staff in the field. However, this is bound to the case for any computerized "expert system", which purports to replace or enhance military judgment of any but the most routinized kind.

A problem with PDA models for expert systems is that the knowledge of any given human may not fit readily into any particular structure. Cognitive structures differ from one individual to another. (However, this will be a problem with *any* formal representation of knowledge which seeks to accommodate multiple sources). Moreover, important knowledge may be "lost in the translation" as qualitative perceptions of uncertainty, value etc. are forced into the numerical form of probabilities, utilities etc. This is a good argument for making sure that the user of a system can override it with his direct judgment and generally use it as a "collaborator" rather than a replacement.

Key technical issues which still need to be addressed include:

- verbal elicitation protocols for structure and content;
- computer-aided interaction between model and knowledge user (who may also be a source of last-minute knowledge on the battlefield); and
- selecting modeling approaches to match situations.

Although the PDA approach to prediction is well-established, the step-through approach to KE is still at the conceptual stage of development and its implementation. Evaluation of its potential, remains to be established. Interactive software, though primitive, has been developed for this purpose (Ulvila and Brown, 1978). Step-through simulation partakes of the same basic logic of PDA, and can be used for many of the same purposes--such as prediction and evaluating courses of action. However, its implementation is likely to be always too cumbersome and demanding to be either adapted or used on the battlefield. Its most promising use is probably for training at garrison headquarters, and for the development of scenarios.

ACKNOWLEDGMENT

Work on this paper was supported by the Army Research Institute under contract No. MDA903-86-C-0383.

REFERENCES

Barclay, S., Brown, R.V., Kelly, C.W., III, Peterson, C.R., Phillips, L.D., and Selvidge, J. *Handbook for decision analysis.* McLean VA: Decisions and Designs, September 1977.

Brown, R.V. and Lindley, D.V. Plural Analysis: Multiple approaches to quantitative research. *Theory and Decision 20*, 1986,133-154.

Ulvila, J.W., and Brown, R.V. Step-through simulation. *Omega: The International Journal of Management Science*, 1978, *6*(1), 25-31.

SURPRISE ATTACK: LABORATORY EXPERIMENTS AND COGNITIVE THEORY ON COMMAND CENTER EFFECTIVENESS *

PAUL J. HINIKER

DEFENSE COMMUNICATIONS AGENCY

ABSTRACT

This paper reports on a set of results obtained from two experiments carried out at the Naval Postgraduate School in 1987-1988 to investigate some conditions for the effective operation of headquarters organizations in wartime situations. In particular, it analyzes the reactions of commanders, with and without contingency plans, to the receipt of discrepant information occasioned by a surprise attack . Whereas commanders of the contingency planning treatment groups conformed to "rational" expectations by moderating their confidence in the initially reported enemy intent upon learning of the surprise attack, by reporting higher subjective workloads at the onset on the attack, by becoming more informed of enemy actions as a result of their information search, and by performing better against the enemy on the battlefield, the single thread commanders did precisely the opposite.

I. INTRODUCTION

"This dispatch is to be considered a war warning. Negotiations with Japan looking toward stabilization of conditions in the Pacific have ceased and an aggressive move by Japan is expected within the next few days... "Execute appropriate defensive deployment preparatory to carrying out the tasks assigned in WPL46 (the Naval War plan)..." War Warning from Admiral Harold Stark, Chief of Naval Operations to Admiral H. E. Kimmel, Commander in Chief of the Pacific Fleet, Pearl Harbor, Hawaii, November 27, 1941.

"In summary, the members of the Navy group in Hawaii in 1941 bolstered their assumption that Pearl Harbor was immune from attack by accepting well-worn ideological assumptions as the basis for flimsy rationalizations, which they thought were based on hard facts..." In Irving W. Janis <u>Groupthink: Psychological Studies of Policy Decisions and Fiascoes</u>.

At 0800 on December 7, 1941, the Japanese attacked Pearl Harbor and destroyed the Pacific Fleet. This despite the fact that earlier in the year cryptographers in Washington had broken the Japanese military codes and provided ample warning of Japanese intent. Admiral Kimmel and his Naval advisors were so confident of the security of the Fleet a few days before the attack that they blithely presumed the Army Command in Pearl was conducting full-time radar surveillance for them. They even ignored the significance of the burning of papers and codes in the Japanese consulate in Hawaii observed two days before the attack. How are we to account for such phenomena; and how, then, can we prevent them?

Decision Theory Approach

Perhaps decision theory can be an aid to our understanding. A military headquarters can be viewed as a multi-decision makers team supervising a highly complex and uncertain environment. Decision theory, as elaborated in the control theory approach is concerned almost entirely with the control of a set of variables in order to maximize a cost function subject to constraints. As noted by Wohl (1981), however, the preponderance of work in decision theory is primarily prescriptive and has concentrated on techniques for rational option selection, while little work has concentrated on those portions of the process which are of greatest interest to military commanders, <u>viz</u> the creation, evaluation, and refinement of hypotheses (i.e., what the situation is) and options (i.e., what can be done about it.)

Wohl (1981) has developed a Stimulus-Hypothesis-Option-Response (SHOR) paradigm which broadens the perspective on the headquarters decision makers in a laudable attempt to account for real headquarters decision making behavior. Wohl focuses on uncertainty reduction as a principle. In a parallel development, the Headquarters Effectiveness Assessment Tool (HEAT) project, which sponsored these 2 experiments, has also broadened the decision theory perspective in similarly viewing the headquarters decision making process as a cycle of information

*Many thanks are due to Dr. Mark Scher and Mr. Robert Cnoisser of DCA, Dr. Robert Tenny, Dr. Elliot Entin, Dr. Jean MacMillen, Ms. Ann Needalman and Ms. Diane Mikaelian of Alpha Tech and Mr. Paul Lentz of ESI for their advice and support on this particular branching of DCA's HEAT Project. Thanks are also due to Prof. Lester Ingber and Joe Stewart and their students at NPS.

processing. The HEAT cycle begins with monitoring the environment, then understanding the environment, generating options, predicting their consequences, deciding among them, directing activities, and back to monitoring. These 2 closely related paradigms both emphasize the role of the hypothesis testing stage of the decision making cycle and cite contingency planning as a key determinant of effectiveness. Let us deepen this perspective by considering some relevant cognitive mechanisms.

Dissonance Reduction Approach

One of the most important hypotheses that the wartime commander must develop during the course of battle is his estimate of enemy intentions. Beliefs about enemy intent, as all human beliefs, are based upon empirical evidence, logic, and the dynamics of human cognitive processes. As the commander encounters evidence that is consistent or inconsistent with his working belief about the enemy, he is governed by certain rules of cognitive processing. One set of such rules is provided by the theory of cognitive dissonance, and as such, it provides us with provisional explanations of the behavior of commanders in wartime headquarters organizations. How, for example, does a commander behave when entertaining a belief that the enemy is mounting a frontal assault and he is then confronted with increasing evidence that the enemy has mounted a massive air drop to his rear? He experiences cognitive dissonance and pressures for its reduction.

Generally, cognitive dissonance is defined to exist between a pair of beliefs when one implies the obverse of the other (Festinger, 1957). If, for example, a person believes (1) "I have positioned my forces to repel a massive frontal attack by Red" and is then confronted with evidence that (2) "Red is launching a massive attack to our rear", he suffers uncomfortable cognitive dissonance. He can reduce this dissonance by changing his belief in either. If (2) is undeniable, he can only change his prior belief in (1), the intent of Red to mass a frontal attack. He will behave this way if he is not highly committed, by past actions, to belief in Reds launching a massive frontal attack. However, if he is highly committed to belief in the frontal attack, he can only reduce his dissonance by bolstering this cognition with additional consonant evidence, since the magnitude of dissonance is a function of the ratio of dissonant cognitions to the total number of relevant (dissonant and consonant) cognitions. In essence, the persc who is bolstering amasses a great body of other beliefs that are consonant with the conflicting one (McGuire in will, in effect, attempt to "drown out" the dissonant cognition of the rear assult; he becomes open to confirmatory evidence for his prior belief, indeed searches for it, and closed to contradiction.

Thus the reactions of highly committed and relatively uncommitted commanders to a surprise attack are very different. While the highly committed intensify their original belief, increase their confidence and become more active in bolstering their belief in the frontal attack by argumentation and by searching for additional evidence consonant with the validity of the frontal attack, the poorly committed diminish their credence. They feel no need to persuade others of the belief or to obey its behavioral constraints. They become unbelievers and adopt a different, and perhaps more rational, course of action.

II. EXPERIMENTAL METHOD AND RESULTS

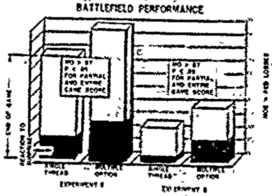
Two experiments were conducted at the Naval Postgraduate School at Monterey using 2 different wargame simulators with military officers as subjects to test the major hypothesis that contingency planning leads to greater battlefield effectiveness and to investigate the cognitive mechanisms that mediate the relationship. In both experiments military officers were organized into a headquarters organization under a commander and fought through a realistic 3 hour battle scenario developed for National Defense University. The scenario depicts a conflict in Iran with Blue forces in defense of the Darzin-Bam Pass through the mountains and roughly equivalent numbers of Red forces driving from Afghanistan to secure the pass. After the first half hour of battle, the Blue forces are hit with a surprise attack off the main axis of Red advance and contrary to prior intelligence reports on Red intentions.

In both experiments, the major independent variable was contingency planning. Contingency planning was operationally defined as use of an oplan that has multiple estimates of enemy intent, multiple responses and a better starting position to meet a variety of threats. Single thread planning was operationally defined as using an oplan that has 1 primary estimate of enemy intent, and 1 primary course of action tailored specifically to meet that threat.

In both experiments, the major dependent variable was an Measure of Effect (MOE) computed from Red attritica (or a combination of Red attrition and Red position). This MOE was computed at end game and also for the first 2 half hourly time periods following the surprise. In addition, several measures of the operation of headquarters information processing were also collected. Figure 1 depicts the results of the tests of the major hypothesis in both experiments: Contingency planning leads to enhanced battlefield effectiveness both at end game and in response to the surprise attack.

FIGURE 1

CONTINGENCY PLANNING
AND
BATTLEFIELD PERFORMANCE



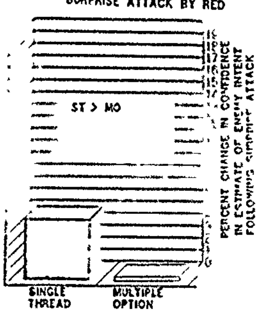
EXPERIMENT I

The first experiment was conducted on the Joint Theater Level Simulator (JTLS) at the wargaming laboratory at the Naval Postgraduate School, August 24 - September 4, 1987 (DCA, 1987). The subjects consisted of 2 random assignments of 14 students to 1 of 2 teams each organized into 5 command cells viz. division, 2 brigades, aviation and discom. Each team participated in 4 counter balanced trails of 3 hours each resulting in a total of 4 contingency planning trials and 4 single thread trials.

As shown in Figure 1, the teams engaged in the multiple option trials performed significantly better on the battlefield. When one examines the reactions and the 2 commanders to the surprise attack, a curious phenomenon appears: the single thread commanders actually increased their certainty about enemy intent following onset of the surprise attack. As shown in Figure 2, whereas the multiple option commanders showed no appreciable change in their estimates of enemy intent before and after the surprise attack, the single thread commanders displayed a significant 5 percent increase in their certainty of enemy intent, i.e., they bolstered their initial false belief. (t=2.00, p .10 for 2df, 1 tailed).

FIGURE 2

BOLSTERING OF BLUE CONFIDENCE
IN ESTIMATE OF ENEMY INTENT FOLLOWING
SURPRISE ATTACK BY RED



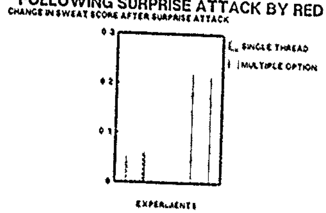
EXPERIMENT I

Furthermore, this surge of false certainty on the part of the single thread commanders was debilitating: The correlation across the treatment groups between the amount of bolstering by the commander and the overall battlefield performance of his headquarters team was -.70, accounting for roughly half of the variability in overall battlefield performance. In terms of battlefield performance in the rear area, where the surprise attach occurred, the correlation with bolstering was even greater (r = -.96), accounting for roughly 90 percent of the variability in the rear exchange ratio.

Finally, experiment I, yielded some surprising findings regarding the subjective workload reported by the Blue team members in response to the surprise attack. Through the use of the Subjective Workload Evaluation Assessment Tool (SWEAT), Blue team members rated the levels of stress, mental effort, and time pressure which they had experienced during the previous half-hour. As shown in Figure 3, whereas the multiple option teams realistically reported a significant increase in subjective workload of 22% during the time period following the surprise attack, the single thread teams, who were less prepared for the attack, reported only a marginal increase of 6% in their subjective workload following the surprise attack. (t=1.53, p=.13 for 2df, 1 tailed).

FIGURE 3

CHANGE IN BLUE TEAM'S SUBJECTIVE WORKLOAD FOLLOWING SURPRISE ATTACK BY RED



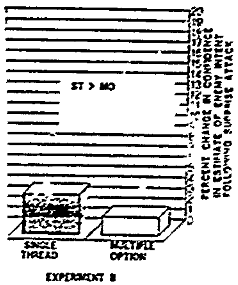
With these findings in mind, we sought to put our experimental hypotheses to another test.

EXPERIMENT II

The second experiment was conducted on the JANUS simulator at the Army Training and Doctrine Command Wargaming Laboratory at the Naval Postgraduate School from January 19 - February 3, 1988 (DCA, 1988). The subjects consisted of 3 student teams of 3 students each organized at the brigade level and led by a seasoned commander from the Army's 101st Airborne Division. Ten trails were successfully completed resulting in a total of 6 contingency planning trials and 4 single thread trails of 3 hours each. As shown in Figure 1, the teams engaged in multiple option trails, again, performed significantly better on the battlefield. Again when one examines the reactions of the 2 commanders to the surprise attack, in this new, more granular context, the same curious phenomenon reappears: whereas the multiple option commanders show no appreciable change in their estimate of enemy intent following the surprise attack, the single thread commanders again significantly increase their confidence in their incorrect estimate of enemy intent. As shown in Figure 4, the single thread commanders increased their certainty of belief in a frontal attack by Red by several percentage points. (t=.52, p=.30 for 8df, 1 tailed).

FIGURE 4

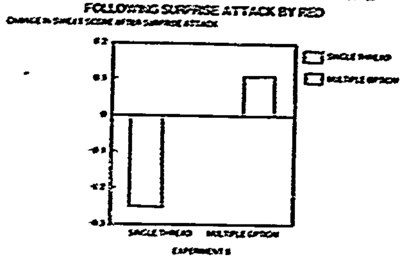
BOLSTERING OF BLUE CONFIDENCE
IN ESTIMATE OF ENEMY INTENT FOLLOWING
SURPRISE ATTACK BY RED



In a further investigation of this bolstering tendency, we examined the change in reported subjective workload by the commanders following the surprise attack. Through use of the SWEAT, the commanders rated the levels of stress, mental effort, and time pressure which they had experimented during the previous half hour. As shown in Figure 5, the multiple option commanders realistically increased their reported subjective workload after the surprise attack by 10%; the single thread commanders actually decreased theirs by 25% in the face of the unanticipated attack. (t=1.87, p .05 for 8df, 1 tailed).

FIGURE 5

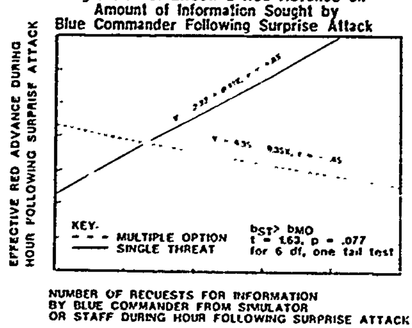
CHANGE IN BLUE COMMANDER'S SUBJECTIVE WORKLOAD
FOLLOWING SURPRISE ATTACK BY RED



Finally, the information search activity of the commanders with and without contingency plans diverged significantly in response to the surprise attack. As shown in Figure 5, during the course of the battle regular observations were made of the number of requests for information the Blue commanders made of their staffs or the simulator itself. For the commanders pursuing multiple option plans, the more information they sought, the better they performed in halting the Red advance during the hour following the surprise attack (r = .45). The opposite is true for the commander pursuing the single thread plan: the more information they sought, the more poorly they fared on the battlefield (r = -.65). The latter finding strongly suggests that the single thread commanders directed their post-surprise information search to the pursuit of false hypotheses regarding the battlefield situation.

FIGURE 6

BIASED INFORMATION SEARCH:
Regression of Effective Red Advance on
Amount of Information Sought by
Blue Commander Following Surprise Attack



III DISCUSSION AND CONCLUSIONS

Our 2 experiments have consistently demonstrated the superiority of multiple option planning over single thread planning under the conditions tested which involved a surprise attack. The odds of obtaining joint results like this by chance are less [illegible] 100. Regarding the cognitive mechanisms [illegible], this relationship, our experiments have [illegible] demonstrated and replicated the tendency of [illegible] single thread commanders to intensify their false belief in enemy intent in the face of the surprise attack. The odds of obtaining results like these by chance are less than 3 in 100. Akin to this intensification of false belief, our experiments have also demonstrated the tendency of single thread commanders to deny any increase in subjective workload as a result of the intelligence reports on the surprise attack. The odds of obtaining results like this by chance are less than 1 in 100. Finally, the results of our more detailed investigation into headquarters information processing in the second experiment demonstrated an unmistakable tendency on the part of the single thread commanders to engage in faulty information processing, readily interpretable as being biased by their pre-existing but false single thread orientation to the battle. The odds of obtaining the latter result by chance are less that 1 in 10.

Why do the multiple option plan commanders behave so "rationally" and why do the single thread commanders behave so curiously in the wartime headquarters setting? We suggest that the single thread commanders were not so much acting so as to maximize Red attrition subject to the constraints of their headquarters resources; rather they were acting to reduce the cognitive dissonance they experienced between their commitment to the single thread plan, with its associated initial force deployment, and the intelligence reports they received regarding the off-axis surprise attack by Red. In doing this, they acted like classic highly committed individuals such as Admiral Kimmel and his group at Pearl Harbor, seeking to "drown out" the dissonance created by the intelligence on the surprise attack by bolstering their prior belief, in this case, that "Red is engaged in a massive frontal assault" and by denying (indeed asserting the contrary in Exp. II) that there was any additional increase in workload or anxiety rightfully devolving from these intelligence reports. As one high ranking military observer rationalized "These reports of an oblique surprise attack just proves that the enemy's intent is really a frontal assault."

Bolstering and Baysian Information Fusion. These experiments and other investigations demonstrate the lack of universality of Baysian updating in human information processing. In both experiments the single thread commanders increased their estimates of the likelihood of a massive frontal assault by Red upon receipt of information to the contrary. This phenomenon of moving one's estimates of the likelihood of an event away from the position advocated by the incoming message rather than towards it has long been called the "boomerang effect" in behavioral science literature. Kiesler (1971) recounts half a dozen experiments of his own and alludes to many by other investigators demonstrating this phenomenon. Hiniker (1983) and others have noted its prevalence in political campaigns and persuasion attempts in a wide variety of contexts.

The boomerang effect is generally characterized by an informational attack resulting in the intensification of the subjects prior cognitive orientation. According to Kiesler, the likelihood of such an effect depends upon the subjects degree of prior commitment to his position, where commitment means the pledging or binding of the individual to behavioral acts and its main effect is to make an act more difficult to undo, deny, distort, or reinterpret. Such acts are more committing to the extent that they are voluntary, public, and important. Viewed in these terms, the commanders in the single thread condition were clearly more committed to receipt of a massive frontal assault by Red than were the multiple option commanders. Hence they boomerang while the contingent planners realistically take account of the surprise reports.

The Perception of Workload. The expression of diminished stress in the reactions of the single thread commanders and their teams to the surprise attack demonstrates the importance of subjectivity in the perception of organizational workload. Clearly an objective assessment would state that having to deal simultaneously with both a frontal assault and an off-axis assault with the same resources entails more time pressure, mental concentration and stress than dealing with the former alone. Yet the single thread teams asserted the opposite. As we have suggested, this subjectivity is part of a larger bolstering and rationalization effort on the part of the committed single thread teams to deal with the reports of the surprise attack.

Biased Information Search. The biased information seeking on the part of the single thread commanders in reaction to the surprise attack demonstrates the untenability of the generalization that the more information gathered the better. Clearly in this situation the key is which hypotheses regarding the battle are being tested and how --- not just how much.

The "rational actor" model of policy decision making in crises has been criticized as providing insufficient explanations of real behavior (Allison, 1971; George, 1974, 1984, 1986; Holsti, 1975; Lebow, 1985). The key usually lies not in calcula- ting the most cost/effective option, but in how information is gathered and processed to provide for option selection. In a survey of presidential decision making during 19 international crises since World War II, from the U. S. decision to cross the 38th parallel in Korea in 1950 through the U. S. decision to attack Cambodian sancturaries in 1970, Herek et.al.

(1987) found strong support for the contribution of biased information seeking to U. S. policy disasters. In recommending 7 criteria for vigilance to avoid defective decision making, the authors emerge with a listing that is remarkably similar to stages in the HEAT and SHOR decision process paradigms. While specifically recommending the use of "monitoring and contingency plans", the essence of the maladies underlying the 7 criteria and the associated failed policy decisions is a bias characterized by a failure to consider or accept information which deals with the negative characteristics of the single preferred option. Psychologically unsatisfying as they may seem, contingent hedging and continuous probing are keys to effective headquarters decision making.

REFERENCES

Graham T. Allison Essence of Decision: Explaining the Cuban Missile Crisis, Boston: Little, Brown & Co., 1971.

Defense Communications Agency Experiment II Report: The Effects of Option Planning and Battle Workload on Command and Control Effectiveness (TR-388)(NPSII) Apr. 1988.

Defense Communications Agency Experiment Report: The Effects of Option Planning and Battle Workload on Command and Control Effectiveness (TR-368) (NPSI) Nov. 1987.

Shel Feldman, (Ed.) Cognitive Consistency: Motivational Antecedents and Behavioral Consequences New York: Academic Press, 1966.

Leon Festinger A Theory of Cognitive Dissonance. Stanford: Stanford University Press, 1957.

Alexander L. George "The Impact of Crisis Induced Stress on Decision Making" in The Medical Implications of Nuclear War, Washington, DC: National Academy of Sciences Press, 1986, pp 529-552.

Alexander L. George "Crisis Management: The Interactions of Political and Military Considerations "Survival," Sep/Oct 1984, pp 223-234.

Alexander L. George "Warning and Response. Theory and Practice" In Y. Evron (Ed.), International Violence: Terrorism, Surprise, and Control Jerusalem: Hebrew University of Jerusalem, 1979, pp 12-24.

Gregory M. Herek, Irving L. Janis, and Paul Huth "Decision Making During International Crises" Journal of Conflict Resolution, Vol. 31, No. 2, Jun 1987, pp 203-226.

Paul J. Hiniker "The Cultural Revolution Revisited: Dissonance Reduction or Power Maximization" China Quarterly, London, June 1983, pp 282-303

O. R. Holsti and A. L. George "The Effects of Stress on the Performance of Foreign Policy Makers," in Political Science Annual: An International Review, Indianapolis: Bobbs-Merrill, 1975, pp 255-319.

Irving L. Janis Groupthink: Psychological Studies of Policy Decisions and Fiascoes, Boston: Houghton Mifflin Co., 1983.

Charles A. Kiesler The Psychology of Commitment: Experiments Linking Belief to Behavior, New York: Academic Press, 1971.

Richard Ned Lebow (In Press) Leadership in Crisis, Science, 1985.

Joseph G. Wohl "Force Management Decision Requirements for Air Force Tactical Command and Control" IEEE Trans. Systems, Man Cybern, 11, 1981, pp 618-639.

SYNTHESIS OF DISTRIBUTED COMMAND AND CONTROL FOR THE OUTER AIR BATTLE *

Stamos K. Andreadakis
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology, Cambridge, MA.

ABSTRACT

The objective is to design distributed Command and Control organizations for the outer air battle. The synthesis problem is formulated as follows: Given the decision-making and information processing necessary for the outer air battle, design the C^2 organization that is accurate, timely, exhibits a task throughput rate that is higher than the task arrival rate, and whose decisionmakers are not overloaded. A simple model of the processes pertinent to the outer air battle has been developed. The model, although an abstraction of the actual naval air operations, retains the fundamental decision-making features.
A new quantitative methodology for the synthesis of C^2 organizations is presented. The methodology consists of four phases: (1) Algorithmic generation of data flow structures in the form of Petri Nets that have specified degrees of redundancy and complexity; (2) Transformation of the data flow structures into decision-making organizations by allocating the functions to individual decisionmakers and then into C^2 organizations by incorporating the supporting systems; (3) Evaluation of the resulting designs using three measures of performance - accuracy, response time, and throughput rate - and a measure of effectiveness; and (4) Modification of the candidate designs to increase their measure of effectiveness.

1. INTRODUCTION

Organizations are formed when the task to be performed exceeds the capabilities of a single decisionmaker. Even when a single person can complete the task, he may not be able to produce a satisfactory response within the time limits imposed by the task, and keep up with the arrival rate of the tasks. The organization designer is faced with the problem of designing an organization that will meet these design specifications and, in addition, assign subtasks or functions to members of the organization so that no one is overloaded. The design has to be robust to accommodate the decision-making styles of different actual decisionmakers that may instantiate the organization at different times.

Consider, for example, the design of an air-traffic control center for a busy airport area. The task cannot be performed by a single controller; several controller stations may be required. The designer has to take into account the uncertainty that is inherent in the task, the need for accurate and timely responses by the controllers, and the need to keep up with the rate of the incoming tasks. But he also has to consider that different controllers will be on duty at any instant of time. While they are all well trained for their tasks, their actual information processing rate and decision-making styles will differ.

The quantitative and qualitative analysis and evaluation of such task-oriented organizations has been the subject of recent research: Drenick (1986); Levis (1984). In the latter work, a model of the interacting decisionmaker with bounded rationality was introduced by Boettcher and Levis (1982), in which the individual members' cognitive workload was computed using N-dimensional Information theory and the Partition Law of Information (Conant, 1976). The organizational architecture, i.e., the allowable interactions among decisionmakers and the protocols that govern them, is described using Petri Nets (Peterson, 1981, Reisig, 1982).

The synthesis of Command and Control organizations is a complex process that must address a multitude of issues. specifically, how to partition the task into functions (or subtasks), how many decisionmakers to select, how to allocate the functions to decisionmakers, how to select the schema of information exchange among the decisionmakers (protocols), what kind of communications hardware is required for the timely transmission of information and data, what the structure of the required databases and the specifications for the respective hardware should be, and how to design decision aids and allocate them to the organization members. Finally, there is the issue of evaluation: how to compute the performance and the effectiveness of the designs and how to select the best design for the task. Consequently, it is necessary to develop a methodology so that the design of C^2 organizations becomes a structured process.

In this paper, a methodology for the synthesis of C^2 organizations is presented. The approach taken in this work decouples the decomposition of the decision-making process and the exchange of data among the functions from the allocation of functions to decisionmakers, and the selection of the supporting systems. Thus, the methodology tackles the synthesis problem at two levels: the *data flow structure level* and the *organization architecture level*.

An algorithm for the generation of data flow structures has been developed; the data flow structures are parameterized by the degree of complexity of the information processing and the degree of redundancy of the information within the structure. The data flow structures are transformed into organization architectures by allocating the functions to decisionmakers and by augmenting the structures to incorporate the supporting systems.

A procedure for the analysis and evaluation of organizational designs is described. The following measures of performance

* This work was supported by the Office of Naval Research under contract No. N00014-K-84-0519 (NR 649-003)

(MOPs) are computed: Accuracy, denoted by the Cost index J; Response Time (Time Delay), denoted by T_r; and Throughput Rate, denoted by R. A global measure, called the measure of effectiveness (MOE), is defined, and the designs are evaluated and compared on the basis of their MOE value.

The processing times associated with the individual information processing and decision-making functions are characterized by probability density functions (pdfs). A method for the computation of the pdf of the organization's response time, and of the pdf of the throughput rate is presented. These lead to the definition of a measure of timeliness and a measure of processing capacity.

2. ORGANIZATION MODEL

2.1 Mathematical Representation

Decision-making organizations can be represented by Petri Nets. In Petri Nets, there are two types of nodes: places, denoted by circles representing signals or conditions; and transitions, denoted by bars, representing processes or events. Places can only be connected to transitions, and transitions can only be connected to places. The execution of a Petri Net is controlled by tokens, which are markers, denoted by dots in the places. A Petri Net is said to execute when a transition fires. A transition can fire only when it is enabled, i.e., when all its input places contain at least one token each. When a transition fires, it removes (consumes) one token from each of its input places and creates (deposits) one token in each of its output places.

A transition may have more than one output places. However, to model decision-making, it is convenient to introduce a special transition*, a decision switch (Figure 1), in which the output places represent alternatives (Tabak and Levis, 1985). When a decision switch fires, a token is deposited in *only one* of its output places. A decision rule associated with this special transition determines the place in which the token is deposited. The rule can be deterministic or stochastic; it can be independent of the attributes of the tokens in the input places or it may depend on them.

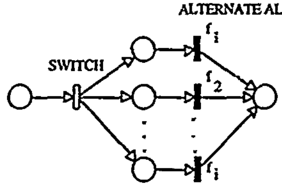


Figure 1. Decision Switch

For example, a decisionmaker may access a decision aid, the decision in this case is whether to access or not the decision aid. Similarly a decisionmaker may have two algorithms or procedures available to process the information. In the case of situation assessment, one algorithm may be complex and the other simple. In the case of developing courses of action, one algorithm may be more detailed and exhaustive, while the other more crude and simplistic. The decision in this case is which of the two algorithms to use. This decision varies among decisionmakers (difference in style) and may be quantified by the relative frequency of algorithm selection. The decisionmaker's workload, the accuracy, the response time, and the throughput rate of the organization are affected by these decisions.

The decisionmakers are assumed to be limited in the number of functions (or subtasks) they may perform at the same time. This limitation is represented by introducing a place for each decisionmaker, called the resource availability place, which is an output place of the last transition and an input place of the first transition assigned to the decisionmaker. The number of functions a decisionmaker may perform at the same time are represented by the number of tokens initially deposited in the resource availability place, i.e., by its initial marking.

2.2 Decision-making Model and Decision Strategies

The decisionmakers are assumed to be well trained and to be able to use more than one procedure to perform some functions (Boettcher and Levis, 1982, Levis and Boettcher, 1983). The selection of one procedure from a set of procedures is modeled in the Petri Net by a switch. When the decisionmaker selects always the same algorithm (when the selection rule is independent of the input) or the same algorithm for each input element (when the selection rule is dependent on the input value), he is implementing a *pure decision strategy*.

Mixed strategies are obtained when the decisionmaker uses a mix of pure strategies (Owen,1968); the mixed strategies are characterized by the relative frequency of use of the pure strategies. When all decisionmakers of the organization select (implement) their mixed strategies, a *behavioral decision strategy* is obtained.

If the probabilities that correspond to the relative frequency of algorithm use are discretized, then a finite number of mixed strategies is defined for each decisionmaker, and consequently, a finite number of behavioral strategies for the organization is obtained.

3. ANALYSIS OF DECISION-MAKING ORGANIZATIONS

3.1 Workload and Bounded Rationality

Workload represents the amount of mental effort expended by the decisionmakers in order to perform their assigned tasks. The analytical framework for the computation of a surrogate for workload is N-dimensional information theory (Reisbeck, 1963, Shannon and Weaver, 1963). This surrogate, denoted by G, is the total activity term in the Partition Law of Information (Conant, 1976). The total activity has units of bits/symbol.

The value of G depends on several factors. First, it depends on the uncertainty of the organization's task, as modeled by the probability distribution p(x) associated with the input set {x}. It depends also on the structure of the organization - the interactions among decisionmakers - and on the algorithms used to represent the various processing functions, such as situation assessment, courses of action development, and response selection. Finally, it depends on the internal decision strategies of each individual decisionmaker. Indeed, in the analysis of organizational performance that follows, for a given organizational design, the independent variables are the decisions of each decisionmaker and the dependent variables are the workload and the measures of performance.

* The use of a special transition can be avoided, if Predicate Transition Nets are used in place of ordinary Petri Nets. However, such a generalization is not necessary for this work.

and g(t).

$$h(t) = f(t) * g(t). \tag{7}$$

For an information flow path (if assumed that its transitions do not receive data from other paths), the total delay is the sum of the delays of the individual transitions and consequently the pdf of the total delay is obtained by repeated convolutions.

For two functions which are concurrently active, i.e., on parallel paths (Figure 3), the total delay is the maximum of the corresponding delays. The pdf h(t) of the total delay, if the two delays are independent random variables, is obtained as follows:

$$h(t) = f(t)\, G(t) + F(t)\, g(t) \tag{8}$$

where f(t) and g(t) are the pdfs of each delay and F(t) and G(t) are the corresponding cumulative distribution functions.

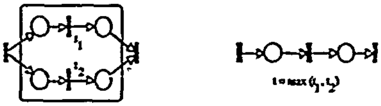


Figure 3. Concurrently Active Functions

For two concurrently active information flow paths, whose transition delays are independent random variables, two cases must be considered:

1) the paths do not have transitions in common
2) the paths have transitions in common

In the first case, the pdf of the total delay of each path is computed, and then the pdf of the maximum of the path delays is obtained. In the second case, if the time delay of the common transition is τ with pdf $f_\tau(t)$, while the total time delays of the unique transitions on each path are τ_1 and τ_2, with corresponding pdfs $g_{\tau 1}(t)$ and $g_{\tau 2}(t)$, first compute the pdf $g_{\tau max}(t)$ of the maximum delay $\tau_{max} = \max(\tau_1, \tau_2)$ of the unique transitions

$$g_{\tau max}(t) = g_{\tau 1}(t)\, G_{\tau 2}(t) + G_{\tau 1}(t)\, g_{\tau 2}(t) \tag{9}$$

and next compute the pdf of the total delay of the two paths by convolving $f_\tau(t)$ and $g_{\tau max}(t)$. Using these procedures, the pdf of each complete path is computed.

Consider two alternate procedures (i.e. substitutes for one another) or two complete paths active with relative frequency of use p_1 and $p_2 = 1 - p_1$, and corresponding delay pdfs f(t) and g(t). Then the pdf h(t) of the total delay is given by:

$$h(t) = p_1 f(t) \quad p_2 g(t) \tag{10}$$

3.4 Computation of the pdf of Throughput Rate

The throughput rate of the organization is equal to the minimum of the processing rates of the sets of functions performed by individual decisionmakers, and the processing rates of sets of functions performed by several decisionmakers in an interleaved pattern. These sets of functions correspond to the transitions of the directed elementary circuits of the net. The processing rate of a directed elementary circuit is the inverse of the total processing time of the circuit. The total processing time, t, of a directed elementary circuit is equal to the sum of the processing times of its transitions, divided by the token content, C, of the circuit (Ramchandani, 1974). The token content is equal to the sum of the tokens initially placed in the resource availability places of the circuit.

$$t = (\Sigma t_i)/C \tag{11}$$

In the case of stochastic processing times, the pdf g(r) of the processing rate r of a directed elementary circuit is given by

$$g(r) = (1/r^2)\, f(1/r) \tag{12}$$

where f(t) is the pdf of the processing time of the circuit. For two directed elementary circuits with no transitions in common, if the processing rates r_1 and r_2 are independent random variables with pdfs f(r) and g(r), the pdf of the minimum processing rate h(r) is

$$h(r) = f(r)\,[1-G(r)] + [1-F(r)]\, g(r) \tag{13}$$

where F(r) and G(r) are the cumulative distribution functions.

If the elementary circuits have transitions in common, then their processing rates are correlated. Let two such circuits have one transition in common with processing time τ, having pdf g(t), and one unique transition in each circuit with corresponding processing times τ_1 and τ_2, with pdfs $q_{\tau 1}(t)$ and $q_{\tau 2}(t)$ (Figure 4). Assume also that the two circuits have the same token content C. Then the pdf of the maximum processing times of the two circuits

$$\tau_{max} = (\tau + \max(\tau_1, \tau_2))/C \tag{14}$$

is computed as follows: first compute the pdf $q_{\tau *}(t)$ of $\tau * = \max(\tau_1, \tau_2)$ by:

$$q_{\tau *}(t) = q_{\tau 1}(t)\, Q_{\tau 2}(t) + Q_{\tau 1}(t)\, q_{\tau 2}(t) \tag{15}$$

Then convolve $q_{\tau *}(t)$ and g(t)

$$s(t) = q_{\tau *}(t) * g(t) \tag{16}$$

and scale the pdf s(t) to obtain the pdf $f_{tmax}(t)$ of the maximum processing time of the two circuits

$$f_{tmax}(t) = C\, s(Ct) \tag{17}$$

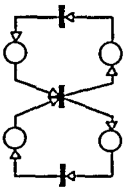


Figure 4. Two circuits with one common transition

Finally, the pdf h(r) of the minimum of the processing rates of the two circuits is

$$h(r) = (1/r^2)\; f_{tmax}(1/r) \tag{18}$$

The computation in the case of different token content is similarly developed.

The qualitative notion that the rationality of the human decisionmaker is bounded, (March, 1978), has been modeled as

$$F \leq F_o \qquad (1)$$

where F is the information processing rate of individual decisionmakers (in bits/sec), and F_o is the maximum information processing rate that characterizes individual decisionmakers. Since the processing time, t, is computed by

$$t = G/F \qquad (2)$$

the minimum processing time t_o, corresponds to the maximum processing rate F_o.

$$t_o = G/F_o \qquad (3)$$

In evaluating decision-making organizations, it is of interest to compute the minimum response time, and the maximum throughput rate, that correspond to the maximum information processing rate F_o.

The maximum processing rate F_o varies among decisionmakers. If the pdf $h(F_o)$ of F_o is known, then the pdf $q(t_o)$ of the minimum processing time of each transition can be obtained:

$$q(t_o) = (G/t_o^2)\, h(G/t_o) \qquad (4)$$

3.2 Measures of Performance

The measures of performance considered in this paper are: accuracy, response time, and throughput rate.

Accuracy quantifies the degree to which the actual organization response, Y_i, matches the desired or ideal response Y_{di}. A cost $C(Y_i, Y_{di})$ is assigned to the discrepancy of Y_i and Y_{di} (Levis, 1984). This cost is computed for each input task, x, and each decision strategy. The accuracy measure J is the expected value of the cost and is computed using the probability distribution of the input tasks (Figure 2).

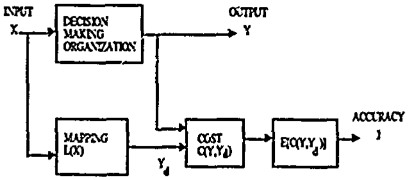


Figure 2. Computation of Accuracy

The response time or time delay of an organization is the time elapsed between sensing the input and producing an output. The expected response time (expected time delay) is a measure of performance that can be used to assess the timeliness of an organization's response.

Timeliness expresses the ability of organizations to produce a response to a given input within an allotted time. The allotted time is a time interval (T_{min}, T_{max}). T_{min} is a time threshold such that if the organization acts in response to the input before the threshold, a cost is incurred through the expense of assets or supplies too early, resulting in the decrease of the probability of success of the response. T_{max} is a time threshold such that if the organization acts in response to the input after the threshold, there will not be enough time left for the implementation of the response. If the expected response time is within the interval (T_{min}, T_{max}) the response is timely. However, this measure of performance does not take into account the variance of the response time. A better measure of timeliness, T, is the probability that the response time, T_r, lies inside the interval (T_{min}, T_{max}), i.e.

$$T = P(T_{min} < T_r < T_{max}) \qquad (5)$$

The Throughput rate of the organization is the maximum task processing rate that can be sustained, without queueing of the inputs, or queueing of information at any stage of processing. For the case of stochastic processing times, the pdf of the throughput rate, R, can be computed and a measure of processing capacity, S, can be defined as:

$$S = P(R > R_o) \qquad (6)$$

where R_o is the task arrival rate. Alternatively, we may be interested in computing the response time and the throughput rate that correspond to the minimum value of the rationality threshold, i.e., $(F_o)_{min}$.

To each behavioral strategy, corresponds a set of values of the measures of performance (MOPs), which defines a vector in the MOP space. Thus, the mathematical models of accuracy, response time, and throughput rate, map the decision strategies into the performance space. As the behavioral strategies change, this vector sweeps a locus in the MOP space, the organization locus. The requirements on the MOPs also define a locus in the MOP space; the requirements locus. Organizational architectures can be evaluated by comparing the organization locus to the requirements locus. Different organizational architectures can be compared on the basis of their corresponding loci.

3.3 Computation of the pdf of Response Time

In the Petri Net representation of an organization, the input (source) and the output (sink) nodes are represented by transitions. Information flow paths are the paths emanating from the source transition and arriving at the output transition. The presence of decision switches in the net, with the position of each switch determined by the internal decision strategies, results in some transitions being active during the processing of any task, and in some being inactive. Therefore for each behavioral strategy, corresponding to pure decision strategies of the decisionmakers, some information flow paths are active (transmitting information) while others are inactive. A set of concurrently active paths is called a complete path.

The simple paths and the complete paths may be identified either by an algorithm developed by Jin (Jin et al., 1986) for acyclical structures, or by an algorithm that computes the elementary directed circuits of the net, developed by Martinez and Silva (1980), and improved by Alaiwan and Toudic (1985).

If a pdf is assigned to the processing time of each processing algorithm, to the transmission delay for each communication process, and to the access time for each decision support system (all of which are represented by transitions on the net), then the pdf of the response time of the organization is computed as follows:

For two cascaded functions with corresponding delay pdfs f(t) and g(t), the total delay is the sum of the two delays. Therefore the pdf h(t) of the total delay is given by the convolution of f(t)

3.5 Measure of Effectiveness

Measures of Effectiveness quantify the degree to which an organization meets its requirements. A Measure of Effectiveness, Q, can be defined by the ratio of the number of behavioral strategies that satisfy the requirements to the total number of behavioral strategies.

$$Q = \frac{\text{number of behavioral strategies satisfying the requirements}}{\text{total number of behavioral strategies}} \tag{19}$$

Recall that individual decisionmakers differ in style, i.e., they tend to use different mixed strategies. In this respect the measure of effectiveness is a measure of robustness of the organization with respect to the different styles of individual decisionmakers.

4. SYNTHESIS OF DECISION-MAKING ORGANIZATIONS

Given a complex information processing and decisionmaking task, there exists a multitude of ways to partition the processing of a task into subtasks (functions), to define the schema of information exchange among the functions, to allocate functions to decisionmakers, and to specify the supporting systems (software and hardware).

4.1 Synthesis Problem Formulation

The synthesis problem is formulated as follows. Given a mission and a set of tasks to be performed, design a decision-making organization that is accurate, timely, has a task throughput rate higher than the task arrival rate, and whose decisionmakers are not overloaded (Andreadakis, 1988). The quantitative formulation is:

Accuracy greater than or equal to a given threshold, or equivalently, expected cost J less than or equal to some threshold J_o:

$$J \leq J_o \tag{20}$$

Timeliness measure greater than or equal to some threshold T_o:

$$T \geq T_o \tag{21}$$

Processing capacity measure greater than or equal to some threshold S_o:

$$S \geq S_o \tag{22}$$

under the constraint that decisionmakers are not overloaded, i.e., that each decisionmaker's information processing rate is less than or equal to his rationality threshold $(F_o)_i$:

$$F_i \leq (F_o)_i \tag{23}$$

An alternative formulation is obtained when the second and third requirements are expressed as:

Response time T_r less than or equal to some threshold $(T_r)_o$:

$$T_r \leq (T_r)_o \tag{24}$$

Throughput rate R greater than the task arrival rate R_o:

$$R > R_o \tag{25}$$

In this work the concepts of *data flow structure* (DFS), *decision-making organization* (DMO) and *Command and Control organization* (C^2O) are contrasted, and are employed in the development of a structured methodology for the synthesis of Command and Control organizations.

The DFS is a representation of the connectivity of the functions performed by the organization and illustrates the flow of information from function to function. The DMO is a DFS whose functions have been allocated to decisionmakers. Finally, a C^2O is a DMO which is supported by hardware and software (the C^3 system) in the execution of its tasks.

In the two level design procedure, the data flow structure design focuses on information processing schemata, while the organization architecture design focuses on function allocation to decisionmakers and on the development of the supporting systems.

The synthesis methodology has four phases (Figure 5). In phase 1, the procedure for generating data flow structures produces a set of candidate designs. In phase 2, each data flow structure is augmented and transformed to one or more decision-making organizations, in which the functions have been allocated to decisionmakers, and then to the corresponding

Command and Control organizations by incorporating the supporting hardware and software. In phase 3, the measures of performance and the measure of effectiveness are computed. The designs obtained in this manner, are revised in phase 4, to increase their measure of effectiveness by changing function allocation, introducing or modifying decision aids and

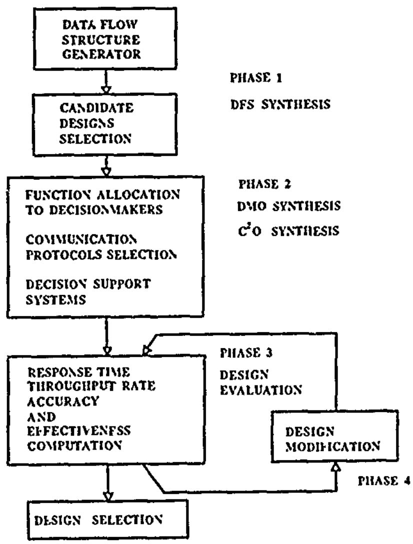


Figure 5. Flowchart of Synthesis Methodology

databases, and improving the communication links. Finally, a C^2 organization is selected on the basis of the highest MOE value.

4.2 Data Flow Structure Design

The information processing is decomposed into five stages (functions): Initial Processing [IP], Data Fusion [DF], Middle Processing [MP], Results Fusion [RF], and Final Processing [FP]. As data are received, they are processed in the IP stage to assess the situation. Information (local or partial situation assessments) of several IP stages are combined (fused) in the DF stage, which produces global situation assessment.

The global situation assessment is fed to the MP stage which develops results (options or courses of action). The results are combined (fused) in the RF stage to eliminate conflicting or infeasible options - courses of action. Finally, a response is selected from the available options in the FP stage.

Each processing stage is represented in the Petri Net of the data flow structure by a transition. An information flow path with all five stages defines a flow type 1 (Figure 6a). Note that some IP transitions may provide results for fusion at an RF stage (DF and MP stages null) (Figure 6c), while some MP transitions may generate output of the organization (RF and FP stages null) (Figure 6b). An information flow path of the latter type defines flow type 2, while one of the former type defines a flow type 3.

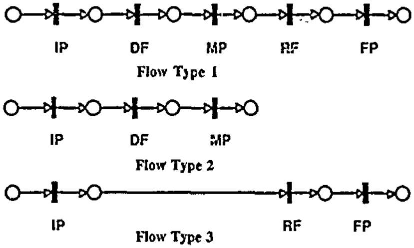


Figure 6. Basic Flow Types

The data flow structures are classified according to the flow types of their information flow paths. If all the paths are of flow type 1, then the DFS belongs to class 1. If some paths are of flow type 1 and some of flow type 2, the DFS class is 12. The feasible classes are: 1, 2, 3, 12, 13, and 123. Class 23 is infeasible because the flow type 2 information paths have data for fusion and DF transitions, while the flow type 3 information paths have results for fusion and RF transitions; and hence flow type 2 and flow type 3 paths cannot exchange information. A DFS with all three flow types (class 123) is shown in Figure 7.

The grammar rules for the connectivity of the processing transitions are:

- exactly one MP node can receive data from a DF node
- exactly one FP node can receive data from an RF node
- one IP transition for each input to the organization
- one FP transition for each output of the organization

The generation of data flow structures takes into account the complexity and redundancy of information processing that is required by the task, and the organization's objectives.

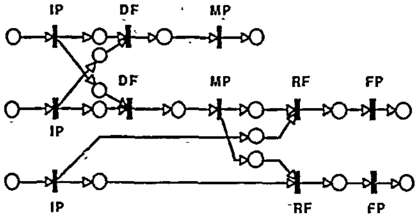


Figure 7. Data Flow Structure with all three Flow Types.

Depending on the degree of centralization of decision-making for global situation assessment and the magnitude of the geographical area for which global situation assessment is desired, the data fusion stage may be more or less complex. Similarly, depending on the degree of centralization for global response selection, and the magnitude of the geographical area where the response needs to be coordinated, the results fusion stage may be more or less complex.

The *degree of complexity* of a DF transition is defined as the number of transitions that feed data to the DF transition. The degree of complexity of the DF stage is defined as the maximum of the degrees of complexity of the DF transitions. The term complexity is justified by the observation that the more data that are fed to a fusion node, the more complex the processing that takes place.

The need for redundancy of information within the structure arises from survivability considerations and topological factors. The *degree of redundancy* of an IP transition is defined as the number of fusion stages that receive the output data of the IP transition. The degree of redundancy of the DF stage is defined as the maximum of the degrees of redundancy of the IP transitions. The term redundancy is justified by the fact that the same information is communicated to more than one fusion nodes, and is therefore redundant in the data flow structure.

The degree of complexity of a RF transition, the degree of redundancy of a MP transition, and the degrees of complexity and redundancy of the RF stage are similarly defined. A data flow structure with degree of complexity $c_1 = 2$ and degree of redundancy $r_1 = 2$ of the DF stage, and degree of complexity $c_2 = 3$ and degree of redundancy $r_2 = 3$ of the RF stage is shown in Figure 8.

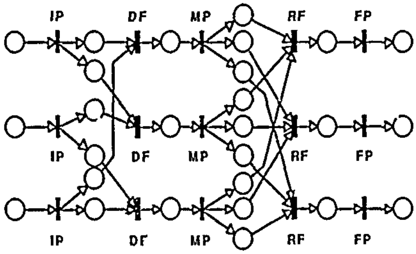


Figure 8. Data Flow Structure $c_1 = 2, r_1 = 2, c_2 = 3, r_2 = 3$

4.3. Data Flow Structure Generation Algorithm

The algorithm for the generation of data flow structures produces the incidence matrix of the corresponding Petri Net, and has seven steps.

The design parameters are:

- the number n_1 of IP transitions that provide data to the DF stage (n_1 less than or equal to the number of IP transitions)
- the number k_2 of MP transitions that provide results to the RF stage (k_2 less than or equal to number of MP transitions)
- the degrees of complexity c_1 and c_2 of the DF and RF stage (less than or equal to the number of transitions that provide information for fusion at the corresponding stage) and
- the degrees of redundancy r_1 and r_2 of the DF and RF stage (less than or equal to the number of processing assets)

Table 1. Transition sets

set	transition type
T_1	initial processing
T_2	data fusion
T_3	middle processing
T_4	results fusion
T_5	final processing

Table 2. Place sets

set	place type
P_1	input places to IP transitions
P_2	output places of IP transitions which are input places to DF transitions
P_3	output places of DF transitions which are input places to MP transitions
P_4	output places of IP transitions which are input places to RF transitions and output places of MP transitions which are input places to RF transitions
P_5	output places of RF transitions which are input places to FP transitions
P_6	output places of MP transitions which are outputs of the DFS
P_7	output places of FP transitions

The incidence matrix has block form: five sets of transitions (Table 1) and seven sets of places (Table 2) are defined. Thus, the incidence matrix is composed of 35 blocks (Figure 9). Each block is denoted by P_iT_j, corresponding to place set P_i and transition set T_j. The flowchart of the algorithm is depicted in Figure 10.

In order to generate data flow structures in a consistent, methodical way, the design parameters are varied between the minimum and maximum value they may obtain.

Step 1: Select the class of the data flow structure.

Step 2: Select the number n_1 of initial processing (IP) transitions that provide data for fusion (DF stage). Let n_2 be the number of initial processing (IP) transitions that provide results for fusion (RF stage). The total number n of IP transitions is:

$$n = n_1 + n_2 \qquad (26)$$

$$
\begin{array}{ccccc}
\begin{bmatrix} P_1T_1 \\ n \times n \end{bmatrix} & \begin{bmatrix} P_1T_2 \\ n \times k \end{bmatrix} & \begin{bmatrix} P_1T_3 \\ n \times k \end{bmatrix} & \begin{bmatrix} P_1T_4 \\ n \times m \end{bmatrix} & \begin{bmatrix} P_1T_5 \\ n \times m \end{bmatrix} \\
\begin{bmatrix} P_2T_1 \\ p \times n \end{bmatrix} & \begin{bmatrix} P_2T_2 \\ p \times k \end{bmatrix} & \begin{bmatrix} P_2T_3 \\ p \times k \end{bmatrix} & \begin{bmatrix} P_2T_4 \\ p \times m \end{bmatrix} & \begin{bmatrix} P_2T_5 \\ p \times m \end{bmatrix} \\
\begin{bmatrix} P_3T_1 \\ k \times n \end{bmatrix} & \begin{bmatrix} P_3T_2 \\ k \times k \end{bmatrix} & \begin{bmatrix} P_3T_3 \\ k \times k \end{bmatrix} & \begin{bmatrix} P_3T_4 \\ k \times m \end{bmatrix} & \begin{bmatrix} P_3T_5 \\ k \times m \end{bmatrix} \\
\begin{bmatrix} P_4T_1 \\ q \times n \end{bmatrix} & \begin{bmatrix} P_4T_2 \\ q \times k \end{bmatrix} & \begin{bmatrix} P_4T_3 \\ q \times k \end{bmatrix} & \begin{bmatrix} P_4T_4 \\ q \times m \end{bmatrix} & \begin{bmatrix} P_4T_5 \\ q \times m \end{bmatrix} \\
\begin{bmatrix} P_5T_1 \\ m \times n \end{bmatrix} & \begin{bmatrix} P_5T_2 \\ m \times k \end{bmatrix} & \begin{bmatrix} P_5T_3 \\ m \times k \end{bmatrix} & \begin{bmatrix} P_5T_4 \\ m \times m \end{bmatrix} & \begin{bmatrix} P_5T_5 \\ m \times m \end{bmatrix} \\
\begin{bmatrix} P_6T_1 \\ k_1 \times n \end{bmatrix} & \begin{bmatrix} P_6T_2 \\ k_1 \times k \end{bmatrix} & \begin{bmatrix} P_6T_3 \\ k_1 \times k \end{bmatrix} & \begin{bmatrix} P_6T_4 \\ k_1 \times m \end{bmatrix} & \begin{bmatrix} P_6T_5 \\ k_1 \times m \end{bmatrix} \\
\begin{bmatrix} P_7T_1 \\ m \times n \end{bmatrix} & \begin{bmatrix} P_7T_2 \\ m \times k \end{bmatrix} & \begin{bmatrix} P_7T_3 \\ m \times k \end{bmatrix} & \begin{bmatrix} P_7T_4 \\ m \times m \end{bmatrix} & \begin{bmatrix} P_7T_5 \\ m \times m \end{bmatrix}
\end{array}
$$

Figure 9. Block Form of Incidence Matrix

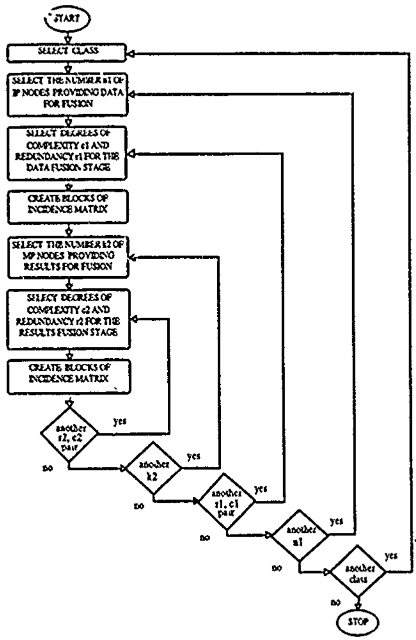


Figure 10. Flowchart of Data Flow Structure Generation Algorithm

Step 3: Select the degree of complexity c_1 and the degree of redundancy r_1 of the DF stage. The number p of output places of IP transitions that belong to the set P_2 is:

$$p = n_1 r_1 \tag{27}$$

and the number k of data fusion transitions is:

$$k = n_1(r_1/c_1) \tag{28}$$

For the pair (r_1,c_1) to be feasible, i.e., for all transitions of the stage to have the same degree of complexity and degree of redundancy, the number k must be integer. Another constraint on k is that it be no larger than the number of available processing assets. Since each DF transition is connected to one middle processing (MP) transition, the number of MP transitions is also k.

Step 4: Since one IP transition is connected to each place that represents an input to the organization, and exactly one MP transition is connected to each output place of a DF transition, the diagonal elements of blocks P_1T_1, P_3T_3 are equal to -1, while the non diagonal elements are equal to 0.

Each DF transition has exactly one output place; thus, the diagonal elements of block P_3T_2 are equal to 1, while the non diagonal elements are equal to 0.

The elements B_{ij} of block P_2T_1 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = 1, 2, \ldots, n_1 \text{ and} \\ & i = (j-1)r_1 + s,\ s = 1, 2, \ldots, r_1 \\ 0 & \text{otherwise} \end{cases} \tag{29}$$

The elements B_{ij} of block P_2T_2 obtain their values according to

$$B_{ij} = \begin{cases} -1 & \text{if place } i \text{ is connected to transition } j \\ 0 & \text{otherwise} \end{cases} \tag{30}$$

The elements of all the other blocks involving place sets P_1, P_2, P_3 are equal to 0.

Step 5: Select the number k_2 of MP transitions that provide results for fusion (at the RF stage). Let k_1 be the number of middle processing transitions that produce outputs. The total number of MP transitions is:

$$k = k_1 + k_2 \tag{31}$$

Step 6: Select the degree of complexity c_2 and the degree of redundancy r_2 of the RF stage. The number q of output places of IP transitions and MP transitions that belong to the set P_4 is:

$$q = (n_2 + k_2)\, r_2 \tag{32}$$

and the number of results fusion transitions, m, is.

$$m = (n_2 + k_2)(r_2/c_2) \tag{33}$$

For the pair (r_2,c_2) to be feasible, i.e., for all transitions of the stage to have the same degree of complexity and degree of redundancy, m must be integer. The second constraint on m is

$$m \leq a \tag{34}$$

where a is the number of available processing assets. Since each RF transition is connected to one FP transition, the number of FP transitions is also m.

Step 7: The elements B_{ij} of block P_4T_1 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = n_1+1, n_1+2, \ldots, n \text{ and} \\ & i = [(j-n_1)-1]r_2 + s,\ s = 1, 2, \ldots, r_2 \\ 0 & \text{otherwise} \end{cases} \tag{35}$$

The elements B_{ij} of block P_4T_3 obtain their values according to

$$B_{ij} = \begin{cases} 1 & \text{for } j = k_1+1, k_1+2, \ldots, k \text{ and} \\ & i = [n_2 + (j-k_1)-1]r_2 + s,\ s = 1, 2, \ldots, r_2 \\ 0 & \text{otherwise} \end{cases} \tag{36}$$

The elements B_{ij} of block P_4T_4 obtain their values according to

$$B_{ij} = \begin{cases} -1 & \text{if place } i \text{ is connected to transition } j \\ 0 & \text{otherwise} \end{cases} \tag{37}$$

Each RF transition has exactly one output place, thus, the diagonal elements of blocks P_5T_4, are equal to 1, while the non diagonal elements are equal to 0.

Each FP transition has exactly one input place; consequently, the diagonal elements of block P_5T_5 are equal to -1, while the non diagonal elements are equal to 0.

Exactly one place representing an output of the DFS is connected to an MP transition which produces a DFS output; likewise, exactly one output place is connected to each FP transition. Hence, the elements B_{ij} of blocks P_6T_3 and P_7T_5, with $i = j$, are equal to 1, and the other elements are equal to 0.

The elements of all the other blocks involving place sets P_4, P_5, P_6 and P_7 are equal to 0.

4.4 Data Flow Structure Selection

Several data flow structures are generated by the algorithm. In order to select the feasible structures, i.e., those that are appropriate for the task, the designer must consider the suitability of the structure to the information processing required by the task. Consequently, the algorithms that implement the processing functions must be developed, and then be associated with the transitions of each candidate structure. During this stage, some links may be removed from the structure. If it is not possible to associate the algorithms with the transitions of a structure, then the structure is discarded.

4.5 Organization Architecture Design

From each data flow structure, one or more decision-making organizations (DMOs) may be developed through function allocation to decisionmakers. Functions allocated to a decisionmaker must observe three requirements:

1) must be connected through an input-output relationship, i.e., the output of the one must be the input to the other, so that the decisionmaker processes information relevant to the same subtask;

2) must belong to different slices (Fernandez and Thiagarajan, 1984) of the Petri Net, so that they observe concurrency, and

3) must conform to the specialization of the decisionmaker.

When a set of functions is allocated to a decisionmaker, a resource availability place is introduced. The addition of these places and of their links, creates the directed elementary circuits of the net, which are used in the throughput rate computations

The transitions of the DFS are in general macro-transitions; they may have internal structure as in the case of functions performed by alternate algorithms. At this point the macro-transitions are substituted by the subnets that they represent.

Next, each DMO is transformed into a C^2O by incorporating the supporting decision systems and the communication links. The data flow structure is augmented by adding the transitions that represent the communication processes and the decision support systems access, and of the places that represent the corresponding protocols. In general, the decisionmakers may or may not use the decision support systems; therefore switches must be introduced to depict the choices available. The switches and the corresponding strategies enable the modeling of the decision-making styles of individual decisionmakers.

4.6 Design modification rules

If the computed Measure of Effectiveness is not satisfactory, then the organization is modified in order to increase the MOE value. The procedure for the modification depends on the location of the organization locus with respect to the requirements locus The existing cases are shown in Table 3.

Table 3. Design Modification Cases

case	J<Jo	$T>T_0$ $T_r<T_{ro}$	$S>S_0$ $R>R_0$	must improve	modification required
1	false	true		accuracy	introduce decision aid
2	true	false		response time	better communications improve database access improve decision aids
3	false	false		accuracy and response time	introduce decision aid and better communications improve database access
4	true	true	false	throughput rate	modify function allocation more processing channels

5. APPLICATION AND RESULTS

The application of the synthesis methodology will be illustrated through the design of Command and Control organizations for the outer air battle. Three C^2 assets are considered: two airborne warning radar aircraft (E2C) and the Combat Information Center (CIC) on the carrier.

A simple model has been developed for the information processing and decision-making pertinent to the outer air battle It should be noted that the model is an abstraction of the actual processes and does not necessarily reflect real naval air operations; it can however be modified to represent reality.

The model presumes that the carrier has four squadrons of interceptor aircraft. Two E2Cs are airborne patroling their assigned sectors One squadron of interceptors is assigned to each E2C, and the other two squadrons are free assets that will be allocated to the appropriate sector(s) depending on the strength of the incoming raid.

The objective of the organization is to develop and implement appropriate plans to engage the incoming threats before they reach the weapons release line Each of the two E2Cs collects information from the area that it surveils, performs situation assessment, develops courses of action, and selects one response from the developed courses of action. Global considerations necessitate the exchange of information between the two E2Cs and possibly the CIC, in order to resolve conflicting courses of action, to allocate assets, and coordinate the response execution. In this example, the vectoring of interceptors to the threats has not been modeled

The model used for this example, incorporates the following functions:

- *local (sector) situation assessment*: classification of enemy aircraft based on their signature and air speed, estimation of number of threats and distance from the E2C.
- *global situation assessment:* estimation of raid strength in both sectors.
- *local (sector) courses of action development*: generation of plans -options- depending on the number and type of aircraft in the sector.
- *global response selection (global resources allocation)*: the free assets are assigned to the sectors, or they remain in the inner battle region
- *local (sector) response selection*: one option is chosen from the developed cources of action, given the available assets.

The complete set of data flow structures generated by the algorithm is given in Table 4. Four representative structures are depicted in Figures 11 through 14. Two of these structures, DFS 7 shown in Figure 11, and DFS 11 depicted in Figure 12, will be used to apply phases two and three of the synthesis methodology.

Table 4. Generated Data Flow Structures

class	c_1	r_1	c_2	r_2	DFS
2	2	1			DFS1
2	2	2			DFS2
2	2	3			DFS3
1	2	1	1	2	DFS4
1	2	1	1	3	DFS5
1	2	2	2	1	DFS6
1	2	2	2	2	DFS7
1	2	2	2	3	DFS8
1	2	3	2	2	DFS9
1	2	3	3	1	DFS10
1	2	3	3	2	DFS11
1	2	3	3	3	DFS12
12	2	3	2	1	DFS13
12	2	3	2	2	DFS14

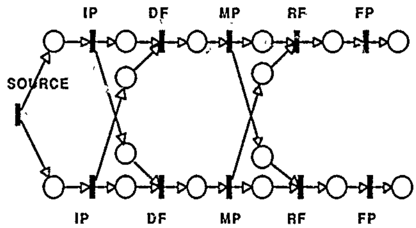


Figure 11. Data Flow Structure 7; $r_1=2, c_1=2, r_2=2, c_2=2$

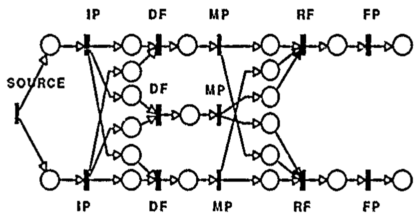


Figure 12. Data Flow Structure 11; $r_1=3, c_1=2, r_2=2, c_2=3$

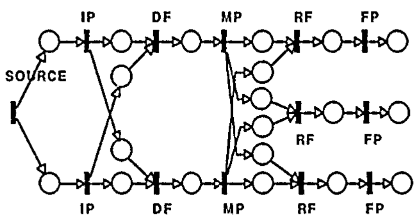


Figure 13. Data Flow Structure 8; $r_1=2, c_1=2, r_2=3, c_2=2$

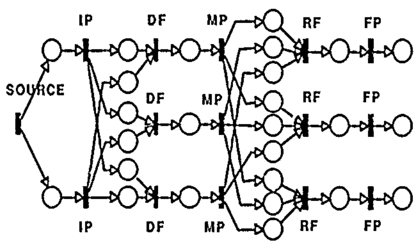


Figure 14. Data Flow Structure 12; $r_1=3, c_1=2, r_2=3, c_2=3$

Two algorithms -procedures- were created for the development of local courses of action: one exhaustive and one crude. Note that the differences in decision-making style, in this example, are manifested in the courses of action development function. The detailed data flow structures, corresponding to Figures 11 and 12, after the elimination of some links that are not required, are shown in Figures 15 and 16.

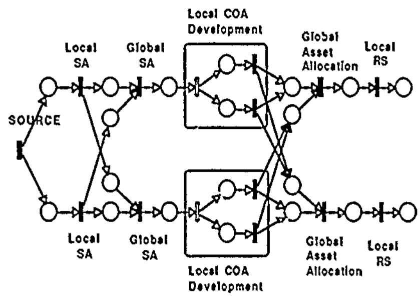


Figure 15. Detailed Data Flow Structure 7

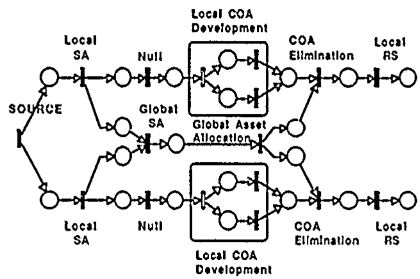


Figure 16 Detailed Data Flow Structure 11

In the data flow structure depicted in Figure 15, the information processing is performed by the personnel of the two E2Cs, while in the data flow structure shown in Figure 16, the CIC personnel participates in the decision-making process by performing the global functions.

From each data flow structure, two Command and Control organizations were developed through different allocation of the functions performed by the E2C personnel: in the first organization, one decisionmaker performs all the functions, while in the second the functions are allocated to two decisionmakers in series. The corresponding Command and Control organizations are depicted in Figures 17, 18, 19, and 20

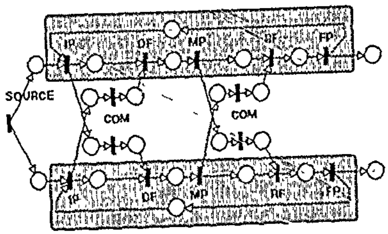


Figure 17. Organization 1; derived from DFS7, one decisionmaker per E2C

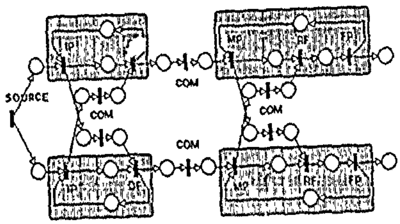


Figure 18. Organization 2; derived from DFS7, two decisionmakers per E2C

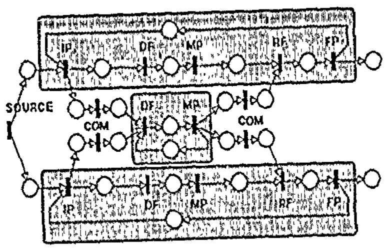


Figure 19. Organization 3; derived from DFS11, one decisionmaker per E2C

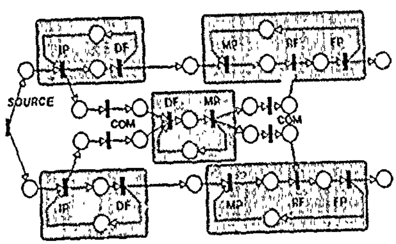


Figure 20. Organization 4; derived from DFS11, two decisionmakers per E2C

The measures of performance, namely the accuracy J, the response time T_r, and the throughput rate R that correspond to the minimum value of the rationality threshold $(F_o)_{min}$, were computed. The value used for $(F_o)_{min}$ is 5 bits/sec (Miller, 1956). Eleven mixed decision strategies were implemented for each MP transition (selecting one of the two COA development algorithms).

$$p_1 = 0.1\,k \qquad\qquad k = 0, 1, 2, ..., 10 \qquad\qquad (38)$$

Consequently, the number of behavioral strategies is 121. To each behavioral strategy corresponds a set of MOP values. The ranges of the MOPs are shown in Figures 21, 22, and 23.

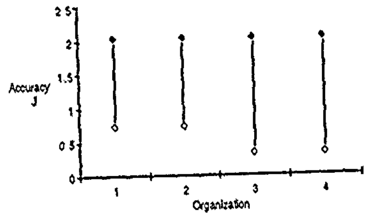


Figure 21. Range of Accuracy

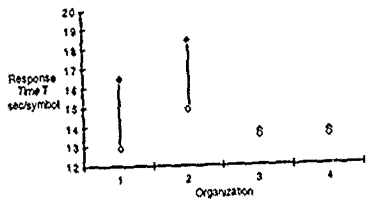


Figure 22. Range of Response Time

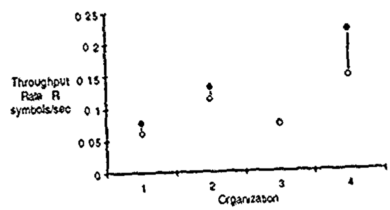


Figure 23. Range of Throughput Rate

Organization 2 (4 DMs) has greater response time than organization 1 (2 DMs) due to the time-consuming communication between the decisionmakers of each E2C. It has also greater throughput rate due to the function allocation to two decisionmakers, which creates directed elementary circuits with smaller processing times.

Organizations 3 and 4 have the same response time for corresponding behavioral strategies, because the information flow path that is dominant (critical circuit) does not contain the communication process between the two decisionmakers of the E2C. Organization 4 (5 DMs) has greater throughput rate than organization 3 (3 DMs), due to the allocation of functions to two decisionmakers per E2C.

The measure of Effectiveness Q has been parameterized by the requirements on accuracy, J_o, response time, T_o, and throughput rate, R_o. The computed accuracy of all organizations is comparable. Thus, a qualitative comparison can be performed by studying the sensitivity of Q to the requirements on response time, T_o, and throughput rate, R_o (Figures 24, 25, 26, and 27). These plots show the value of the Measure of Effectiveness for all combinations of requirements on the response time and the throughput rate, when the requirement on the accuracy is fixed at $J_o = 2.04$.

As the requirement on response time becomes more stringent, organization 2 has fewer behavioral strategies that satisfy the requirement than organization 1, i.e., lower MOE value. Conversely, when the requirement on throughput rate is increased, Organization 2 has more behavioral strategies satisfying the requirement than organization 1, and thus higher MOE value. When the requirement on throughput rate increases, Organization 4 has higher MOE value than organization 3. Finally, it is observed that organization 4 has a higher MOE value for all sets of requirement values than the other organizations; thus organization 4 is the best overall design.

The synthesis methodology, when applied to a problem such as this one in which it is necessary to process large amounts of information and arrive at accurate decisions in a timely manner, has yielded a number of data flow structures. From these structures, alternative organization architectures were obtained. The evaluation of the organizations in terms of their effectiveness has led to the desired result - a preferred organizational design.

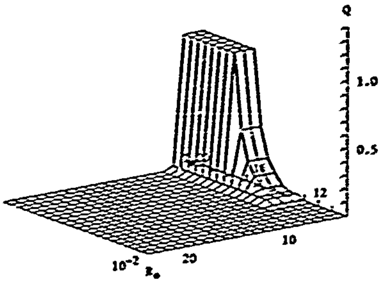


Figure 24. MOE Q of organization 1 vs requirements T_o and R_o

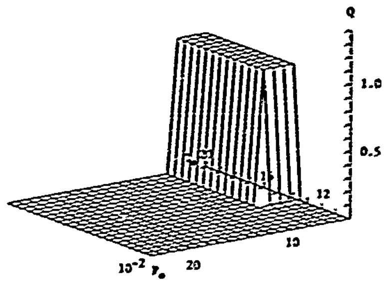


Figure 26. MOE Q of organization 3 vs requirements T_o and R_o

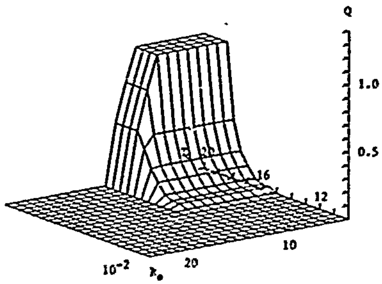


Figure 25. MOE Q of organization 2 vs requirements T_o and R_o

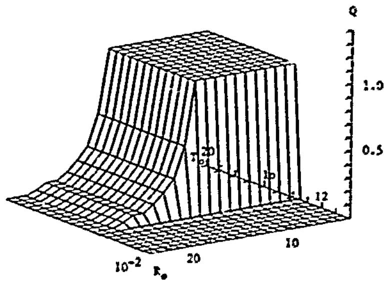


Figure 27. MOE Q of organization 4 vs requirements T_o and R_o

6. CONCLUSIONS

Four Command and Control organizations have been developed for the outer air battle, using a structured synthesis methodology. The synthesis methodology tackles the design problem at two levels: the data flow structure level and the organization architecture level. This decoupling enables the formulation of an algorithm that generates data flow structures parameterized by the complexity and redundancy of the information processing, without consideration of the organizational constraints. The organization architectures are developed from the candidate data flow structures, through the allocation of functions to decision-makers, and the selection of the supporting software and hardware.

The quantitative analysis computes the MOPs and MOE, taking into consideration both the variance of the human decision-makers maximum information processing rate, and differences in decision-making style. The organizations are compared on the basis of their MOE which is a measure of robustness of the design to the strategies implemented by individual decision-makers instantiating the organization.

The qualitative analysis of the sensitivity of the MOE to the requirements on the MOPs, allows for the selection of the best design, i.e. the organization that satisfies the requirements and is more robust to the decision-making styles of the organization members.

The methodology is a flexible top-down approach to the design problem, that results in the expansion of the set of candidate architectures. A potential benefit from the top-down approach is that the requirements for decision aids, databases, and communication links may be derived through the objective evaluation of the effectiveness of the C^2 organization.

Finally, the decoupling of the organization architecture design from the data flow structure design introduces two opportunities for the fine-tuning of the C^2 organization: one at the data flow level and one at the decisionmaker and system level.

7. REFERENCES

Alaiwan, H. and J.M. Toudic (1985). Recherche des Semi-Flots, des Verrous et des Trappes dans les Reseaux de Petri. *Technique et Science Informatiques*. Vol. 4, No. 1, Dunod, France.

Andreadakis S.K. (1988). *Analysis and Synthesis of Decision-making Organizations*. Ph.D. Thesis, LIDS-TH-1740, Laboratory for Information and Decision Systems, Cambridge, Massachusetts.

Boettcher, K. L., and A. H. Levis (1982). Modeling the Interacting Decisionmaker with Bounded Rationality. *IEEE Trans. on Systems, Man, and Cybernetics*, SMC-12, No3, 334-344.

Conant, R. C. (1976). Laws of Information which Govern Systems. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-6, No 4, 240-255.

Drenick, R. F. (1986). *A Mathematical Organization Theory*. North - Holland, New York.

Fernandez C. and P.S. Thiagarajan (1984). A Lattice Theoretic View of K-Density. *Advances in Petri Nets, Lecture Notes in Computer Science*, No 84, Springer Verlag, Hamburg, FGR.

Jin, V. Y., A. H. Levis and P. Remy (1986). Delays in Acyclical Distributed Decision-making Organizations. *Proc. 4th IFAC Symposium on Large Scale Systems: Theory and Application*, Zurich, Switzerland.

Levis, A. H. (1984). Information Processing and Decision-making Organizations: A Mathematical Description. *Large Scale Systems*, Vol. 7, 151-163.

Levis, A. H., and K.L. Boettcher (1983). Modeling and Analysis of Teams of Interacting Decisionmakers with Bounded Rationality. *Automatica*, Vol. 9, No. 6, 703-709.

March, J. G. (1978). Bounded Rationality, Ambiguity, and the Engineering of Choice. *Bell Journal of Economics*, Vol. 9, 587-608.

Martinez, J. and M. Silva (1980). A Simple and Fast Algorithm to Obtain all Invariants of a Generalized Petri Net. In Girault and Reisig, (Eds), *Application and Theory of Petri Nets*. Springer Verlag, Berlin, FRG.

Miller, G.A. (1956). The magical Number Seven, Plus or Minus Two. Some Limits on our Capacity for Processing Information. *Psychological Review*, Vol. 63.

Owen, G. (1968). *Game Theory*, Saunders, Philadelphia, Pensylvania.

Peterson, J. L. (1981). *Petri Net Theory and the Modeling of Systems*. Prentice Hall, Englewood Cliffs, New Jersey.

Raisbeck, G. (1963). Information Theory: *An Introduction for Scientists and Engineers*. MIT Press, Cambridge, Massachusetts.

Ramchandani, C. (1973). *Analysis of Asynchronous Concurrent Systems by Timed Petri Nets*. Technical Report No. 120. Ph.D. Thesis, Laboratory for Computer Science, MIT, Cambridge, Massachusetts.

Reisig, W. (1982). *Petri Nets: An Introduction*. Springer Verlag, New York.

Shannon, C. E., and W. Weaver (1963). *The Mathematical Theory of Communication*. University of Illinois Press, Urbana, Illinois.

Tabak, D., and A.H. Levis (1986). Petri Net Representation of Decision Models. *IEEE Transactions on Systems, Man, and Cybernetics*, Vol SMC-15, No 6, 812-818.

JOINT INTEROPERABILITY EVALUATION WITH JDLNET

William J. Dejka
Naval Ocean Systems Center
Research Evaluation and Systems Analysis
San Diego, California 92152

Naval Ocean Systems Center
San Diego, California 92152

1.0 INTRODUCTION/PURPOSE

The recent interest in distributed simulation as typified by the JDLNET, supported by the Joint Director of Laboratories (JDL) Technical Panel on Command Control and Communications (TPCCC), indicated that new capabilities for evaluation are feasible but actual application of these capabilities are not well understood and difficult to plan. The uses of distributed simulation include technology projection, formulation of architectural concepts, and decision aid evaluation. One area which is now recognized as practical is the evaluation of joint service interoperability and JDLNET is a vehicle that can be used for this evaluation. This paper will describe this approach which has been demonstrated at the Naval Ocean Systems Center. The experiments conducted show feasibility of the approach.

JDLNET will provide a simulation capability by interconnecting RADC, CECOM, NOSC, and NPGS using the warfare environment simulator known as RESA. RESA will be described as well as the JDLNET in the context of interoperability analysis. The methodology used to analyze interoperability will also be presented

2.0 INTEROPERABILITY ELEMENTS.

The definition of interoperability as given by JCS Pub 1 is "the ability of systems units, or forces to provide services to and accept services from other systems, units, or forces and to use the services exchanged to operate effectively together". The definition implies that the evaluation involves actual systems, communications, warfare environment, and other elements. These elements of interoperability are listed below and will be discussed.

a. ENVIRONMENT. This element includes threat assets, mission ROE, area of operations and those factors generally out of the control of the commanders but important in that they contain the warfare environment that limits the scope of interoperability evaluation.

b. REQUIREMENTS. There are restrictions placed on the commands/nodes that are in place mainly as a result of time and these elements include shore and battle force connectivities, surveillance data collection methods, and other command and control capabilities. These requirements are complex reflecting the broad nature of warfare.

c. SYSTEMS. The command/nodes in the centers consist of systems which perform various functions including correlation, track management, force control, and message handling. These systems are a principal part of interoperability, but not all the functions of the system may be involved. Some of the systems are the Naval Tactical Data System (NTDS), the USA's TSQ-73, and the Air Force MCE.

d. MEDIA (COMMUNICATIONS). Nearly all of the forms of communication such as data links, TTY circuits, and voice circuits would have an impact on the ability of a command or node to provide services.

e. LANGUAGE. The standard message formats such as Tadil-A, Tadil-B, Tadil-J, JINTACCS-MTF are critical to the effectiveness of the functional operations. These formatted messages must be understood at all the commands or nodes to operate effectively.

f. OPERATIONAL PROCEDURES. The interface operational procedures (IOP) involve the management of data, air space control, search and rescue procedures, weapons engagement and status, and other function procedures which can only be completed with the human in interoperability and they describe the role of the human in the operational process.

The above discussion introduces the notion of interoperability and many interoperability issues are well known. Issues include timely correlation of data, consistent tactical picture, effectiveness of planning, operational situation reporting, and others. Once the issue is identified, the elements of interoperability are selected and an operational configuration can be selected. Some typical configurations will be described in a later section.

3.0 BACKGROUND

3.1 RESEARCH, EVALUATION AND SYSTEMS ANALYSIS (RESA).

The warfare environment simulator called RESA is a VAX-based force-level simulator which supports command, control analysis and is the simulator used in JDLNET. The RESA capability was designed for fully interactive gaming between forces and allows man-in-the-loop operations. There is also a capability to script and conduct exercises without the human participation. This scripting permits the system to be run many times which allows a Monte Carlo analysis and opens the way to reduced manning operation. Lastly the simulator will also generate actual message traffic in standard operational formats as part of the real-time scenario. The latter capability is utilized in the joint interoperability evaluation concept presented here. Other basic features of the system which are of interest are summarized in the following:

* Real-time interactive system with one minute step size-track motion, sensors, weapons updated each minute.

* Platforms (ships, subs, aircraft) are individually controlled or can be collectively controlled (e.g., battle group).

* Models are table-driven which allows hypothetical objects to be simulated (future scenarios are possible).

* Forces are assigned to views which are controlled by the simulator.

* Each emulated command center receives sensor data and messages which are computer generated.

* Control/umpire view has complete force information.

The models used in RESA are summarized in Table 1. Most are self-explanatory but a discussion of a few will give an insight into the models. One model that demonstrates the built in logic is the flight operations model. Aircraft are launched and will automatically rendezvous and, once all aircraft have arrived they will proceed with their mission. In addition, platforms will automatically engage a threat once the command for engagement has been given. The simulator will represent engagements without a human in the loop

Other sensor models are designed to provide force level data. The radar model uses the radar equation which is then converted to a probability of detection. The sonar models use the sonar equations. The navigation model uses error mean and variances of the navigation system to introduce positional inaccuracies.

TABLE NO. 1 RESA MODELS

PLATFORM MOTION - AIRCRAFT, SHIPS, SUBS

SUB BASES

SATELLITE MOTION MODEL

FLIGHT MODEL - CARRIER LAUNCH MODEL
REFUELING MODEL

LOGISTICS - FUEL, WEAPONS

SENSORS - RADAR
ESM
JAMMING
SONAR, ACTIVE
SONAR, PASSIVE

NON-ORGANIC SENSORS
HFDF
SONABUOYS

AUTOMATIC TRACK CORRELATION

COMMUNICATION NET MODEL

CHARACTER-ORIENTED MESSAGE GENERATION

BIT-ORIENTED MESSAGE GENERATION

ENGAGEMENT MODELS

DAMAGE MODEL

Other model data can be obtained from the references.

Another feature of the simulator is its ability to operate in a geographically distributed environment. The VAX-based system will allow several VAXs to be netted together to participate in the same exercise. The host machine (master) will process all commands, platform motion, sensors, and weapons, Then the communication software distributes the dynamic data base to all VAXs on the net where command centers can be supported. This capability to distribute the data base is necessary and fundamental to the operation of JDLNET.

3.2 JDLNET.

The purpose of JDLNET is to provide a distributed simulation capability to support command and control research and development goals. The concept evolved when it was recognized that RESA could operate distributed and no costly software development was neeeded to complete the simulation network. A network of sufficient bandwidth was required to interconnect the principal laboratories (Figure 1). Though there were several options for the JDLNET, the DDN-DISNET was selected to demonstrate the concept. DISNET is a GENSER secret network similar to ARPANET. The network along with RESA will allow a shared real-time exercise to be conducted among the JDL laboratories. The commands are sent to the master host and then the host distributes the dynamic data that changed. The data is distributed to the remote hosts. Each site will have as many work stations as are necessary to support the exercise objectives. The simplest use of JDLNET is the conduct of joint exercises. A more complex use of the network is interoperability evaluation.

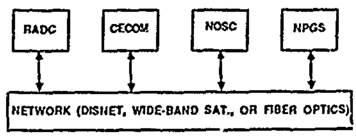


FIGURE NO. 1 JDLNET

4.0 JOINT INTEROPERABILITY EVALUATION.

The air defense operational concept is represented in Figure 2. This concept is a typical joint operational configuration. There is a Joint Operations Commander (JOC) and an Area Air Defense Coordiantor (AADC) who have joint mission responsibilities. Each service has a command responsiblity. For the Navy that is a battle group commander. Navy assets include the guided missile ships and aircraft (E-2C, F-14). Air Force command responsibility rests with the Regional Air Defense Coordinator and their assets include AWACs and F-15s. The Army command in air defense is the Patriot/IHAWK Brigade. The Patriot and IHAWK firing units are his to control. This operational concept is represented with JDLNET in Figure 3. Each service laboratory is developed to represent the command centers and the inter connections include the data communications and simulation data. Voice communications would require separate circuits in the current configuration of JDLNET but later when more bandwidth is available, it would be included in the net. Each command center and/or systems in the command center would be stimulated by JDLNET's simulator (RESA). The simulator's ability to remotely generate both bit oriented messages (BOM) and character oriented messages actually is the stimulus at each node.

The evaluation methodology is presented in the following steps:

a. Define the objective (based on issues)
b. Analyze operational conept/functions
c. Select evaluation configuration/implement configuration
d. Develop scenario/files
e. Develop data collection plan/implement
f. Conduct exercise
g. Perform analysis

The approach is expandable to address any interoperability issues.

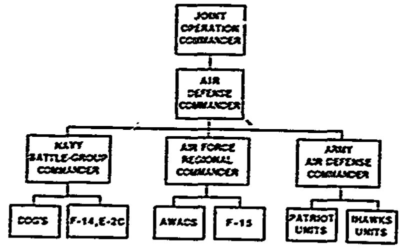


FIGURE NO. 2 AIR DEFENSE OPERATIONAL CONCEPT

5.0 SUMMARY.

The JDLNET and the RESA simulator is an effective capability to conduct interoperability issue evaluations. The current system is adequate for some applications, but plans to upgrade to either wide band satellite or long haul fiber optics will significantly expand its capabilities. In addition the current simulator, RESA, can also grow into a significant system through the use of parallel processors and architecture re-design while maintaining the existing software. As the number of interoperability experiments grow, the capabilities of JDLNET will be recognized and the concept will be more widely accepted.

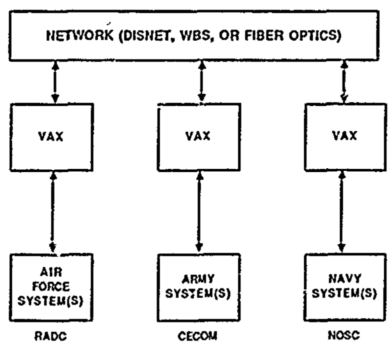


FIGURE NO. 3 JOINT INTEROPERABILITY CONFIGURATION

Project Juniper
Cooperative Joint Mission Planning
by

Dr. Ronald W. Larsen

Naval Ocean Systems Center
Artificial Intelligence Technology Branch, Code 444
San Diego, CA 92152-5000

ABSTRACT

Project Juniper addresses the technology of distributed expert decision aids in the context of cooperative and supportive joint Navy/Air Force air strike mission planning. The Navy and Air Force have developed expert systems, the Air Strike Planning Advisor (ASPA) and the Knowledge-based Replanning System (KRS), respectively, which support the planning of strikes against land targets. They have been networked to demonstrate the feasibility of distributed planning for joint Navy/Air Force air strike missions.

INTRODUCTION

Project Juniper was initiated in FY-85 under the JDL C^3 technology program. The overriding Joint Service need in C^3 is rapid, reliable and effective exchange of timely information for planning, decisions and command action. Project Juniper addresses the technology of distributed expert decision aids in the context of cooperative and supportive joint Navy/Air Force air strike mission planning. Air strike mission planning is currently a time consuming and man intensive process. To respond to rapidly changing battle environments, the time to accomplish all levels of mission planning must be significantly reduced. An increasingly lethal threat environment requires that planning effectiveness must also be improved.

The Navy and Air Force have developed expert systems, the Air Strike Planning Advisor (ASPA) and the Knowledge-based Replanning System (KRS), respectively, which support the planning of strikes against land targets. Each of these developments utilize different hardware and software. They have been networked to demonstrate the feasibility of distributed planning for joint Navy/Air Force air strike missions. That is, the system users work together, sharing data and results, to plan a joint mission.

BACKGROUND

The Navy and the Air Force have separately been working on expert system projects designed for the air strike mission. These computer-based systems are decision aids which provide the assistance one might expect from an expert air strike planner. That is, as the air strike mission is being planned, the system will interpret the data that is entered and make inferences on that data based on rules within the system. For example, if the plan requires that a certain aircraft with a specific weapons load fly from point A to point B at a specified altitude and speed, the expert system can be used to develop a route plan, an itinerary and predict fuel usage. If the system also knows where enemy SAM sites are located, it can recommend countermeasures.

Because the Navy and Air Force may be called on to support each other in an air strike mission, the abilities of the expert systems developed by the services must be mutually supportive. That is, the two systems should be able to work together, sharing data and results, to plan the joint mission. Project Juniper provides a method for testing the two systems ability to work together to support a joint Navy/Air Force air strike mission.

APPROACH

This joint technology demonstration draws upon current on-going efforts in expert systems technology in the Navy and Air Force. Project Juniper involves the integration of five separate technologies:

Expert systems: Air Strike Planning Advisor (ASPA) and Knowledge-based Replanning System (KRS).

User Interface: Map Object Oriented System Environment (MOOSE) and natural language and mixed initiative interfaces.

Networking: Local area and long haul networking of distributed planning systems.

Simulation: The Research, Evaluation and Systems Analysis (RESA) warfare simulator provides a validated event driver for testing the expert systems technology.

Decision-making: Protocols have been developed which support, via network communications, a real-time distributed planning system for joint Navy/Air Force air strike missions.

A fuzzy logic decision tool has been integrated into Juniper to support plan evaluation and selection.

JOINT SERVICE COMMAND STRUCTURE

The Joint Service Command Structure for combined Navy and Air Force operations provides the operational contect for Project Juniper. The existence of the Joint Force Commander ensures that the individual services involved in joint operations have a common goal and understand their responsibility in achieving it. However, the planning and execution of that responsibility normally requires close coordination between the component commands.

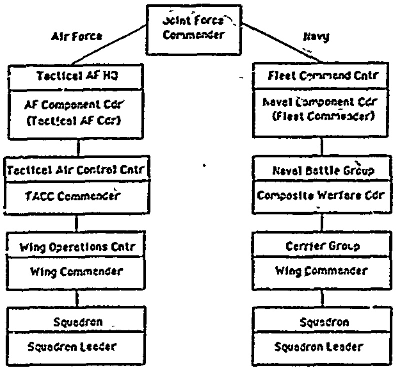


Fig. 1 Joint Operations Command Structure

In the case of air strike operations, the Navy and the Air Force command structures parallel each other in a manner as shown in Figure 1. In particular, the planning of Air Force strike operations is distributed between the Tactical Air Control Center which develops high level plans for the theater of operation, and the Wing Operations Center which does detailed planning of particular missions. In a similar respect, Navy planning is distributed between the Composite Warfare Commander, which develops operations plans for an entire battlegroup operation, and the Wing and Squadron Commanders and their staffs which plan specific strike missions.

Figure 2 shows the general relationship between mission planning and the command hierarchy. At the highest levels, the joint force and theater commands, emphasis is on advanced planning, characterized by course granularity and stored plan options. High level planning primary deals with warfare objectives, resource requirements and long term strategy.

At the intermediate planning level, the tactical commanders generate daily schedules, allocate resources, compare options and replan in near real time. At the lowest level, the strike leader must develop fine grain executable plans in real time. These plans must be responsive to a dynamic environment in terms of weather, resource availability and threat conditions.

Command	Product	Concerns
Joint Force Commander	Tasking Directive	Objectives Priorities Joint Operations Intelligence Fusion
Theater Commander	Operations Order	Timing Sensitivity Joint Suppression Weather
Tactical Commander	Air Task Order	Asset Allocation Scheduling Refueling
Wing Commander	Air Task Message	Ordnance Selection Delivery Tactics Countermeasures
Strike Leader	Strike Plan	Survivability Route Planning Formation Tactics

Figure 2 Mission Planning vs Command Hierarchy

JUNIPER CONCEPT

The Navy and Air Force have developed expert systems, the Air Strike Planning Advisor (ASPA) and the Knowledge-based Replanning System (KRS), respectively, which support the planning of strikes against land targets. These developments utilize different hardware and software, but they have been networked to demonstrate the feasibility of distributed planning for joint Navy/Air Force air strike missions. That is, the system users work together, sharing data and results, to plan a joint mission.

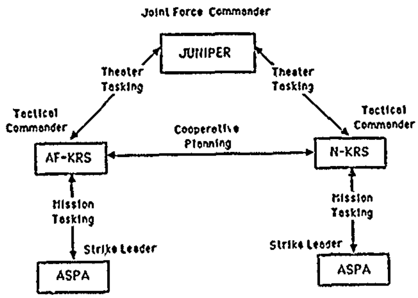


Figure 3 Joint Operations Command Structure

Figure 3 shows how the Juniper system supports distributed planning in a joint operations command structure. The top level Juniper system supports the generation of theater tasking. The second level consists of an Air Force and Navy version of KRS which support cooperative planning. At the lowest level the ASPA system supports the generation of plans for a specific mission.

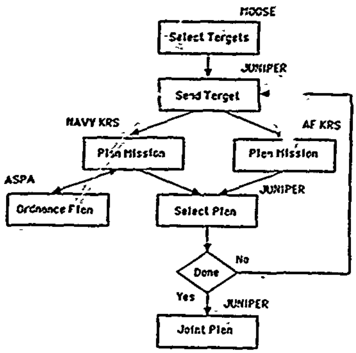


Figure 4 Cooperative Planning Cycle

Figure 4 shows a typical cooperative planning cycle using the Juniper system. The first step, target selection, is accomplished with the aid of the Map Object Oriented System Environment (MOOSE). MOOSE is a flexible, color graphic oriented, map and information display system that uses World Data Bank I and II. Object oriented techniques allow the user considerable flexibility in building a customized collection of entities with associated characteristics and relationships. MOOSE is used in Juniper to graphically identify, review and select targeting information, position carriers and modify strike aircraft routes and refueling stations.

The next step in the Juniper planning cycle is to generate and send joint tasking directives or target lists to the respective Navy and Air Force KRS planning systems. KRS deals primarily with the allocation and scheduling of resources for manned aircraft strike missions and the associated support missions such as air escort, SAM suppression, ECM and refueling. It also plans cruise missile missions. KRS can be used interactively by the user for data base query, plan verification or plan generation at the theater level. The Navy and Air Force version of KRS work together to plan joint missions in three different theaters of operation, Europe, Southeast Asia and the Mediterranean. When standard plan knowledge cannot be used at the theater level, the Navy KRS can task ASPA to automatically develop sub-plans tailored to new targets, a changing threat environment or changing weather conditions.

ASPA functionality currently includes weaponeering and defense suppression planning. The weaponeering module integrates existing stand-alone decision aids, large data files, and expert knowledge to support the selection of aircraft, weapons loads, load configuration and delivery tactics needed to achieve desired strike mission effectiveness. The defense suppression module is designed to support a strike leader in the selection of resources and tactics to suppress the local area defensive threat in support of a land air strike. The defensive threat includes early warning radars, surface-to-air missiles, anti-aircraft artillery and air intercept. The resources and tactics to be considered include the electronic countermeasures, decoys, anti-radiation missiles and deception employed by Naval aircraft.

When strike plans have been developed by the individual services, they are sent back to the top Juniper level where plans are selected and integrated by a process of machine assisted review and comparison using a fuzzy logic decision tool. The user may select his own plan evaluation criteria and their relative importance. The joint plan is determined as the best combination of Navy/Air Force assets needed to accomplish the primary and support missions, considering primary mission parameters such as (1) availability of assets like aircraft, weapons and fuel, (2) number of assets required including refueling and other support assets, (3) mission duration and time over target, and (4) likelihood of mission success.

AN EXAMPLE

Although the Juniper system can be used to plan attacks on any portion of the globe, several detailed scenarios are available for test and evaluation. The most recently developed scenario is a recreation of the April 15, 1986 Libyan air strike. This military action is particularly pertinent to Project Juniper because the Joint Chiefs of Staff (JCS) participated in the planning and it was executed using both Air Force and Navy resources. The Libyan raid required air power projection, refueling, electronic counter-measures (ECM), SAM suppression and air escort missions. Juniper provides the capability to quickly plan and re-plan complicated air strikes of this nature. The remainder of this section will detail a typical Juniper planning cycle, utilizing key Libyan targets as an example.

The Libyan attack scenario, as re-created on the Juniper system, demonstrates Juniper's flexible capability to plan and re-plan single service or joint service air strikes and the accompanying supporting missions. These tools would have been helpful to the JCS staff in rapidly exploring and testing contingency plans. The specific details of the mission and target area defenses have been altered or omitted to keep the example unclassified.

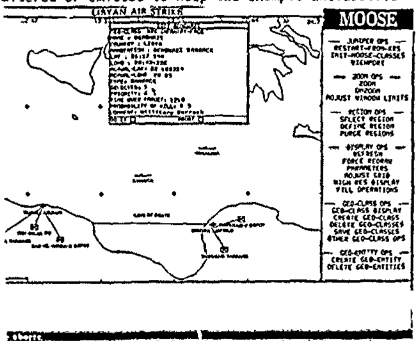


Figure 5 Target Selection - Benghazi Barracks

By clicking on the installation labeled "Benghazi" Barracks", the information window in Figure 5 is shown. This window contains the major properties of the targets knowledge base for this installation. The user may change any of these properties by clicking on the property name and entering a new value. To select this installation as a target, the operator changes the SELECTED property from F for false to T for true. This installation is then put on the target list. The user may also modify properties which will also be included in the target list, such as priority, time over target (TOT), probability of kill (PK) and a comment.

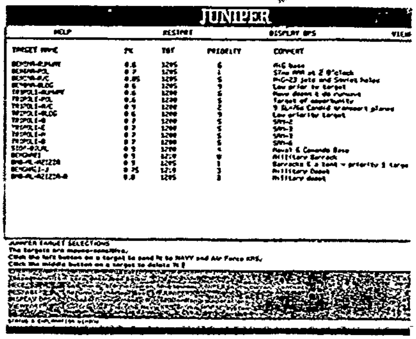


Figure 6 Target Selection - Remaining Targets

The target selection procedure is repeated until all desired targets are on the targets list. This list can then be displayed on the Juniper interface by selecting the LIST TARGETS function under the Display OPS menu (See Figure 6).

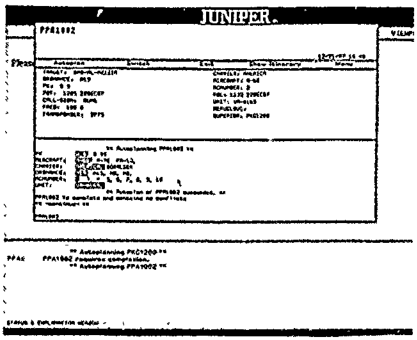


Figure 7 N-KRS System Planning Bab-al-azizia Target

Figure 7 shows the Navy mission planning window. The data outlined in black are legal values for certain aspects of the plan. These values get filled into the detailed plan summary in the window above. The "PPA1002" designation in the upper left corner identifies this as a power projection mission.

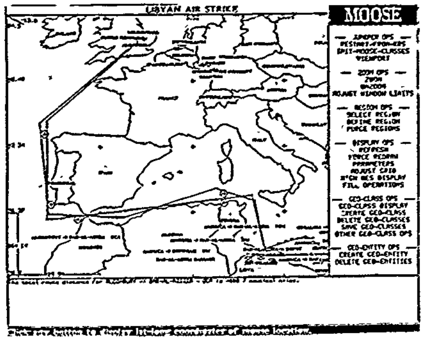


Figure 8 MOOSE - Estimate Refueling Orbits

The system can now estimate refueling locations for the route (See Figure 8). The refueling icon is the arrow inside of the ellipse. The ellipse represents the orbit of the tanker and the arrow indicates the direction the strike force is moving. In this case the length of the route is 4880 nautical miles and requires 5 refuelings.

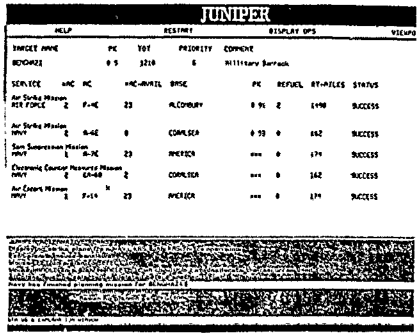


Figure 9 Juniper System - Summary of Candidate Missions

Both services have returned all of their candidate missions for the Benghazi military barracks. A summary of the plans is displayed on the Juniper interface (See Figure 9). It is now up to the user to select among these candidate missions based on some theater level strategy or other outside intelligence. The fuzzy logic tool can also be used to help the user make a decision.

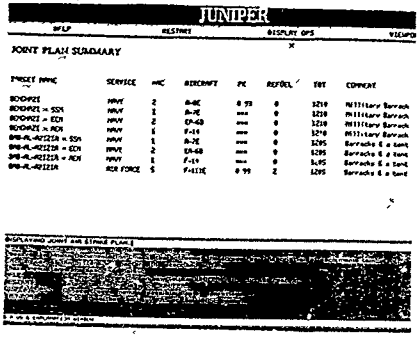


Figure 10 Juniper System - Joint Plan Summary

The selected Benghazi and Bab-al-azizia missions are now added to the Joint Plan Summary and displayed for operator approval (See Figure 10). This, as well as most of the other stages of planning, can be printed out in Navy hardcopy format.

MODEL-BASED HUMAN INTERFACE DESIGN FOR DISTRIBUTED TACTICAL COMMAND AND CONTROL*

Donna L. Saisi Daniel Serfaty

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, Massachusetts 01803

ABSTRACT

An adaptive, model-based interface provides a way to enhance the quality of team decisionmaking. An operator function model of human decisions was used as a tool to guide interface development for a tactical decisionmaking environment. The resulting model-based interface self-adapts to the information needs of the team members which vary with the system state and the operator activities. Both the information content of displays and the command structure of the interface support the human operators. The interface was designed to decrease the amount of low-level perceptual and manual tasks, thereby freeing team members to concentrate on advanced decisionmaking problems. A preliminary evaluation of the model-based interface showed that teams using this interface perceived a lower workload, yet were able to achieve a significantly higher level of system performance.

INTRODUCTION

Modern computer technology provides the human-system interface designer with almost unlimited options for both the content and format of displays. When digital computers were first introduced, their displays tended to reflect precomputer practices (Mitchell and Miller, 1986). Before the advent of computers, when systems were manipulated directly, multiple sensors, each tracking a separate low-level process, reported their current status to the operator This so-called single-sensor, single-indicator layout was directly transferred to computer displays, so that the system representation, as viewed by the user, consisted of multiple status indicators, a subset of which updated whenever a system event occurred. Given the information processing capabilities of both humans and computers, this proved to be a poor interface design approach. Searching for and integrating unprocessed data is a time consuming responsibility for a human operator. Time that could be spent in high level decisionmaking is spent doing tedious data aggregation tasks - tasks better suited for a digital computer than a human decisionmaker.

The importance of interfaces which reflect the operator's information needs is now widely recognized. Smith and Mosier's (1986) user interface design guidelines state that we should,

> "Tailor displayed data to user needs, providing only necessary and immediately useable data for any transactions; do not overload displays with extraneous data." p. 98

Although the goal is recognized, little has been written on methods for achieving it. As noted by Mitchell and Miller (1986) the current focus of human factors literature is on enhancing the format of information, rather than on determining how to generate the information content of displays. The latter issue is addressed here.

THE OPERATOR FUNCTION MODEL

A modeling approach has been developed from which adaptive interfaces can be created. The interface is designed to adapt to the human operator in the following respect. As the operator task and the system state evolve over time, the displays vary in information content, the set of available commands changes, and the procedures to accomplish commands alter.

Human functioning in complex systems has been successfully represented by "discrete control models" (Miller, 1985) Systems of interest are those in which alternatives available to the human are finite and sequential decisions are made Precedent decisions affect the system and, therefore, influence future decisions. Discrete control models create a mathematical representation of the decomposition of a complex system, together with a description of how operator activities are related and coordinated.

The "operator function model" (Mitchell and Miller, 1986) is built upon discrete control modeling constructs, but redefines the network elements so that the model may serve as a basis for user interface design (Mitchell and Saisi, 1987) Like the discrete control model, the operator function model is a heterarchical / hierarchical network of arcs and nodes with the links between nodes expressing how system elements are related. However, the lowest-level system components (i.e., system outputs) are redefined as operator control actions (either cognitive or manual) and information needs. Cognitive action nodes are linked to the information a person requires to carry out the mental activity. These modeling enhancements allow the human's dynamic focus of attention and evolving information requirements to be explicitly included in the network, even in cases where the person engages in cognitive

* This work was sponsored by the Office of Naval Research under contract N00014-84-C-0577.

activities, such as system monitoring or situation assessment. Modeling cognitive activity is not as straightforward as modeling manual actions, because the former are not directly observable. During system analysis, cognitive activity may be inferred from information requests and may be understood through questioning system operators.

THE DDD EXPERIMENTAL PARADIGM

The operator function model was applied to determine the system commands, the command procedures, and the display content for the Distributed Dynamic Decisionmaking (DDD) experimental paradigm. The DDD paradigm is an simplified representation of a tactical command and control environment and was developed to examine how a team of decisionmakers solves problems of situation assessment and resource management under uncertainty. The paradigm involves the allocation of heterogeneous resources to prosecute different types of threats in a dynamic, uncertain, multi-object environment (Kleinman, Serfaty, and Luh, 1984; Serfaty and Kleinman, 1985). The following is a sample of the functional characteristics supported by the paradigm.

OBJECT TYPE

Incoming objects are assigned an identity. There are two types of objects that are threats and should be destroyed before they penetrate; these can be thought of as air and surface threats. There is a friendly or neutral type that should not be prosecuted. Some objects are assigned an unknown type and require identification. The identity can be found either through probing the environment or receiving information from one's partner, or can be estimated from object attributes.

FINITE OPPORTUNITY WINDOW

The DDD paradigm supports time-restricted decisionmaking. Each incoming object is assigned a finite time available (a random variable), and decisionmakers are able to process objects only within that time span.

TARGET ATTRIBUTES

Each object is characterized by an attribute set $\underline{a}$. The first attribute component can be thought of as strength, the second as evasiveness. The a_i are random variables selected from a distribution whose range depends on object type. The $\underline{a}$ ranges for different object types partially overlap.

RESOURCES

There is a fixed amount of renewable resources $\underline{r}$ available to the team. Resource components can be thought of as aircraft, ships, and radars. At any time each team member owns (i.e., controls) some of each resource. Resources may be transferred between team members, but the total amount of resources for the team remains constant. Resources are used for two purposes - information gathering from the environment and threat prosecution.

For each type of threat, a mapping function transforms attributes of the threat into a set of resources required for prosecution. The decisionmakers are given a decision support system (DSS) to accomplish the mapping at a level of accuracy pre-specified by the system to reflect the decisionmaker's level of expertise.

COMMUNICATION

Team members do not necessarily have access to the same object database. Depending on the experimental condition, a target may be visible to one team member, but not the other. Both team members could view an object, but its identity could be given to one person and unknown to the other. To share information on the presence, type, or attributes of an object team members must communicate. In addition to sharing information, they can support one another by transferring resources and can also communicate actions they intend to take.

THE DDD OPERATOR FUNCTION MODEL

Since there are multiple decisionmakers there should be multiple operator function models to characterize each team member's responsibilities. However, in the present study since we focus on a two-person, parallel team (a dyad) where team members have the same type of responsibilities, one model was sufficient. The following description of the DDD operator function model is given from the perspective of one team member, "the decisionmaker", who supports and is supported by the other team member, "the partner".

There are five major DDD operator functions (Figure 1). The model indicates that situation assessment is the central function. Situation Assessment refers to both external (object) and internal (resource) states. The decisionmaker assesses the situation and makes one of three decisions. The first leads to target processing, the second leads to resource redistribution, and the third results in monitoring for a relevant event to occur. When any of these three functions is completed, the decisionmaker returns to situation assessment. The situation assessment function can be interrupted by a request from the decisionmaker's partner. Once the request is acknowledged, the decisionmaker resumes assessing the situation.

Each major operator function is decomposed into subfunctions, tasks, subtasks, etc. To transition between these activity nodes, a system event must occur or a system condition must hold. Transition arcs are labeled with the appropriate system event / condition. Information nodes are connected to the lowest level activity nodes with information arcs. Approximately 150 activity and information nodes were used decompose the five major operator

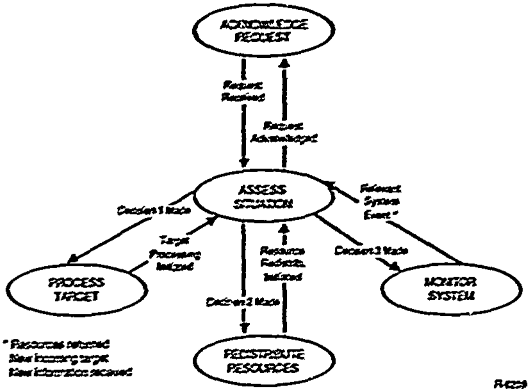


Figure 1. Major DDD Operator Functions.

functions. The detailed decomposition is beyond the scope of this paper, but can be found in Saisi et al. (1988).

THE OPERATOR FUNCTION MODEL BASED INTERFACE

Figure 2 shows an overview of an interface designed based upon a DDD operator function model (OFM). The top half of the screen is dedicated to providing object information required for situation assessment and for system monitoring. When the decisionmaker determines that no action is currently appropriate, he moves into a less cognitively demanding system monitoring function in which he waits for the arrival of a new object, new information, or more resources. In addition to the object information, the amounts of current and returning resources are also required for situation assessment - once objects have been prioritized, it is necessary to determine the amount of team resources available to process top priority objects. The decisionmaker allocates a great amount of time to the situation assessment function, leaving it to process a target, redistribute resources, or respond to a request from his partner, and then returning. Therefore, there are no commands required to access information required for situation assessment, and this information is displayed continuously.

The command execution area is where the remaining major operator functions (i.e., process a target, redistribute resources, and acknowledge request) are accomplished. The command buttons directly correspond to the subfunctions of these three top level functions. The information required to accomplish these subfunctions are displayed as needed in both the attribute information area and the command execution area. The commands to accomplish subfunctions adapt to the system state, for example, if a procedure will not yield new information, it is not presented as an option.

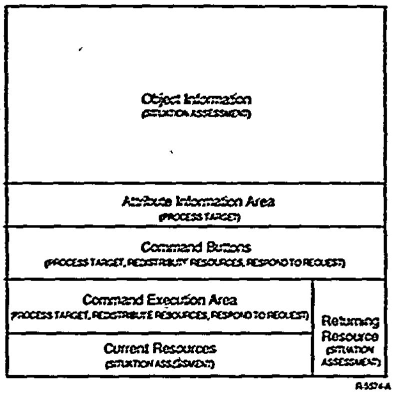


Figure 2. An Overview of the OFM-Based Interface.

EVALUATION OF THE OPERATOR FUNCTION MODEL BASED INTERFACE

An experiment was conducted to evaluate the OFM-based interface whose design was guided by a model of the the DDD team members' functions. We were interested in testing the hypothesis that the operator function model can be used as a design tool to produce an interface that causes a reduction in workload and an improvement in team performance. To evaluate the operator function model interface design approach, the OFM-based interface was compared to a baseline interface developed using a typical design approach in which all information associated with system elements (in this case incoming objects) is displayed continuously at a very detailed level. For the DDD paradigm, the baseline I interface consisted of two terminals with keyboard entry. The top screen was a graphics display largely dedicated to indicating the current location of objects. Below it was an alphanumeric screen consisting of a row of data items for each object.

It would be difficult to test our experimental hypotheses simply by comparing the OFM-based and the baseline I interfaces, since there are many discrepancies between these two interfaces in addition to their underlying design approaches. One set of subjects was run in 1985 on the baseline I interface; another subject pool was used to test the OFM-based interface in 1987. The interfaces were implemented on different hardware. The baseline I interface was implemented on a PDP-11/60 computer, the OFM-based interface on a network of SUN-3/160C workstations. The SUN-3/160C has many features not available on the PDP-11/60, including more memory, multiprocessing capabilities, high resolution graphics, color, overlapping windows, and mouse input.

In an effort to separate the effects of external factors (i.e., subjects, time, hardware, and software) from the dependent variable of interest (i.e., the underlying interface design approach), two intermediate interfaces were developed and evaluated. The first of the new interfaces, the baseline II interface, replicated the baseline I interface on the SUN system. A few changes were made to simplify the programming of the baseline II displays, for example one terminal rather than two was used; instead of a radarscope, a linear display was implemented to show the location of incoming objects; and object data items were arranged in columns rather than rows. The second intermediate interface incorporated the SUN's advanced hardware and software features using human factors guidelines (Smith and Mosier, 1986) and, thus, is referred to as the "human factors" interface. The information content and the command structure of the human factors display is that of the baseline displays, however data format and layout are enhanced (e.g., using graphics and color) and a new command input device (i.e., mouse) is employed. The most salient differences between interfaces are summarized in Table 1.

GLOBAL HYPOTHESES

Moving from the baseline interfaces to the human factors interface and the OFM-based interface, a monotonic increase in team performance was expected to occur. The baseline I and baseline II interfaces are basically the same interfaces implemented on two different computer systems. The first hypothesis was that no differences would be found between the baseline I and baseline II interfaces.

The human factors interface incorporates modern hardware, a new input device (i.e., mouse), and software advancements including color coding, shape coding, and dynamic icons in accordance with human factors guidelines. The intent of the human factors interface design was to enhance the data format and simplify the manual requirements for command execution. The second hypothesis was that in comparing the baseline II and human factors interfaces, human factors enhancements would lead to improved timeliness measures and would cause workload to decrease.

The OFM-based interface incorporates the data enhancement techniques of the human factors interface, but these principles were applied after new data and command procedures had been generated. Display content and system commands were developed based upon a model of operator functions in the system. Higher level commands were defined that mapped directly to major functions and subfunctions; the steps required to accomplish commands were designed to adapt to the current state of the system. In addition to the information content of displays, when information appears and where it is placed were also guided by the DDD operator function model. These two features of the OFM-based interface were expected to lead to improved timeliness measures and improved performance along the accuracy dimension. It was also expected that coordination would improve, since the OFM-based interface was designed to keep track of what information has and has not been transferred, and the display's information content helps each decisionmaker anticipate support required by his partner. A significant decrease in workload was not expected. At first glance it appears that the features of the model-based interface would promote a workload decrement - commands better map to operator intent, data is aggregated, and information content is adaptive. However, the

TABLE 1. INTERFACE FEATURES

Features / Interface	Hardware	Subjects	Display Radial	Display Linear	Human Factors Features (Mouse, Color)	Formal Planning Capacity	Decision Support Automated	Systematic Task Analysis
BASELINE I	PDP-11/60	1985	X					
BASELINE II	SUN	1987		X				
HUMAN FACTORS	SUN	1987		X	X	X		
OFM-BASED	SUN	1987		X	X	X	X	X

R4813-B

DDD operator function model indicated that there was information required by the team that was not displayed by the baseline interfaces. The teams using the OFM-based interface had more relevant information available for use, which was expected to counteract the workload decrement. In summary, the third hypothesis was that in a comparison between the human factors and OFM-based interface, the OFM-based interface would lead to improved timeliness, accuracy, and coordination, without causing an increase in workload level.

EXPERIMENTAL DESIGN

SUBJECTS

Four two-member teams participated in the experiment. Subjects were graduate and undergraduate students from the University of Connecticut. Each team used all three interfaces, i.e., baseline II, human factors, and OFM-based. As mentioned previously, performance data for the baseline I condition was obtained in an earlier experiment.

PROCEDURE

Subjects received an average of 15 hours training to learn the game using their first interface. Teams required an average of 5 hours to learn to use subsequent interfaces. Subjects were trained until 1) they were above a specified minimum performance level, 2) performance appeared to have stabilized, and 3) team members felt confident in their understanding of the interface. There were a total of four interfaces evaluated in this experiment (one of which is not discussed in this paper), and each team was presented a randomized sequence of the four interfaces to prevent order effects.

Like the DDD subjects who used the baseline I interface in 1985, these subjects were paid an hourly rate for their participation and a cash bonus based on their average score. After each session, subjects received feedback on team performance.

INDEPENDENT VARIABLES

The main independent variable of interest was, of course, Interface. The interfaces were run across a variety of scenarios in which the External Load on the team members (i.e., tempo of incoming objects and scarcity of resources) and the Overlap of abilities and information were manipulated. This was done to find global, rather than scenario specific, interface effects. The two-way and three-way effects will be described in a subsequent paper (Saisi and Serfaty, 1988).

DEPENDENT VARIABLES

In the preliminary data analysis five dependent variables were examined. The first of these was an overall system performance measure called Final Strength. Final Strength takes into account the number of threats attacked, the values of threats, the timeliness of the attack, and the accuracy of assigning resources. Two of the contributors of Final Strength are Accuracy and Timeliness which were examined separately. The Communication Rate was also a dependent measure of interest, as well as a subjective measure of team Workload, the Subjective Workload Assessment Technique or SWAT (Reid, Shingledecker, and Eggemeir, 1981).

PRELIMINARY RESULTS

This section presents a sample of preliminary results available at this time. A complete set of results will be available in Saisi and Serfaty (1988).

Most, but not all, hypotheses were supported. The baseline I and the baseline II interfaces (Table 1) were compared on performance measures, i.e., Final Strength, Timeliness, and Accuracy, and on the process measure, Communication Rate, but have yet to be compared on the Workload dimension. No significant differences were found between these two interfaces.

Next the baseline II, human factors, and OFM-based interfaces were experimentally compared. The expected improvement on the Final Strength measure resulted (Figure 3). When teams used the OFM-based interface, they maintained a higher level of overall system performance than with the human factors interface, and the human factors interface led to improved system performance over the baseline II interface.

As expected there was a significant improvement in Timeliness for the human factors interface over the baseline II interface. The OFM-based interface equalled, but did not surpass the Timeliness improvements of the human factors interface (Figure 4).

In addition to improving on timeliness over the baseline II interface, the OFM-based interface led to a significant Accuracy improvement. The human factors interface, on the other hand, did not contribute to an increased Accuracy score over the baseline II interface (Figure 5).

As far as Communication Rate, there was a great increase in rate moving from the baseline II interface to the human factors interface and, then again, from the human factors interface to the OFM-based interface (Figure 6). Comparing the elements of Communication Rate, it can be seen that the in the increase in the human factors case is due to a high transfer rate of intended actions (this capability was not explicitly available in the baseline II interface,

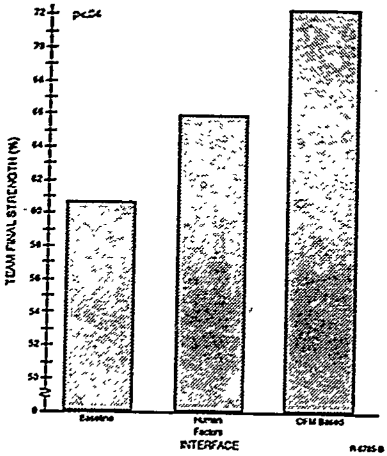


Figure 3. The Effect of Interface on Final Strength.

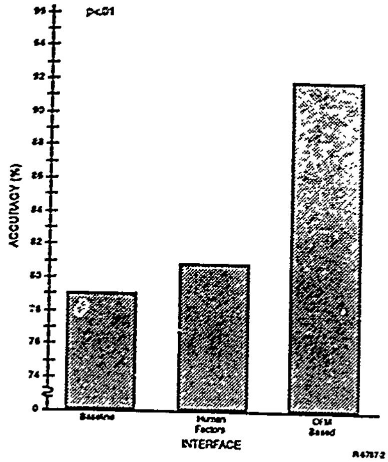


Figure 5. The Effect of Interface on Accuracy.

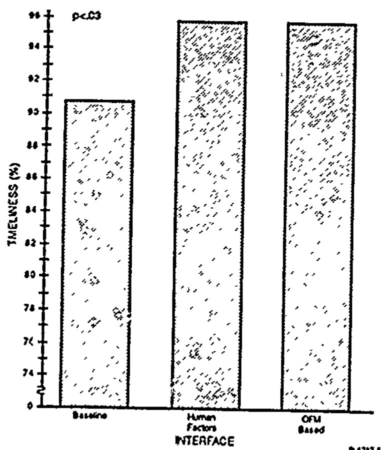


Figure 4. The Effect of Interface on Timeliness.

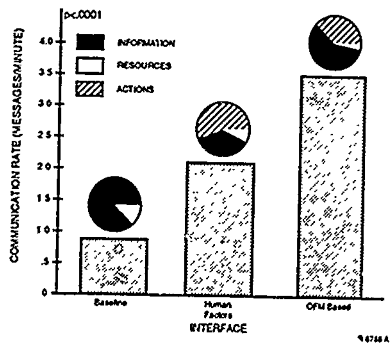


Figure 6. The Effect of Interface on Communication Rate.

although the information transfer channel may ha[illegible] [illegible]een used to express intended actions). Comparing th[illegible] [illegible]-based to the human factors interface, we see that the action transfer rate is held constant, but the information transfer rate is greatly increased. Within all three interfaces, the resource transfer is done at a low and equal rate.

Surprisingly there was not a team Workload decrease between the human factors and baseline II interface, but there was a significant drop in Workload in the OFM-based interface condition (Figure 7).

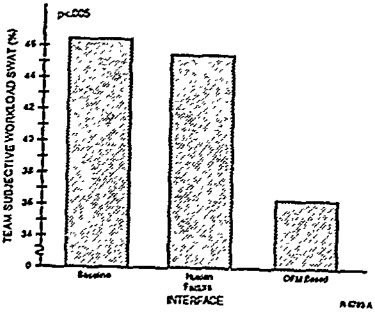


Figure 7. The Effect of Interface on Workload.

DISCUSSION

The DDD paradigm has been used as an experimental testbed for the past three years to study issues in distributed tactical command and control. During this time, it has been found that the Final Strength measure is quite inflexible, that is not easily influenced by system manipulations. The improvement in system performance through interface design is a powerful result for two reasons. First, we have uncovered a variable having a large effect on system performance even at the global level, and, second, this is an internal variable that is under the system designer's control. Velocity of incoming objects is an example of a variable that influences overall system performance but is external, i e., not controllable by a system designer.

The Final Strength can be broken into a number of components, two of these being Timeliness and Accuracy. A somewhat surprising result was that Timeliness in the OFM-based interface condition was not better than within the human factors condition. The possibility will be explored that a ceiling effect was present. In could be the case that to receive the minimally required information on the objects (i e., identity), a mean Timeliness score of 96 percent is the highest that could be realized.

Accuracy is comparable between the baseline II and human factors interface, and much higher for the OFM-based interface There are a number of contributors to the accuracy in assigning resources. The first is the number of attribute measures collected for a threat. A Probe per Threat measure was taken that indicates that no more probes (i.e., information sampled from the environment) were sent out in the OFM-based condition. However, it is shown from Communication Rate that in this condition more attribute measures were collected through communicating with one's partner. The second contributor to Accuracy in resource assignment is how well the attribute estimates are combined. The OFM-based interface is designed to aid the operator in weighing the estimates optimally. The third element of Accuracy is the mapping from threat attributes to the resources required for attacking the threat. The decisionmakers have a decision support system (DSS) to aid them with the mapping. Since this system is free to use (costs no resources or time) the OFM-based interface automatically presents the DSS results just before the decisionmaker assigns resources. The elements of Accuracy will be examined separately in the next phase of data analysis.

Communication Rate more than doubles from the baseline II to the human factors conditions. This is probably due to the simplified manual requirements of the human factors interface as well as the new ability to transfer intended actions explicitly. In moving to the OFM-based interface there was another leap in Communication Rate. The OFM-based interface was designed to promote an increased awareness of the partner's needs. The higher information exchange rate may be due to team members having more accurate mutual internal models and therefore being able to better anticipate and respond to one anothers need for information.

One of the most interesting results was the effect of Interface on Workload. The human factors and baseline II interfaces were perceived as producing the same level of Workload. The application of human factors principles to an existing display, without altering *information content* or command procedures is a common approach to interface design. Although the human factors interface was more visually pleasing (e.g., having color and shape coding) and allowed the operator to react more quickly (i e., with mouse input), the cognitive task requirements were virtually the same, which may account for the lack of a workload decrement. On the other hand, the OFM-based interface did not merely enhance the format of existing data, but used a model of the operators functions in the system to generate relevant information. This interface led to a *significant* decrease in subjective team workload.

CONCLUSION

An operator function model was developed for the DDD paradigm to represent the team decisionmaking process in a tactical command and control environment. This model was used as an interface design tool to create an interface in which the content of displays and the command structure support team decisionmaking in the system. Decisionmakers are supported in two ways. First, information is presented only when needed and is aggregated to the required level of detail. Second, commands map to the high level operator intent. Given system state, the interface leads the human operator through the sequence of actions that will most efficiently accomplish the intended goal. This alleviates the operator from determining the required low level steps and stringing together elementary control actions.

Preliminary data analysis shows that teams using the model-based interface communicated more, performed tasks more quickly and accurately, and perceived workload to be lower. The operator function model proved to be a valuable tool for the design of an interface that produced a high level of system performance without overloading the operator.

REFERENCES

Kleinman, D.L., D. Serfaty, and P.B. Luh, "A Research Paradigm for Multi-Human Decisionmaking," Proceedings of the American Control Conference, San Diego, CA, pp. 6-11, 1984.

Miller, R.A., "A Systems Approach to Modeling Discrete Control Performance", in Advances in Man-Machine Systems Research, Vol. 2, W. B. Rouse, Ed., Greenwich, CT:JAI Press, pp. 177-248, 1985.

Mitchell, C.M. and R.A. Miller, "A Discrete Control Model of Operator Function: A Methodology for Display Design," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-16, No. 10, pp. 343-357, 1986.

Mitchell, C.M. and D.L. Saisi, "Use of Model-Based Qualitative Icons and Adaptive Windows in Workstations for Supervisory Control Systems," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-17, No. 4, pp. 573-593, 1987.

Reid, G.B., C.A. Shingledecker, and F.T. Eggemeir, "Application of Conjoint Measurement to Workload Scale Development," Proceedings of the 1981 Human Factors Society Annual Meeting, pp. 522-526, 1981.

Saisi, D.L., W.C. Rooney, D. Serfaty, E.E. Entin, "Final Report on Distributed Tactical Decisionmaking, Volume 7: Model-Based Interface for Team Decisionmaking", ALPHATECH, Inc., Burlington, MA, 1988.

Saisi, D.L. and D. Serfaty, "Final Report on Distributed Tactical Decisionmaking, Volume 8: Evaluation of Model-Based Interface for Team Decisionmaking", ALPHATECH, Inc., Burlington, MA, 1988.

Serfaty, D. and D.L. Kleinman, "An Experimental Plan for Distributed Tactical Decisionmaking," Proceedings of the 8th MIT/ONR Workshop on C^3 Systems, Cambridge, MA, pp. 65-68, 1985.

Smith S.L. and J.N. Mosier, "Guidelines for Designing User Interface Software," ESD-TR-86-278, MTR 10090, The MITRE Corporation, Bedford, MA, 1986.

A COMPUTATIONAL MODEL OF EXPLANATION FOR A TACTICAL MISSION PLANNER

Lt. Mark Maybury
Rome Air Development Center
Griffiss AFB, Rome NY 13441-5700
maybury@radc-tops20.arpa

ABSTRACT

The current explanations of expert system reasoning are often incoherent or ambiguous. One problem is that explanation components fail to explicitly represent the linguistic knowledge necessary to adequately construct and present multisentential texts. A theory of the rhetoric of explanative texts is presented together with explicit rhetorical grammars and their semantic mappings onto backend knowledge formalisms.

INTRODUCTION

With the increasing complexity of expert systems -- significant components of C^3I applications -- comes an increasing need for coherent and cogent explanation of reasoning. Explanation facilities contribute to both the development and use of sophisticated systems (planners, diagnosticians, etc.). During development, explanation serves as a debugging aid. Also, explanation facilities augment system flexibility by supporting not only problem solving but also instructional tasks. From the user's perspective, justification of reasoning promotes confidence. Finally, investigation of explanation can provide an empirical test of cognitive theories of knowledge.

This research is aimed at the design and development of a computational model of textual presentations of explanation for the Air Force mission planning system, KRS [Dawson *et. al.*, 1987]. Justifying KRS' plans requires manipulation of both domain knowledge, knowledge of the process of planning, as well as linguistic knowledge of natural language utterances and their organization into texts. This work builds on a previous text generation system designed and implemented by the author [Maybury, 1987, 1988]. The major issues investigated by this work are:

- Conceptual models of the task
- Linguistic knowledge of rhetorical techniques (e.g. discourse strategies)
- Expertise level of hearer (expert, student, novice -- an expertise continuum)
- Attitude of hearer toward topic (attentive, indifferent, disinterested)
- Speaker's attitude toward topic (attentive, indifferent, disinterested)

These knowledge sources should have several effects on the text produced the natural language generator including:

- molding perspective of description or explanation (e.g. componential vs. evidential justification)
- limiting completeness of explanation
- need to sell, interest reader (illustrate why explanation important -- subtlety)
- length, verbosity, lexical choice

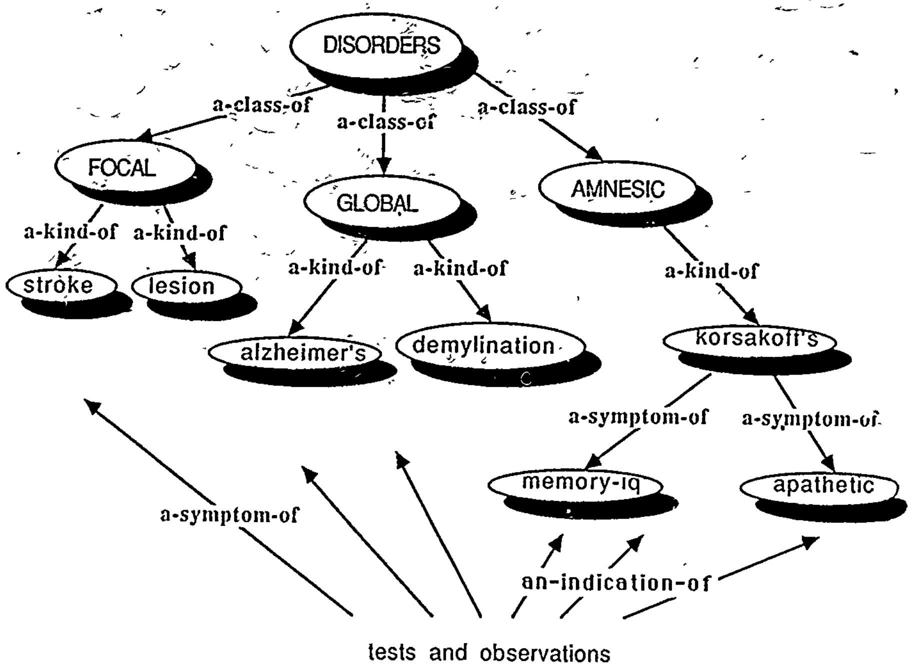


Figure 1: Relevant Portion of the Knowledge Base

EXPLANATION RHETORIC

Previous explanation systems have primarily achieved textual coherence from the actual underlying plan, rule chain, or procedure (e.g. [Appelt, 1985]). In contrast, the approach presented here is to expand on previous work on discourse strategies [McKeown, 1985] and develop an explanation rhetoric (ER) which provides textual coherence. ER are strategies that humans employ to describe or justify their reasoning. The impetus of this approach is work in text generation for presentation of the diagnostic results of frame based expert systems [Maybury, 1987, 1988]. For example in response to the query, *Why did you diagnose Korsakoff's disorder?*, typed in the functional notation *(explain Korsakoffs)*, the generator, GENNY, highlights relevant portions of the knowledge base, as in figure 1 above. Then the system generates a pool of relevant rhetorical propositions (possible utterances) by instantiation with information from the knowledge base. The propositions are organized using rhetorical schema representing common explanation strategies (a local focus algorithm controls topic shift) and then realized with a phrase structure grammar, lexicon and morphological synthesizer:

Why did you diagnose Korsakoff's disorder?

(explain Korsakoffs)

reason
evidence

cause-effect
attributive
attributive

Korsakoffs disorder is manifest because a memory-iq observation and an apathetic observation indicate damage. The memory-iq observation has a likelihood value of nine. The apathetic observation has a likelihood value of ten.

Current research in the presentation of reasoning in a tactical mission planner, KRS, is leading to a more explicit and detailed formulation of explanation rhetoric. In the following example, the user has selected the incorrect aircraft for this mission and the expert system has to express that to encourage the user to recover from the error. The planner's underlying justification is presented in figure 2.

Replacing this directly using a pattern matcher yields the current explanation implementation:

"By TARGET-AIRCRAFT-2: There is a severe conflict between the target and the aircraft since:
1. DATA: The target of OCA1002 is BE50318
2. DATA: Part of BE50318 is BE50318-SEARCH-RADAR
3. INHERITANCE: BE50318-SEARCH-RADAR is active
4. DATA: The aircraft of OCA1002 is F-111E and F-111E is not a F-4G."

One suggested output (Meteer and McDonald, 1988) is: *The target has an active search radar, but the mission aircraft are F-111Es rather than F-4Gs.* This might indeed be sufficient for a mission planning expert end user.

A different strategy would exploit all of the relevant knowledge, organizing this with rhetorical models of explanation strategies (e.g. explanation --> justification + evidence). Once again, the primitive messages (e.g. componential, justification) are selected using a focus algorithm which controls the attentional shifts in the text to ensure proper coherence.

```
(target-aircraft-2      (data         (target OCA1002 BE50318))
              (data        (powa BE50318 BE50318-SEARCH-RADAR))

              (inheritance      (is-a  BE50318-SEARCH-RADAR
                                       ELECTRONICS))
              (data        (aircraft OCA1002 F-111E)
                           ((NOTEQ F-111E 'F-4G))))
```

```
where: OCA     -- offensive counter air mission

       powa    -- part of which are
       BE      -- battle element
       NOTEQ   -- not equal
```

Figure 2: KRS' justification of an automatic planning failure

A proposed response to the simulated query, *Why was there a conflict between the target and the aircraft?*, would be:

(explain target-aircraft-2)

EXPLAN
justification
evidence

componential justification
componential elaboration/attributes
attributive
attributive justification

The OCA1002 mission plan failed because there is a severe conflict between the target and the aircraft. The target has an active search radar. The aircraft is an F-111E instead of an F-4G. An F-111E can only block radar but an F-4G can destroy it.

or

componential justification
componential elaboration/attributes
definition
definition (juxtaposition)

The OCA1002 mission plan failed because there is a severe conflict between the target and the aircraft. The target has an active search radar. The aircraft is an F-111E which can block radar. An F-4G, on the other hand, can destroy radar.

These explanation strategies were abstracted from explanative texts which humans generate. These were formalized into the explanation rhetoric presented in figure 3.

Explanation predicates are mapped to the backend knowledge formalism using a range of semantic relationships. (This mapping could be a non-trivial task, particularly if the system were not based on some semantically explicit model of knowledge, such as frames, rules, or if the knowledge itself was poorly represented.) The variety of the explanations will partially depend in part on the semantic richness of the underlying expert system. The explanation predicates are:

Explanation predicates
componential justification
evidential justification
physical/causal justification
classificatory justification
generalization justification
associative justification
characteristic justification
functional justification

The semantic relationships which correspond to these (in order) are:

Semantic relationships
aggregation: "a part of"
indication: "an indicator of"
causation: "a cause of"
classification/specialization: "a kind of"
instantiation: "an instance of"
association: "a set of"
attribution: "a property of"
capability: "a capability of"

EXPLAN	⇒	justification + evidence
reason	⇒	componential | evidential | physical/causal | classificatory | generalization | associative | characteristic | functional justification
evidence	⇒	attributive* | definition* | inference | reason

Figure 3: Explanation Rhetoric

CONCLUSION

This paper presents an approach to the presentation of expert system explanations which are guided by an explicit rhetorical model of justification. This approach is in contrast to recent work in explanation presentation which presents explanations as a trace of the causal links between events in the expert system [Paris, 1987]. By taking into account the semantic relationships, the resultant text should more clearly present the justification. This is due to the fact that lexical and syntactic selection, as well as the choice of discourse markers (e.g. "because", "therefore", "then") can be guided by the type of rhetorical predicate being realized. The empirical success of this approach is the focus of current research.

REFERENCES

Appelt, D., *Planning English Sentences*, Cambridge University Press, 1985.

Dawson, B. C., R. H. Brown, C. E. Kalish and S. Goldkind, "Knowledge-based Replanning System," RADC Final Technical Report 87-60, May 1987.

Gazdar, G., "Phrase Structure Grammar." In P. Jacobson and G. K. Pullum (eds.) *On the Nature of Syntactic Representation*, Dordrecht: Reidel, 1982.

Grosz, B. J. and C. L. Sidner, "Plans for Discourse," Harvard University TR-11-87, to appear in *Intentions in Communication*, P. Cohen, J. Morgan and M. Pollack (eds.) MIT Press, 1988.

Maybury, M. T. "A Report Generator," M Phil Dissertation, Computer Speech and Language Processing, Cambridge University Engineering Department (CUED) Cambridge University Engineering Department, September, 1987.

Maybury, M., "GENNY: A Knowledge Based Text Generation System," *Proceedings of the RAIO-88 Conference on User-Oriented Content-Based Text and Image Handling*, MIT, Cambridge, MA, March 21-24, 1988.

Maybury, M. "Towards a Portable Natural Language Generator," AVIGNON-88, *Eighth International Workshop on Expert Systems and Their Applications*, France, May 30 - June 3, 1988.

McKeown, K. R., *Text Generation*, Cambridge University Press, 1985.

McDonald, D. and Meteer, M. "From Water to Wine: Generating Natural Language Text from Today's Applications Programs," *ACL 2nd Conference on Applied Natural Language Processing*, Austin, TX, Feb 9-12, 1988.

Paris, C. "Combining Discourse Strategies to Generate Descriptions to Users Along a Naive/Expert Spectrum," *Tenth Annual IJCAI*, Milan, Italy, 23-28 August, 1988.

Sidner, C. L. "Focusing in the Comprehension of Definite Anaphora," In Brady and Berwick (eds.), *Computational Models of Discourse*, 1983, pp. 267-330.

Stevens, A. and C. Steinberg, "A Typology of Explanation and its Application to Intelligent Computer Aided Instruction," BBN Technical Report NO. 4626, March 5, 1981.

COORDINATION IN ORGANIZATIONS WITH DECISION SUPPORT SYSTEMS*

Jean-Louis M. Grevet
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139 USA

ABSTRACT

A methodology to model, analyze, and evaluate coordination in organizations with decision support systems is presented. The issues of inconsistency of information and synchronization are emphasized. Predicate Transition Nets are used as the basic technique for representing organizational structures and for characterizing the coordination of processes. Protocols of interaction are modeled by transitions for which the rule of enablement is that the decisionmakers, when interacting, must refer to the same state of the environment. Two measures of coordination are then introduced, information consistency and synchronization. These measures are defined on the basis of the attributes of the tokens belonging to the input places of transitions modeling interactions. A recently developed simulation system for Predicate Transition Nets is used for investigating, through an example, the dynamics of such organizations and for analyzing how a decision support system can alter the coordination in an organization.

INTRODUCTION

The decisionmaking by organization members implementing the command and control process must be coordinated in order to improve their effectiveness. Decision aids, which are part of the C^3 systems, aim at increasing the ability of decisionmakers to perform their mission effectively. By offering faster processing capabilities as well as access to databases, they may help the organization members to achieve the requirements of the mission. However, decision aids also increase the possible alternatives among which to choose in order to process information, and in so doing, modify the nature of the decisionmakers' activities. In this context, it is important to *evaluate the extent to which decision aids*, and more particularly decision support systems (DSS), *can alter the coordination of the various decision-making processes.*

The framework used to address this problem is the quantitative methodology (Levis, 1984, 1988) for the analysis and evaluation of alternative organizational structures. In order to provide some insight on the cohesiveness of organizations carrying out well-defined tasks, a mathematical description of coordination is developed as it relates to decision-making processes. The Predicate Transition Net formalism (Genrich and Lautenbach, 1981) used in this paper builds on Petri Net theory (Brams, 1983), but allows the modeling of coordination based on the attributes of symbolic information carriers in the net. In this model, when decisionmakers interact, they must have some protocol to recognize that they are exchanging information pertaining to the same event. Two measures for evaluating coordination are introduced, information consistency and synchronization. The latter measure relates to the value of information when the decisionmakers actually process it.

A generic model of a decision-maker interacting with a DSS is presented. The focus is on the architecture of the system and on the different system components that the decision-maker can access. DSS's have become an increasingly important part of the military Command, Control and Communications (C^3) systems (Waltz and Buede, 1986). In this context, the DSS's, also called battle management systems, automate the fusion of data concerning the tactical situation and the quantitative evaluation of alternative courses of action.

Decision aids are defined as any technique or procedure that restructures the methods by which problems are analyzed, alternatives developed and decisions taken. Keen and Scott Morton (1978) emphasize that decision support systems, a particular form of decision aids, have specific advantages:

"(i) *the impact is on decisions in which there is sufficient structure for computer and analytic aids to be of value, but where decisionmakers' judgment is essential.*

(ii) *the payoff is in extending the range and capability of decisionmakers' decision processes to help them improve their effectiveness.*

(iii) *the relevance for decisionmakers is the creation of a supportive tool under their own control, which does not attempt to automate the decision process, predefine objectives, or impose solutions.*"

Thus, DSS's do not automate the decisionmaking process, but must facilitate it. When confronted with a particular task, the decisionmaker keeps the choice of performing it by himself or requesting information from the DSS. This selection depends on the reliability of the DSS or, more exactly, on the extent to which the organization members rely on it.

The evaluation of the effectiveness of a decisionmaking organization consisting of human decision makers aided by a DSS is a complex issue: many interrelated factors affect the effectiveness of the overall system, e.g., the limited information processing capacities of the decision makers, the hardware and software characteristics of the DSS, or the extent to which the organization members use and rely on the decision aid. One important question is to know whether or not the overall organization, when aided by the DSS, is more effective in fulfilling its mission.

*This work was carried out at the MIT Laboratory for Information and Decision Systems with support provided by the Office of Naval Research under Contract No. N00014-84-K-0519 (NR 649-003).

Earlier work has assessed the impact of preprocessors (Chyen and Levis, 1985; Weingaertner and Levis, 1987) and databases (Bejjani and Levis, 1985) on the workload of the decisionmakers However, it seems necessary to measure the extent to which the DSS can affect the coordination of the various decisionmakers who use it. Indeed, the introduction of a DSS in an organization can lead either to an improvement or to a degradation of its cohesiveness, depending on the functionality and capabilities of the DSS, as well as on the perception of and access to the DSS that the decisionmakers have.

Simulation of Predicate Transition Nets is introduced to investigate the dynamics of decisionmaking processes - especially phenomena not captured by analytically tractable models, such as the use of different protocols by different decisionmakers. An example demonstrates that decision aids can degrade the coordination of decision-making organizations by affecting the dynamics of the activities and by increasing the number of alternatives for processing information.

PREDICATE TRANSITION NET MODEL OF COORDINATION

The organizations under consideration consist of groups of decisionmakers processing information originating from a single source and who interact to produce a unique organizational response for each input that is processed. In Petri Nets terms, there exists a source place, p_{so}, and a sink place, p_{sk}. A resource place, p_{rs}, is introduced to model the limited organizational resources. A transition t_{par} models the partitioning of the input from the single source into inputs received by different organization members or C^3 system components. Furthermore, if several decisionmakers provide responses that must be fused in order to obtain the organizational response, this stage of response fusion is modeled by the transition t_{rf} (see Figure 1).

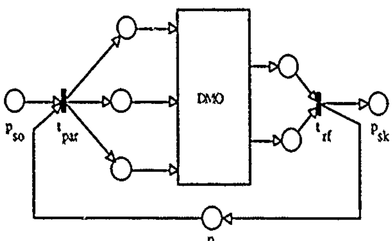


Fig. 1 Petri Net of Interactions between a Decisionmaking Organization and the Environment

The source p_{so} generates single tokens that arrive sequentially and are marked with the arrival time. The task is modeled by the alphabet $\mathcal{X} = \{x_1, .., x_n\}$ and a probability distribution prob($x = x_i$), denoted by prob(x_i) defined on $\mathcal{X}$ The set of subsets of $\mathcal{X}$ is denoted by $\Pi(\mathcal{X})$, then,

$$\Pi^*(\mathcal{X}) = \Pi(\mathcal{X}) - \{\emptyset\} \qquad (1)$$

where $\emptyset$ denotes the empty set.

A clock is used to mark the instants T_c at which the process is observed. In accordance with the formalism of Timed Petri Nets, this clock provides non-negative rational numbers.

Distinguishability of Tokens

The fundamental assumption of the model is that a decisionmaker can process only one input at a time in any of his internal stages it follows that any other input that is ready to be processed by the same stage waits in memory. Therefore, queues of tokens can build in the places of the system.

At any internal stage of the decision-making process, a decisionmaker can discriminate between different items of information on the basis of three characteristics:

- the time T_n at which the inputs that these items of information represent entered the organization.
- the time T_d at which the item of information entered the internal stage where it is currently located.
- the class C associated with any item of information by the previous processing stage.

The definition of the attributes T_n, T_d and C derives from the following considerations:

(i) Since inputs originate from a single source, one at a time, the attribute T_n corresponds to the time at which the input represented by this token entered the organization.

(ii) Since in stochastic timed Petri Nets, the firing of any token takes an amount of time that depends on the processing time of the corresponding transition. One can assign to any token in a place p the time T_d at which it entered this place.

(iii) Since tasks are modeled by the alphabet $\mathcal{X} = \{x_1,..., x_n\}$, it is assumed that each place p is associated with a partitioning $\mathcal{D}(p)$ of this alphabet. The number of elements of this partitioning is denoted by e(p). This partitioning is such that $\mathcal{D}(p) = \{D(p,1),.. , D(p,e(p))\}$ where $D(p,i)$ denotes an element of $\Pi^*(\mathcal{X})$. Thus, the third attribute C of each token belongs to a certain partitioning $\mathcal{D}(p)$ of $\mathcal{X}$, this partitioning depending on the place p where the token is located.

The different resources that the organization has are assumed to be indistinguishable; this might not be the case when organizational resources are allocated to different inputs in accordance with some doctrine. In the same way, the resources that represent the decisionmakers' processing capacities are not distinguishable. Consequently, three types of places are defined: *Memory places* carry information internally processed by each decisionmaker; *structural places* carry information exchanged between a decisionmaker and the environment or other organization members; and *resource places* that model the limitation of resources that constrains the processing of information by individual DMs. Memory and structural places contain tokens that have an identity since they model information carriers, while resource places contain tokens with no identity

Each place is associated with one of the variables χ or ϕ The variable χ takes its values in the set X where each element of X is a color represented by (T_n, T_d, C) A token with an identity is an individual that is assigned a color. All the tokens with no identity are denoted by the color ¢. The variable ϕ takes its values in the set Φ such that $\Phi = \{¢\}$.

The marking of PN is defined as follows· For each place p, M(p) assigns to each value of the variable associated with p a non-negative integer number which represents the number of tokens in the place that have the corresponding color If m designates a certain color, M(p)[m] will denote this number Since each color m corresponds to a triplet (T_n, T_d, C), this number will be also denoted by $M(p)((T_n, T_d, C))$. In the case of a resource place, the tokens can have only the color c and $M(p)[c]$ can be denoted simply by $M(p)$.

The following example (Figure 2) illustrates these definitions:

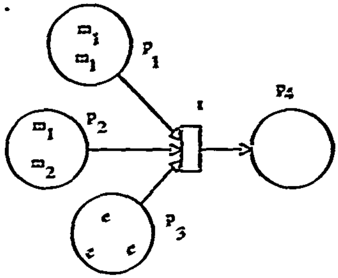


Fig. 2 Example of Marking

In this example, the following relations hold:

- $m_1 \in X, m_2 \in X$.
- $M(p_1)[m_1] = 2; \forall m \in X-\{m_1\}, M(p_1)[m] = 0$.
- $M(p_2)[m_1] = 1; M(p_2)[m_2] = 1;$
  $\forall m \in X-\{m_1, m_2\}, M(p_2)[m] = 0$.
- $M(p_3) = 3$.
- $\forall m \in X, M(p_4)[m] = 0$.

The firing of a transition t is characterized by the following:

- if p' is a resource place, m' is the color c.
- if p and p' are memory or structural places, m and m' are elements of X.

The attribute T_n characterizes one and only one input since the source generates one input at a time. Furthermore, two representatives of the same input cannot stand in the same place. Indeed, the net is an Marked Graph and, so, each place has only one input transition which produces in each of its output places only one token per firing.

Proposition 1: A place cannot contain two tokens which have the same attribute T_n.

Protocols of Interaction

One must recall that the set of input places of any transition t can contain a resource place. The rule according to which the resource place must contain at least a token in order for t to be enabled will apply. However, since resource places do not constrain the rule of enablement of a transition, but by requiring the presence of a token, the discussion on enablement that follows focuses on structural and memory places. The Petri Net model of transitions where fusion of data is done is shown in Figure 3.

When the fusion of data is performed by a decisionmaker, *only one of the places* $p_1, \ldots, p_r$ is a memory place. We denote it by p_k. Any rule of enablement can be introduced at this point. Let M denotethe marking of the net. Then, two possible rules are.

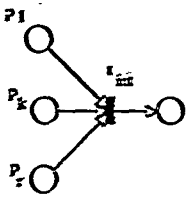


Fig. 3 Petri Net Model of Interaction with Fusion of Data

Rule 1: t_{int} is enabled, if and only if all its input places contain a token with the same value of the attribute T_n

Rule 1 means that the transition t_{int} is enabled if and only if all the places of its preset contain at least a representation of the same input. This results from the fact that memory and structural places contain only tokens of the (T_n, T_d, C) type, and that tokens having the same attribute T_n represent the same input. From the organizational standpoint, it means that, when decisionmakers interact, they must refer to the same input.

Rule 2: t_{int} is enabled if and only if rule 1 applies or there exists a token in the memory place p_k which has been in it for more than d units of time.

Rule 2 models the interactions where decisionmakers wait for information from other parts of the organization but only for a certain amount of time.

In this paper, a transition will be enabled if and only if rule 1 is verified. In the case of internal transitions, rule 1 is always verified when all its input places have a token since the preset contains only one place that is not a resource place. It means that the attributes T_n^i of the colors $m_1, \ldots, m_r$ must have the same value.

Token Selection

The problem of token selection arises since the tokens are distinguishable. These rules operate on the tokens of the input places that enable the transition t. This is illustrated by the example of Figure 4 where rule 2 of enablement applies.

Suppose that:

- $m_1 = (T_n^1, T_d^1, C^1)$; $m_1' = (T_n^1, T_d^1, C'^1)$, $m_1'' = (T_n^1, T''_d{}^1, C''^1)$.
- $m_2 = (T_n^2, T_d^2, C^2)$; $m_2' = (T_n^2, T_d'^2, C^2)$; $m_2'' = (T_n^2, T''_d{}^2, C''^2)$.

Since the enabling condition is that the tokens have the same arrival time T_n, it follows that the transition t is enabled by both sets $\{m_1, m_1', m_1''\}$ and $\{m_2, m_2', m_2''\}$. Therefore, a rule must exist to decide which token set will be removed by the next firing of transition t.

It is assumed that this rule works as follows: it selects a token in a certain place p of the preset of transition t ; then the set of tokens removed is the one to which the token selected belongs. Therefore, before applying the rule, it is necessary to decide in which place p the selection will be done. One can see on the example of Figure 4 that p_1, p_2 and p_3 contain each two tokens that enable transition t. This means that the selection of the tokens that will be fired next can be done in place p_1 or place p_2 or place p_3. Different strategies can be applied to choose the place. In this paper, the choice of the place on which the token selection rule will apply is done according to some well-known rule PS(t), for each transition t, given the state of the system. Then, the selection of a token in this place determines an attribute T_n. The knowledge of this attribute allows to select the corresponding tokens in the other places. In the example of Fig. 4, if PS(t) selects p_1, the token selection rule must discriminate between m_1 and m_2 and m_3. If m_1 is selected, then m_1' and m_1'' are automatically selected in places p_2 and p_3.

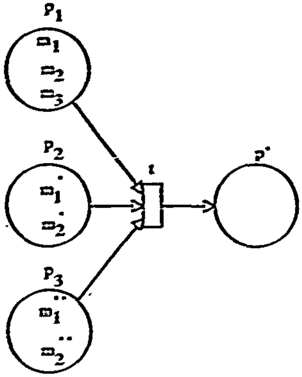


Fig. 4 Token Selection

Proposition 2: The selection in the place p of a token among the tokens that can be fired by transition t determines uniquely the tokens that will be fired in the other places. Once a token has been selected in the place p, its attribute T_n corresponds to one and only one token in any other place of the preset of the transition t.

Four types of rules of selection PS(t) can be considered:

(i) rules that discriminate with respect to the attribute T_n.
(ii) rules that discriminate with respect to the attribute T_d.
(iii) rules that discriminate with respect to the attribute C.
(iv) rules that combine different rules of the previous types.

Some example of possible rules are the following:

FIFO: the decisionmaker can decide to process first the inputs that entered the organization first. In this case, the token with the lowest T_n is selected.

LIFO. the decisionmaker decides to process first the inputs that entered the organization last. Then, the token with the highest T_n is selected.

LOCAL FIFO (LFIFO): the decisionmaker decides to process first the inputs that entered the internal stage where they currently are first. The token with the lowest T_d is selected.

LOCAL LIFO (LLIFO): the decisionmaker processes first the inputs that entered the internal stage where they currently are last. The token with the highest T_d is selected.

PRIORITY. the decisionmaker can assign priorities to certain classes of inputs, i.e., can set priorities on the basis of the attribute C. He selects first the items of information with the highest priority.

MIXED: if several pieces of information have the same highest priority, the decisionmaker can then decide to apply some rule of the type (i) to (iv) to discriminate between them.

CHARACTERIZATION OF COORDINATION

The Petri Net representation of the transitions considered in this section is shown in Figure 2. The characterization of the coordination for an interaction t_{int}, using the Predicate Transition Net model introduced in the previous section, derives from the definition of an order relation on the set of tokens fired by transition t_{int}. The following relations are defined:

Ψ_1 is a binary relation defined by:

$$((x, y, z)\,\Psi_1\,(x', y', z')) \Leftrightarrow ((x = x') \text{ and } (y \leq y')) \quad (2)$$

Ψ_2 is a binary relation defined by:

$$((x, y, z)\,\Psi_2\,(x', y', z')) \Leftrightarrow ((x = x') \text{ and } (z = z')) \quad (3)$$

Ψ_3 is a binary relation defined by:

$$((x, y, z)\,\Psi_3\,(x', y', z')) \Leftrightarrow (((x, y, z)\,\Psi_1\,(x', y', z')) \text{ and } ((x, y, z)\,\Psi_2\,(x', y', z'))) \quad (4)$$

The relation Ψ_3 defines an order relation on the set X. Let $m_1, \ldots, m_r$ denote the elements of X which represent the colors of the r tokens removed from places $p_1, \ldots, p_r$, respectively, by transition t_{int}; let m_k denotes the color of the token removed from the memory place p_k. Furthermore, each color m_i corresponds to some triplet (T_n^i, T_d^i, C^i).

The firing of t_{int} is synchronized if, and only if:

$$\forall i \in \{1, \ldots, r\} \quad (T_n^i, T_d^i, C^i)\,\Psi_1\,(T_n^k, T_d^k, C^k) \quad (5)$$

This definition allows to discriminate between firings that are synchronized and firings in which one or several tokens m_i arrive in their respective places later than m_k in p_k.

The firing of t_{int} is consistent if, and only if:

$$\forall (i, j) \in \{1, \ldots, r\} \times \{1, \ldots, r\}, \; (T_n^i, T_d^i, C^i)\,\Psi_2\,(T_n^j, T_d^j, C^j) \quad (6)$$

i.e., the data fused by DM_k are consistent, if they correspond to the same class C. On this basis, the following definition for the coordination of an interaction is obtained:

The firing of t_{int} is **coordinated** if, and only if, it is **synchronized** and **consistent**.

It is possible now to characterize a coordinated transition firing by the order of arrival of the tokens in the places of its preset.

Proposition 3. When the firing of $m_1, \ldots, m_r$ by t_{int} is coordinated, the relation Ψ_3 induces an order relation on the set $\{m_1, \ldots, m_r\}$ for which m_k, the token of the memory place, is the unique greatest element.

The definition of coordination applies to a single interaction. The definitions of the coordination of a single task, i.e., for a sequence of interactions concerning the same input, as well as for all tasks executed are as follows.

The execution of a task is coordinated if, and only if, it is coordinated for all interactions that occur during the task.

The execution of a Petri Net PN is coordinated if, and only if, it is coordinated for all the tasks performed.

INFORMATION CONSISTENCY

Given an interaction stage, t_{int} denotes the interactional transition that models this stage in the Petri Net representation, as shown in Figure 3. At each transition t_{int}, the decisionmaker DM_h associates a class C^h with each input x_i; this class is denoted by $C^h(x_i, t_{int})$ and belongs to $D(p_h)$, a partition of the alphabet X, that the designer defines a priori.

In order to achieve a higher consistency, the designer has to ensure that the r decisionmakers who interact in a particular stage are provided with the same set of classes; therefore, it is assumed that:

$$\forall (i, j) \in \{1, \ldots, r\} \times \{1, \ldots, r\}, \quad D(p_i) = D(p_j) \qquad (7)$$

If $m_1, \ldots, m_r$ designate the colors of the tokens in the preset of t_{int} that correspond to input x_i and that are fired by t_{int}, then the quantities $C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int})$ denote their attribute C. Let $V(x_i, t_{int})$ designate the vector $(C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int}))$, element of $[\Pi^*(X)]^r$. Let $\text{prob}(C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int}))$ denote the probability of having tokens with attribute $C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int})$ for the input x_i at the stage t_{int} in places $p_1, \ldots, p_r$. It will be written as $\text{prob}(V(x_i, t_{int}))$. If $z(V(x_i, t_{int}))$ is the number of subsets of two elements $\{C^a(x_i, t_{int}), C^b(x_i, t_{int})\}$ of $\{C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int})\}$, we have:

$$z(V(x_i, t_{int})) = \binom{r}{2} = \frac{r!}{2!\,(r-2)!} \qquad (8)$$

where $n(V(x_i, t_{int}))$ is the number of subsets of two elements $\{C^a(x_i, t_{int}), C^b(x_i, t_{int})\}$ of $\{C^1(x_i, t_{int}), \ldots, C^r(x_i, t_{int})\}$ such that $C^a(x_i, t_{int}) = C^b(x_i, t_{int})$. Finally:

The degree of information consistency for stage t_{int} and input x_i is:

$$d(x_i, t_{int}) = \sum_{V(x_i, t_{int})} \text{prob}(V(x_i, t_{int})) \frac{n(V(x_i, t_{int}))}{z(V(x_i, t_{int}))} \qquad (9)$$

By adding the degrees of information consistency $d(x_i, t_{int})$ for each organizational interaction t_{int} and each input x_i and weighing by the probability of having that input, one can measure the organizational degree of information consistency, D, for the task at hand:

$$D = \sum_{x_i} \text{prob}(x_i) \sum_{t_{int}} d(x_i, t_{int}) \qquad (10)$$

This measure varies between 0 and 1, with 1 being the ideal information consistency of all interactions for the whole task.

SYNCHRONIZATION

The total processing time of an item of information for decisionmaker DM_i consists of two parts: (i) the total time T_i^t during which the decisionmaker actually operates on the information; and (ii) the total time T_i^p spent by the information in memory prior to being processed.

The time T_i^p is due to two factors: (i) Information can remain in the memory of the decisionmaker until he decides to process it with the relevant algorithm. Since an algorithm cannot process two inputs at the same time, some inputs will have to remain unprocessed in memory for a certain amount of time until the relevant algorithm is available. (ii) Information can also remain in memory because the decisionmaker has to wait to receive data from another organization member.

An organization is not well synchronized when the decisionmakers have to wait for long periods before receiving the information that they need in order to continue their processing. Conversely, the organization is well synchronized when these lags are small.

The sojourn time $T_s^h(x_i, t_{int})$ of the token m_h, representing the input x_i in the place p_h of the preset of transition t_{int}, measures the amount of time spent by the token in the place before it is fired:

$$T_s^h(x_i, t_{int}) = T_e - T_a^h \qquad (11)$$

This quantity is zero when the firing occurs at the same time the token enters the place. Conversely, it differs from zero when the firing cannot be initiated at the same time the token enters the place. The following quantity can now be introduced.

$$S_L^{hj}(x_i, t_{int}) = T_s^h(x_i, t_{int}) - T_s^j(x_i, t_{int}) \qquad (12)$$

The quantity $S_L^{hj}(x_i, t_{int})$ measures the difference between the sojourn times of the tokens representing x_i in p_h and p_j, i.e., the difference between the lengths of time that the information sent by DM_h and DM_j to DM_k remained inactive before being processed.

When p_k represents the memory place, $S_L^{kj}(x_i, t_{int})$ will be computed for each structural place p_j. If it is positive, it implies that the token m_k has spent more time in p_k than the token m_j in p_j. If it is negative, the opposite is true. In the latter case, there is no degradation of synchronization, because DM_k is not ready to process the next task.

Let F(x) denote the function defined on the set of rational numbers, Q, by:

$$\forall x \in Q, \; (x \geq 0) \Rightarrow (F(x) = x)$$
$$(x < 0) \Rightarrow (F(x) = 0) \qquad (13)$$

Let $INT(t_{int})$ denote the set of indices h for the structural places p_h of $Pre(t_{int})$. Then the total lag for the transition t_{int} in processing input x_i, $S(x_i, t_{int})$, can now be defined as follows.

$$S(x_i, t_{int}) = \max_{h \in INT(t_{int})} (\bar{F}[S_L^{h}(x_i, t_{int})]) \quad (14)$$

or, from (12),

$$S(x_i, t_{int}) = \max_{h \in INT(t_{int})} (\bar{F}[T_s^{k}(x_i, t_{int}) - T_s^{h}(x_i, t_{int})]) \quad (15)$$

Thus, $S(x_i, t_{int})$ measures the maximum of all the lags during which the decisionmaker has to wait before having all the information he needs to continue his processing. The measure S does not take into consideration the items of information for which the decisionmaker does not wait.

The measure of synchronization for decisionmaker DM_k and the rest of the organization, S_k, is defined as:

$$S_k = \sum_{x_i} prob(x_i) \sum_{t_{int} \in \lambda(k)} S(x_i, t_{int}) \quad (16)$$

It is the expected value of the sum of the maximum lags for the interaction stages executed by decisionmaker DM_k for the inputs x_i.

The measure of synchronization for the organization, S_T, is given by:

$$S_T = \sum_{x_i} prob(x_i) \sum_{t_{int} \in \lambda} S(x_i, t_{int}) \quad (17)$$

It is the expected value of the sum of the maximum lags over the overall decisionmaking process for the inputs x_i.

On the one hand, the measures S_k, for each k, and S_T achieve their best values when they are zero. On the other hand, there is no upper bound on the values taken by these measures; they grow to infinity if a deadlock occurs. Since each interactional transition t_{int} belongs to one decisionmaker, and one only, the following relation holds:

$$S_T = \sum_{k} S_k \quad (18)$$

Thus, one can compute the contribution of each individual decisionmaker DM_k to the total synchronization measure S_T for the organization by taking the ratio S_k/S_T.

A MODEL OF A DECISION SUPPORT SYSTEM

The amount of data that must be handled by C^3 systems for a typical mission is very large For example, the antisubmarine warfare (ASW) mission requires the surveillance of a vast area where multiple sensors gather information on the environment. The typical information requirements (Waltz and Buede, 1986) are the following:

- the surveillance area covers 2000 × 2000 km.
- the sensor systems consist of 4 surveillance aircraft, 12 ASW ships, and 2 ASW submarines.
- the number of targets can be as high as 200.
- the number of reports per minute ranges from 1000 to 5000.

In this context, there is clear need for a computerized decision aiding system for the coordination of the activities of the various decisionmakers. Such a decision support system can modify the activities of a decision-maker because the latter has to consider the possibility of querying the system (Weingaertner and Levis, 1987). For each input and each stage of his internal decisionmaking process, the decisionmaker must make meta-decisions concerning the use of the DSS. These meta-decisions are of three types:

- the DM does not query the DSS and performs all processing by himself.
- the DM sends a query to some component of the system and relies totally on the response.
- the DM sends a query to some component of the system, but compares its response with his own assessment.

When several decisionmakers use a DSS for a common task, the DSS can increase or decrease the coordination of the group.

It is not possible to define a generic type of decision support system because DSS's are, in general, application-oriented and, therefore, quite specific to the organizations which use them and to the task that must be performed. The following model takes into account several capabilities and characteristics which are common to most of the real systems. In particular, it takes into consideration the fact that most real DSS's have facilities shared by several users and facilities accessed individually. From a physical standpoint, the DSS consists of a mainframe shared by the organization and which is accessed by the decision-makers through remote intelligent terminals and a communication network. The terminals are called "intelligent" to the extent that they provide the users with the opportunity to do local processing without querying the central system.

The DSS provides a multiple-access capability to the decisionmakers who can query it in parallel. Several databases are stored in the mainframe so that a decisionmaker can get information concerning the state of the environment as well as the possible responses that he can give to any input; it implies that the decisionmaker can query the database both in his Situation Asessment stage and in his Response Selection stage.

The applications implemented on the system do not embody any heuristic and do not develop alternative solutions. They implement models and doctrines well known to the decision-makers. Consequently, the processing of any particular task by DM_i involves some or all of the four essential components described in Figure 5: the decisionmaker DM_i, the intelligent terminal i that he uses, the communication network, and the mainframe.

For each of the three paths illustrated above, the amount of time that it takes to process the input for each internal stage of DM_i depends on several factors:

(i) in path 1, the decision-maker processes the information by himself, this takes an amount of time equal to the processing time of the corresponding protoco..

(ii) in path 2, the decision-maker uses only the intelligent terminal. The total amount of time taken by this operation corresponds to the sum of the following delays:

- time spent by the decision-maker to query the terminal,
- time spent by the terminal to process and display the information;
- time spent by the decision-maker to assess the response.

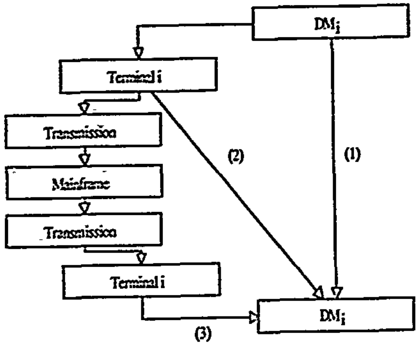


Fig. 5 DM_i Interacting with the DSS

(iii) in path 3, the decision-maker uses the terminal as a dumb terminal to query the mainframe. The total delay of this operation is the sum of the following delays:

- time spent by the decision-maker to query the mainframe;
- time spent by the terminal to access the network;
- time of transmission to the mainframe;
- time spent by the mainframe to recognize the query and initiate the processing;
- time spent by the mainframe to process the information;
- time spent by the mainframe to access the network;
- time of transmission to the terminal;
- time spent by the terminal to display the information;
- time spent by the decision-maker to assess the response.

The use of the mainframe involves the execution of operations that can take an amount of time which depends to a large extent on the physical configuration of the system. In particular, the delay of transmission through the communication network can vary over a wide range according to the specific route use which depends, in turn, on the origin and the destination. Furthermore, a query to the mainframe may be much more subject to errors due to noise and the distortion in the transmission than a query to the intelligent terminal.

Petri Net Model of Decisionmaker Aided by a DSS

The Petri Net model of a decisionmaker DM aided by a DSS is given in Figure 6 This model represents the different information flow paths that exist when a DM interacts with the DSS at any internal stage of his decisionmaking process. Figure 6 illustrates the information flow paths for the case where the DM uses only one algorithm f for performing his task. The symbols in the figure are defined as follows:

- u is the decision variable for choosing between the five alternatives:

  (1) DM performs the stage by himself.
  (2) DM queries the mainframe, performs his own processing, and compares the two results.
  (3) DM queries the intelligent terminal, performs his own processing, and compares the two results.
  (4) DM queries the mainframe and relies on its response.
  (5) DM queries the intelligent terminal and relies on its response.

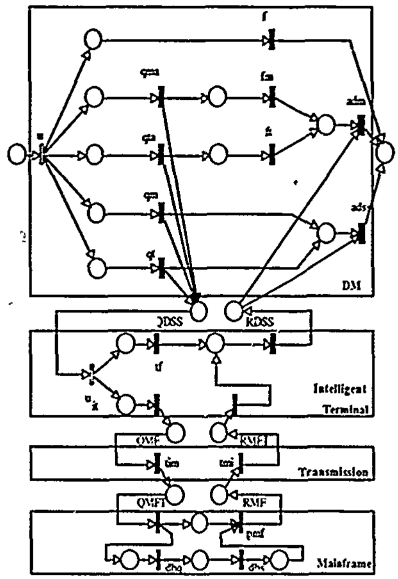


Fig. 6 Petri Net Model of DM Aided by the DSS

- qma is the algorithm used by DM to query the mainframe in alternative 2.
- qta is the algorithm used by DM to query the intelligent terminal in alternative 3.
- qm is the algorithm used by DM to query the mainframe in alternative 4.
- qt is the algorithm used by DM to query the mainframe in alternative 5.
- fm is the algorithm that DM executes when he has queried the mainframe in alternative 2.
- ft is the algorithm that DM executes when he has queried the intelligent terminal in alternative 3.
- adm is the algorithm used by DM to assess the response of the DSS and to compare it with the result of his own processing in alternatives 2 and 3.
- adss is the algorithm used by DM to assess the response of the DSS in alternatives 4 and 5.
- QDSS is the query sent by DM to the DSS.
- RDSS is the response sent by the DSS to DM
- u_{it} is the decision variable which determines whether the intelligent terminal or the mainframe must process the query.
- tf is the algorithm performed by the intelligent terminal to process the query.
- QMF is the query sent by the intelligent terminal to the mainframe.
- RMFT is the response from the mainframe transmitted by the

network to the intelligent terminal.

- tim is the protocol of transmission from the intelligent terminal to the mainframe.
- tmi is the protocol of transmission from the mainframe to the intelligent terminal.
- QMFT is the query from the intelligent terminal transmitted by the network to the mainframe.
- RMF is the response from the mainframe.
- pmf is the algorithm performed by the mainframe for processing the query.
- dbq is the algorithm that queries the database.
- dbs is the algorithm that performs the search in the database.

This model shows that the decisionmaker interacts with the DSS by fusing the information that the latter produces. Therefore, it is possible to evaluate the synchronization between DM and the DSS.

The places labelled QDSS and RDSS represent the structural places that contain the information exchanged by DM and the DSS. In accordance with the Predicate Transition Net model, the transitions adm and adss are the only interactional ones. These transitions will fire only if the tokens in their input places have the same attribute T_n, i.e., they correspond to the same input from the environment. The measure of the synchronization between DM and the DSS evaluates, for each input and each stage, the sojourn time of the item of information in the memory place of the preset of adm or adss. Since the emphasis in this study is the coordination between DMs, for simplicity, it is assumed that the synchronization between DM and DSS is perfect The same approach can be used to analyze the case when synchronization between DMs and the DSS is not perfect.

EXAMPLE

The impact of a decision support system on the coordination of a two-person organization is the key question addressed in this example. The degradation of the synchronization of a decisionmaking organization can result from two types of factors:

- the dynamics of the activities which lead the decisionmakers to process various inputs with different priority orders.
- the information flow paths that each decisionmaker uses to perform his task.

The impact of the first category of factors on the decisionmaking process was discussed in Grevet et al. (1988).This example assesses the second type of factors. Such a situation arises when the decisionmakers are provided with a DSS which allows them to access different local or remote computer facilities. The DSS can alter significantly the coordination of the activities, depending on the configuration of the system with respect to the organization.

The Organization and the Task

The example presented in this section aims at modeling the organizational structure and decisionmaking activities of a two-person organization in a simple ASW context The task models a mission of surveillance that consists of listening to detect enemy submarines. In such an environment, the use of decision support systems to process the signals and discriminate between them is necessary.

The organization consists of a submarine and a surface ship which are in charge of tracking enemy submarines It is a hierarchical organization where the submarine is the subordinate and the surface ship the commander. This example has been studied from another standpoint by Papastavrou (1986) The Petri Net model of such an organization is presented in Figure 7.

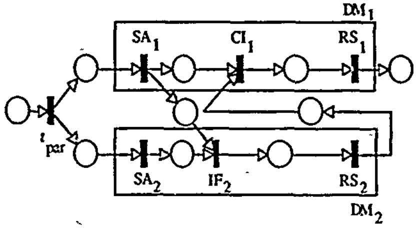


Fig. 7 Petri Net Model of Subordinate (DM_1) and Commander (DM_2)

The decision-making process of the commander and the subordinate have three stages each. In the Situation Assessment stages, they assess the signals that they receive from the environment. The subordinate sends the result of his own assessment to the commander, who fuses in the Information Fusion stage this information with his own assessment On the basis of the result of this interaction, the commander identifies the signal and produces an order which is the sent to the subordinate. The latter interprets the order in the Command Interpretation stage and produces the organizational response

The task is modeled as the alphabet X and the probability distribution prob(x) such that:

$$X = \{x_i = a_ib_ic_id_ie_if_i \mid (a_i,b_i,c_i,d_i,e_i,f_i) \in \{0,1\}^6\}$$

$$\forall (x_i, x_j) \in X \times X, \quad prob(x_i) = prob(x_j)$$

Therefore, each input consists of an ordered string of six bits. There are 64 possible inputs that represent the signals that must be identified by the organization in order to produce the response. It is assumed, furthermore, that these inputs are equiprobable, so that the probability distribution prob(x) is defined by:

$$\forall x_i \in X, \ prob(x = x_i) = \frac{1}{64} \qquad (19)$$

The organization can produce four responses, labelled R_1, R_2, R_3 and R_4:

- if the bits a_i and d_i are both equal to 0, the signal does not come from an enemy submarine and, therefore, the submarine DM_1 should not do anything. This response is R_1. The probability of having such an input is 1/4
- if b_i and e_i are both equal to 0, the signal comes from an enemy submarine which is trying to test the capabilities of submarine DM_1 This one should deceive it by under-reacting. This response is R_2. The probability of having such an input is 3/16.
- if the bits c_i and d_i are both equal to 0, the signal comes from an enemy submarine which is moderatly threatening submarine DM_1 The latter should over-react to this threat to deter the enemy submarine This response is R_3 The probability of having such an input is 9/64

- otherwise, the signal comes from an enemy submarine which is threatening submarine DM_1. In this case, DM_1 should also over-react but at a higher level than previously. This response is R_4. The probability of having such an input is 27/64.

Table 1 summarizes these possibilities. The partitioning of the input is done according to the following rule:

- the submarine, DM_1, receives the first three bits $a_i b_i c_i$.
- the surface ship, DM_2, receives the last three bits $d_i e_i f_i$.

The decisionmaking process takes place on the basis of this partitioning The Table 2 presents the cost matrix that gives the costs associated with the discrepancies between the ideal responses and the actual responses provided by the organization.

TABLE 1 Organizational Responses

input	response
$(a_i, d_i) = (0, 0)$	R_1
$(a_i, d_i) \neq (0, 0)$ $(b_i, e_i) = (0, 0)$	R_2
$(a_i, d_i) \neq (0, 0)$ $(b_i, e_i) \neq (0, 0)$ $(c_i, f_i) = (0, 0)$	R_3
$(a_i, d_i) \neq (0, 0)$ $(b_i, e_i) \neq (0, 0)$ $(c_i, f_i) \neq (0, 0)$	R_4

TABLE 2 Cost Matrix

ideal R \ actual R	R_1	R_2	R_3	R_4
R_1	0	2	3	4
R_2	4	0	1	2
R_3	6	4	0	2
R_4	8	4	3	0

The accuracy, J, of the organization is computed as the expected value of the cost for the particular set of inputs. It is assumed that, when DM_1 and DM_2 assess the input by themselves, without querying the DSS, they produce correctly the first two bits of the strings of three bits. That is, for an input $x_i = a_i b_i c_i d_i e_i f_i$, the result of the SA of DM_1 is $a_i b_i u_i$ where u_i is the value of the third bit that DM_1 produces. It is assumed that this value is equal to c_i with probability 1/2. In the same way, the result of the SA of DM_2 is $d_i e_i v_i$ where v_i is the value of the sixth bit that DM_2 produces. this value is equal to f_i with probability 1/2.

It is furhter assumed that the decisionmakers query the DSS only during their Situation Assessment stages. Figures 8 and 9 provide the models of the organization aided by the DSS. In Figure 8, the model of DSS is aggregated, in Figure 9 the whole model is shown.

Only three of the five alternatives a DM has to perform his processing in any internal stage where he can use the DSS are considered:

(i) DM_i does not access the DSS.

(ii) DM_i queries the intelligent terminal and relies on its response.

(iii) DM_i queries the DSS and compares its response to his own assessment.

In the remainder of this section, the following notation will hold.

- SA_i represents alternative (i).
- IT_i represents alternative (ii).
- MF_i represents alternative (iii).

This model shows that multiple flow paths can be used to process the information. Since each DM has three alternatives with respect to the use of the DSS, there are nine pure organizational strategies:

- (SA_1, SA_2)
- (SA_1, IT_2)
- (SA_1, MF_2)
- (IT_1, SA_2)
- (IT_1, IT_2)
- (IT_1, MF_2)
- (MF_1, SA_2)
- (MF_1, IT_2)
- (MF_1, MF_2)

A mixed strategy $\delta_i(p_i^1, p_i^2, p_i^3)$ for DM_i corresponds to a convex combination of his three pure strategies SA_i, IT_i and MF_i weighted by the probabilities p_i^1, p_i^2, p_i^3.

An organizational behavioral strategy is the combination of the mixed strategies of DM_1 and DM_2. Therefore, it corresponds to

$$(\delta_1(p_1^1, p_1^2, p_1^3), \delta_2(p_2^1, p_2^2, p_2^3)).$$

It is assumed that the processing of information through the use of the DSS provides different results depending on whether the intelligent terminal or the mainframe is queried. When the intelligent terminal is accessed, the decision-makers can produce correctly the first bit of the strings of three bits. That is, for an input $x_i = a_i b_i c_i d_i e_i f_i$, the result of the SA of DM_1 for the alternative IT_1 is $a_i u_i y_i$ where u_i and y_i are the values of the second and third bits that DM_1 produces. each of these two values is equal to the actual value with probability 1/2 In the

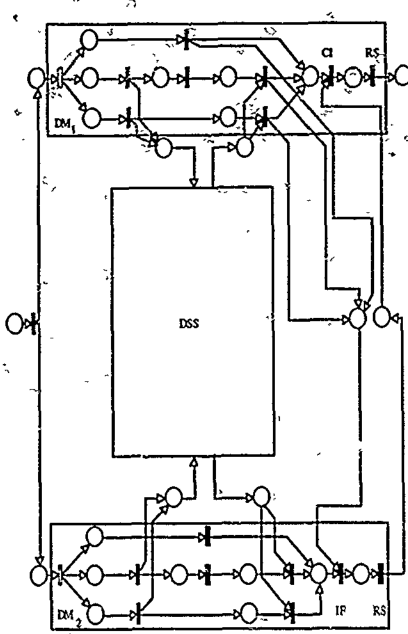


Fig. 8 Two-Person Organization Aided by DSS - Aggregated Representation

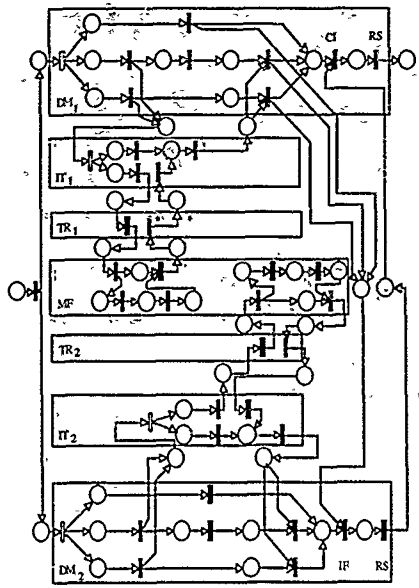


Fig. 9 Two-person Organization Aided by DSS

same way, the result of the SA of DM_2 for the alternative IT_2 is $d_iv_iz_i$ where v_i and z_i are the values of the fifth and sixth bits that DM_2 produces each of them corresponds to the actual value with probability 1/2.

When the mainframe is accessed, the decision-makers are able to produce correctly all three bits of their respective strings That is, for an input $x_i = a_ib_ic_id_ie_if_i$, the result of the SA stage of DM_1 for the alternative MF_1 is $a_ib_ic_i$. The result of the SA stage of DM_2 for the alternative MF_2 is $d_ie_if_i$. This means that, when the organizational strategy is (MF_1, MF_2), the organization will be able to produce the correct response for all inputs. For all other strategies, the responses provided may differ from the ideal response.

The access to the intelligent terminal provides, however, a means of improving the timeliness of the decision-making process. Indeed, it will be assumed that the amount of time necessary to process the information is lower when the decisionmaker uses his intelligent terminal than when he queries the mainframe or performs his processing alone.

The amount of time taken by the decisionmakers to execute the different algorithms is equal to one unit of time, except for the Situation Assessment algorithms. Different cases have been investigated in which the processing times of these algorithms differ from unity.

These considerations account for what occurs in most situations in C^3 systems. The different decisionmakers have access to different facilities which do not have the same response time or the same accuracy. On the one hand, an intelligent terminal is likely to provide faster responses because it is co-located with the decisionmaker. However, it has no centralized database which can aggregate data from multiple sensors to get a global picture of the situation and, therefore, the responses it can provide are necessarily less accurate. On the other hand, the access to the mainframe may require the communication of data from and to remote locations through a network. the response time can be quite long.

The next section contains the results obtained for different access times to the mainframe. In each case, the performance loci have been constructed for the three measures, accuracy J, expected delay T, and synchronization S_T.

Results

The results on the accuracy of the responses produced by the organization for the nine pure organizational strategies, are listed in Table 3. Accuracy is maximal when both decision-makers query the mainframe and reaches its worst level when they both query their intelligent terminal.

TABLE 3 Accuracy of the Organization

strategy	SA_1 - SA_2	SA_1 - IT_2	SA_1 - MF_2	IT_1 - SA_2	IT_1 - IT_2	IT_1 - MF_2	MF_1 - SA_2	MF_1 - IT_2	MF_1 - MF_2
J	0.42	0.76	0.21	0.76	1.01	0.69	0.21	0.69	0.00

Two cases have been investigated as far as the processing times of the Situation Assessment stages are concerned:

(i) case 1: SA_1 and SA_2 take 10 units of time. IT_1 and IT_2 take 5 units of time. MF_1 and MF_2 take 15 units of time. This case corresponds to the situation where the processing times of both decision-makers are equal when they use the same strategy with respect to the use of the DSS.

(ii) case 2 SA_1 and SA_2 take 10 units of time. IT_1 and IT_2 take 5 units of time. MF_1 takes 15 units of time but MF_2 takes 10 units of time. This corresponds to the situation where the commander has a faster access to the mainframe than the subordinate because of a better transmission time.

In both cases, when the two DMs perform their situation assessment by themselves, they take the same amount of time to do it. This means that the Information Fusion stage of DM_2 is perfectly synchronized.

Tables 4 and 5 show the results for the expected delay, T, and the synchronization, S_T, in case 1 and case 2, for the nine pure organizational strategies.

TABLE 4 Delay and Synchronization in Case 1

strategy	SA_1 SA_2	SA_1 IT_2	SA_1 MF_2	IT_1 SA_2	IT_1 IT_2	IT_1 MF_2	MF_1 SA_2	MF_1 IT_2	MF_1 MF_2
T	15	15	20	15	10	20	20	20	20
S_T	2	7	7	7	2	12	7	12	2

TABLE 5 Delay and Synchronization in Case 2

strategy	SA_1 SA_2	SA_1 IT_2	SA_1 MF_2	IT_1 SA_2	IT_1 IT_2	IT_1 MF_2	MF_1 SA_2	MF_1 IT_2	MF_1 MF_2
T	15	15	15	15	10	15	20	20	20
S_T	2	7	2	7	2	7	7	12	7

In case 1, the maximum delay is obtained when at least one of the decisionmakers accesses the mainframe. The minimum delay is reached when both decisionmakers use their intelligent terminal. When the maximum delay is reached, the synchronization can have very different values depending on the coordination of the strategies of the decisionmakers. When they both access the mainframe, the synchronization is optimal with a delay of 20 units of time. For this same delay, this synchronization can degrade considerably, if one of them accesses his intelligent terminal as the other queries the mainframe.

In case 2, the maximum delay is reached when DM_1 accesses the mainframe Nevertheless, when DM_2 queries the mainframe alone, the delay does not increase to this level. The optimal synchronization can no longer be obtained when the delay is maximal. Furthermore, the worst value for the synchronization is reached only for one pure organizational strategy, i e., for (MF_1, IT_2). This value was reached for two pure organizational strategies in case 1, i e., (MF_1, IT_2) and (IT_1, MF_2).

The interpretation of the results can be done through the consideration of the performance loci for the measures J, T and S_T. The performance loci for the two cases presented in the previous section are shown in Figures 10 and 11. They represent the values of J, T and S_T reached for each organizational strategy, pure or behavioral.

These figures show the relations and tradeoffs between the various measures of performance. It is recalled that:

- the lower the value of T, the better the delay.
- the lower the value of S_T, the better the synchronization.
- the lower the value of J, the better the accuracy.

In case 2, the part of the locus where J is the lowest, i e., where the accuracy is the best, corresponds to higher values of S_T than in case 1 This shows that there exists a trade-off between accuracy and synchronization when the DSS does not have the same response time for the two decisionmakers.

In both cases 1 and 2, when the expected delay, T, is minimal, the synchronization, S_T, is also minimal. This is due to the fact that the intelligent terminals provide the fastest way of performing Situation Assessment, and that the delay will be minimal only if both decisionmakers query their terminal. The assumption that these terminals give the responses to both decisionmakers in the same amount of time is realistic because there is no delay due to transmission and the algorithms that they use are similar. Conversely, the fact that the synchronization is minimal does not imply that the delay will be minimal. In case 1, the synchronization reaches its lowest value for all possible values of the delay. It corresponds to the fact that, for any delay, the decisionmakers can find some way to be as well synchronized as possible.

If a constraint is imposed on the delay, the synchronization of the organization does not degrade. One can notice that the more stringent the constraint on T, the more likely the synchronization will reach a good value. In case 2, the synchronization does not reach its lowest value for all values of T; as in case 1, the best values of S_T are obtained for the lowest delays.

In case 2, the more the timeliness of the organization degrades, the more the synchronization will degrade too. When DM_1 uses the mainframe, there is no way for the organization to be well synchronized. DM_2 will have to wait for long inervals of time before receiving the data that he needs in his information fusion stage.

These facts show that the introduction of a decision support system in an organization can have different effects on the coordination of the activities If the organization members are well coordinated when they do not use the DSS, the latter can

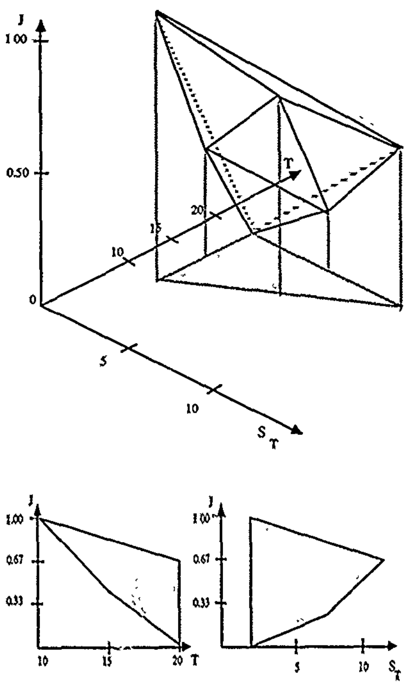


Fig. 10 Performance Locus in Case 1

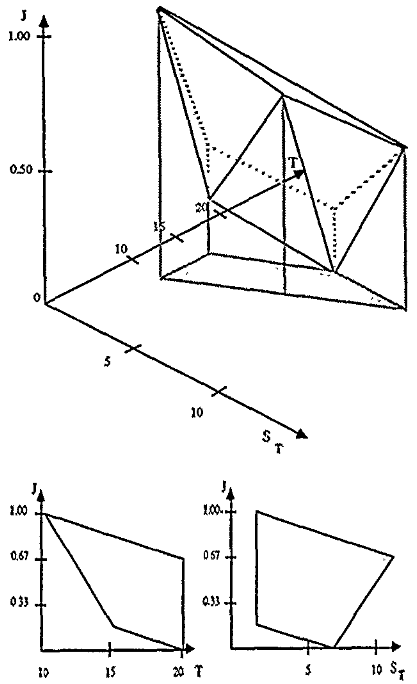


Fig. 11 Performance Locus in Case 2

degrade this coordination because of two factors:

- the decision-makers can use a larger number of facilities, it is difficult to coordinate their access.
- the quality of the responses provided by the DSS, i.e., the response time and the information produced, can be very different from one decision-maker to another.

Therefore, on the one hand, *the decision-makers have many more alternatives that they can use to perform their task; the coordination of these activities is consequently more difficult to achieve* On the other hand, in coordinating their activities, the organization members must take into account the fact that the *DSS does not perform equally well for all of them.*

This latter consideration is illustrated by case 2 of the example. there is a tradeoff between accuracy and timeliness which is coupled with a tradeoff between accuracy and synchronization. In order to achieve good accuracy, the organization members must use strategies which lead to a degradation in timeliness and synchronization. Conversely, if the decisionmakers wants to be well synchronized, the accuracy will degrade because they cannot access the mainframe together.

Therefore, the decision support system, depending on its characteristics, leads to mixed effects on the effectiveness of the organization. As in case 2, it can lead to an improvement both in accuracy and timeliness of the organization, but the coordination then degrades. Conversely, as in case 1, it cannot produce an improvement both in accuracy and timeliness, but coordination is always highest when accuracy or timeliness are optimal.

CONCLUSIONS

The concept of coordination was defined as relating to the consistency of the information exchanged by the different organization members and to the synchronization of the various activities. The latter bears directly on the dynamics of the decisionmaking process. A decisionmaking organization is perfectly synchronized for the task at hand if none of its members waits for the information that he needs at any stage of the process. If it is not the case, the value of information when it is actually processed may have decreased, leading to a degradation of the organizational effectiveness. The consistency of information shows the extent to which different pieces of information can be fused without contradiction.

The modeling of processes that require coordination has been developed using the basic model of the single interacting decisionmaker refined through the use of the Predicate Transition Net formalism. In particular, tokens representing symbolic information carriers have been differentiated on the basis of three attributes which account for characteristics that decisionmakers can use to discriminate between various data.

The protocols of interactions between organization members model the fact that they must refer to the same input when they fuse data. Different strategies for selecting the information to process have been introduced, e.g., FIFO or priority order between classes of data.

The evaluation of the coordination is based on a characterization of the firing of interactional transitions in the Predicate Transition Net model developed. Furthermore, two measures are introduced in order to perform a quantitative evaluation of the coordination of decision-making processes, i.e., information consistency and synchronization.

A methodology for assessing quantitatively the impact of a decision support system on the activities of a decision-making organization has been presented. It was used to show that the introduction of a decision support system can alter considerably the synchronization of the various activities because the capabilities offered to the various decisionmakers by the system may differ For example, a certain decisionmaker may have faster access to the central database than another one, because of different transmission times. However, the fact that some decisionmakers are provided with better capabilities can allow the organization to improve both the timeliness and the accuracy of the process.

REFERENCES

Bejjani, G.J., and A.H. Levis, 1985, "Information Storage and Access in Decision-Making Organizations," *Proc. 8th MIT/ONR Workshop on C^3 Systems*, LIDS-R-1519, Laboratory for Information and Decision Systems, MIT, Cambridge, MA.

Brams, G W., 1983, *Réseaux de Petri. Théorie et Pratique, T.2, Modélisation et Application*, Masson, Paris.

Chyen, G H-L., and A H Levis, 1985, "Analysis of Preprocessors and Decision Aids in Organizations," *Proc IFAC/IFIP/IFORS/IEA Conference on Analysis, Design and Evaluation of Man-Machine Systems*, Varese, Italy.

Genrich, H J , and K. Lautenbach, 1981, "Systems Modeling with High-Level Petri Nets," *Theoretical Computer Science*, No.13, pp.109-136.

Grevet, J.-L., L. Jandura, J Brode, and A. H. Levis, 1988, "Execution Strategies for Petri Net Simulations," *Proc 12th IMACS World Congress on Scientific Computation*, Paris, France

Keen, P G W and M.S. Scott Morton, 1978, *Decision Support Systems an Organizational Perspective*, Addison-Wesley, Reading, MA.

Levis, A. H., 1984, "Information Processing and Decision-Making Organizations a Mathematical Description," *Large Scale Systems*, No. 7, pp. 151-163.

Levis, A H., 1988, "Human Organizations as Distributed Intelligence Systems," *Proc. IFAC Symposium on Distributed Intelligence Systems,"* Pergamon Press, Oxford.

Papastavrou, J.D., 1987, "Distributed Detection with Selective Communications," MS Thesis, LIDS-TH-1563, Laboratory for Information and Decision Systems, MIT, Cambridge, MA.

Waltz, E.L., and D.M. Buede, 1986, "Data Fusion and Decision Support for Command and Control," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-16, No. 6, pp.865-879.

Weingaertner, S T., and A.H. Levis, 1987, "Evaluation of Decision Aiding in Submarine Emergency Decision-Making," *Proc. IFAC Symposium on Man-Machine Systems*, Pergamon Press, Oxford.

MODELING AND EVALUATION OF VARIABLE STRUCTURE COMMAND AND CONTROL ORGANIZATIONS*

Jean-Marc Monguillet
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

Distributed decisionmaking organizations with variable structure are those in which the interactions between the members can change, or which can process the same task with different combinations of resources. Variable structure could be a possible design solution when no fixed structure organization can meet the requirements of the mission. A modeling methodology is introduced to represent variable structure organizations that is based on the theory of Predicate Transition Nets. Decisionmaking organizations are then viewed from a new perspective in which the types of interactions which can exist between the decisionmakers are first considered without taking into account the identity of the decisionmakers themselves. The latter are represented by individual tokens (instead of subnets of a Petri Net) moving from one interaction to the other, and as such, are treated in the same manner as any other resources needed for the processing of a task. Interactions, resources, and tasks are modeled independently, i.e., the representation of the interactions, resources, and tasks is done separately in separate modules, and modifications in one module can be made without affecting the others. The methodology is illustrated by an example of a three member decisionmaking organization carrying out an air defense task.

1. INTRODUCTION

The need to meet ever increasing performance levels and to satisfy conflicting requirements has led to the investigation of organizations whose structure is variable. Variable structure organizations could be a possible design solution when no fixed structure organization can meet such requirements as robustness or survivability. The modeling of variability in the structure of organizations constitutes another step towards the representation of more realistic decisionmaking organizations.

The mathematical formulation of the modeling and analysis problem is based on the theory of Predicate Transition Nets, which is an extension of the Petri Net Theory using the language of first order predicate logic (Genrich and Lautenbach, 1981). The information processing and decisionmaking organizations that have been modeled and analyzed in earlier work (Levis, 1984; 1988) have been depicted as systems performing tasks in order to achieve a mission. These organizations are now viewed from a new perspective. The types of interactions which can exist between the decisionmakers are first considered without taking into account the identity of the decisionmakers themselves. The latter are represented by individual tokens (instead of subnets of a Petri Net) moving from one interaction to the other, and as such, are treated in the same manner as any other resources needed for the processing of a task. Interactions, resources, and tasks are modeled independently, and this new way of describing decision making organizations allows the development of a modeling methodology with a modular architecture. By modular is meant that the representation of the basic components of the information processing (interactions, resources, and tasks) is done in separate modules, and that modifications in one module can be made without affecting the others.

In the next section, variable structure organizations are defined, while in the following one the modeling methodology is described. A case study is presented in the fourth section, it illustrates the whole procedure through the design of a set of three candidate structures for a given mission, one of which is variable. Measures of Effectiveness are used to select the most effective candidate for a specific mission.

2. VARIABLE STRUCTURE ORGANIZATIONS

A variable structure decisionmaking organization (VDMO) is a DMO for which the topology of interactions between the elements or components can vary. Analogously, a DMO which has a constant pattern of interactions among its components, i.e., a fixed structure, is called a FDMO.

The relationships which tie the components together are defined at three different levels physical arrangements, links between components, and protocols ruling the arrangements of these links. The architecture of the organization allows the topology of interactions to vary. The way it does vary is implemented in the protocols themselves. The rules setting the interactions can be of any kind We distinguish three types of variability, each corresponding to characteristic properties that a VDMO may exhibit, an actual VDMO may very well have these properties (to some extent) together and simultaneously

* Type 1 variability. The VDMO adapts its structure of interactions to the input it processes Some patterns of interactions may be more suitable for the processing of a given input than others.

* Type 2 variability: The VDMO adapts its structure of interactions to the environment The performance of a DMO depends strongly on the characteristics of the environment as perceived by the organization. For example, an air defense organization may be optimized for some types of threats and

*This work was carried out at the MIT Laboratory for Information and Decision Systems with support provided by the Office of Naval Research under contract no. N00014-84-K-0519 (NR 649-003).

their probabilities of occurence. Now, if the adversary's doctrine changes, or the deployment of his assets changes, then the probability distribution of the occurence of the threats is modified. The organization (with the interactions set as before the changes in the environment) may not meet the mission requirements any more.

* **Type 3 variability:** The VDMO adapts its structure of interactions to the system's parameters. The performance of a system changes when assets are destroyed or become unavailable because of countermeasures such as jamming of communications.

These three different types of variability can be related to the properties of **Flexibility, Reconfigurability,** and **Survivability.** A DMO is survivable when it can achieve prescribed levels of performance under some wide range of changes either in the environment, or in the characteristics of the organization, or in the mission itself. The extent to which a DMO is survivable depends on the extent to which it is flexible, and reconfigurable. Flexibility means that the DMO may adapt to the tasks it has to process, to their relative frequency, or to its mission(s). Reconfigurability means that it can adapt to changes in its resources. Both properties overlap, and their quantitative evaluation clearly falls outside the scope of this paper.

The organizations under consideration are restricted to the class of teams of boundedly rational decisionmakers (DM's) (Boettcher and Levis, 1982). Each DM is well trained and memoryless. The Petri Net formalism has been found to be very convenient for describing the concurrent and asynchronous characteristics of the processing of information in a decisionmaking organization. The internal processing which takes place in any decisionmaker has been modeled by a subnet with four transitions and three internal places. A simplified version of this so-called four stage model is shown in Fig. 1.

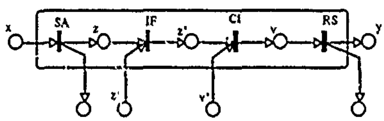


Figure 1 Four stage Petri Net model of a DM.

This model allows to differentiate among the outputs and the inputs of the decision maker, and to describe the types of interactions which can exist between two decisionmakers.

The decisionmaker receives an input signal x from the environment, from a preprocessor, from a decision-a' , or from the rest of the organization. He can receive one input to the Situation Assessment stage (or SA) at any time. He then processes this input x with a specific algorithm which matches x to a situation the decisionmaker already knows. He obtains an assessed situation z which he may share with other DM's. He may also receive at this point other signals from the rest of the organization. He combines the information with his own assessment in the Information Fusion stage (IF), which leads to the final assessment of the situation, labeled z'. The next step is the possible consideration of commands from other DM's which would result in a restriction of his set of alternatives for generating the response to the input. This is the Command Interpretation stage, or CI. The outcome of the CI stage is a command v which is used in the Response Selection stage (RS) to produce the output y - the response of the decisionmaker - which is sent to the environment or to other DMs.

As shown in Fig. 1, the decisionmakers can only receive inputs at the SA, IF, and CI stages, and send outputs from the SA and RS stages (Remy and Levis, 1987). The interactions which are the most significant are shown in Fig. 2. For the sake of clarity, however, this figure only accounts for the interactions as directed links from DM_i to DM_j. Symmetrical links from DM_j to DM_i e exist as well.

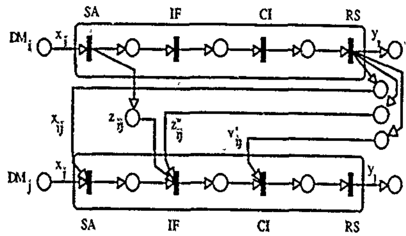


Figure 2 Allowable interactions from DM_i to DM_j.

Two kinds of places can be distinguished. **internal places,** or **memory places,** where the decisionmaker stores his own information, between SA and IF, IF and CI, or CI and RS. The places between the DM's and the sensors, the preprocessors, or the actuators, as well as those between two DM's are called **interactional places.** Knowledge of the set of interactional places is equivalent to that of the whole structure of the net.

A decisionmaker may not have all his four stages present. Depending on the interactions he has with the rest of the organization and with the environment he may exhibit different internal structures:

- SA alone.
- SA, IF, CI and RS (IF and CI can be simple algorithms that copy the signal).
- IF, CI, and RS.

Depending on what the designer of the organization requires, different constraints on the allowable interactions can be expressed, which limit or expand the set of possible organizations.

In the Petri Net representation, the transitions stand for the algorithms, the connectors for the precedence relations between these algorithms, and the tokens for their input and output. The places act like buffers, hosting the tokens until all the input places of a transition t are non empty, in which case the algorithm embodied in t can run and remove the tokens. The time taken by the algorithm to run is the transition processing time $\mu(t)$. The tokens in this model are all indistinguishable. A token in a place p means simply that a piece of information is available there for the output transition(s) of p.

In the earlier model, the SA and RS stages contained several algorithms and a switch that detrmined the choice of algorithm. The switch position was in turn determined by the decision strategies of each individual DM. The extension of the concept of a switch to model the changing interactions in a variable structure organization turned out not be useful, it introduced a set of problems :

- many switches are needed.
- the needed intercorrelation between the switches cannot be indicated on the net. A table has to be attached to it. Thus, the net representation is not complete.
- the relation between the inputs and the patterns of interactions is not shown explicitly. The high illustrative power of Petri Nets is lost since the behavior of the net can not be deduced from its representation.
- the representation becomes quite complex even for simple organizations.
- the addition of decisionmakers, of possible links, or their removal, obliges the designer to redesign the net and the attached table totally.

Attributes can be used to describe what the tokens represent. For instance, if the decisionmaker has to identify an incoming threat and to respond to it, then a token on the input place of his SA stage may be just a blip on the DM's radar screen. The token that the SA algorithm produces is in turn formatted information which includes the DM's measurement, or assessment, of the position, speed, nature, behavior, or size of the threat. The DM can receive from elsewhere in the organization other formatted information, not necessarily of the same format, provided that it matches what his IF algorithm expects as inputs formats. The different tokens in the different places have then different formats, and different attributes. But as long as the protocols ruling their processing do not vary from one set of attributes to the other, they are indistinguishable tokens.

What is needed is a tool which would allow to distinguish among the tokens, and which would have the capability to implement logic able to determine explicitly what interaction and what DM's have to be active for the processing of a given input. Individual tokens, Predicates, and Operators can meet these requirements. The application of the Predicate Transition Nets to that purpose is developed in the next section.

3. MODELING METHODOLOGY FOR VARIABLE DMO'S

In this section, a step-by-step procedure for the modeling of VDMO's using Predicate Transition Nets is developed. An example of a three member organization with type 1 variability illustrates the methodology. The methodology has a modular architecture (Fig. 3):

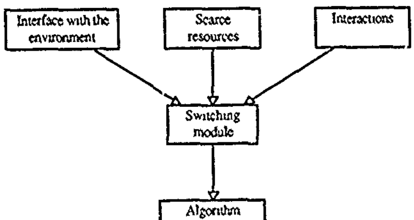


Figure 3. Architecture of the modeling methodology.

There are five modules,

1. Interface with the environment.
2. Scarce resources.
3. Interactions.
4. Switching module.
5. Algorithm implementation.

Each of the first three modules can be executed independently and in arbitrary order. The three modules address the sub-problems (a) of modeling of the inputs that the DMO receives and the responses that it gives, (b) of representing the scarce resources that the DMO needs, and (c) of modeling the possible interactions which can exist between the components.

When the first three modules have been completed, the switching module is executed. The switching module is the part of the model where the logic, which controls the variability of the organization, is implemented. For each incoming input, this is the part of the model which decides what particular resources, and what particular set of interactions will be adopted. The way this choice is made will determine what type of variability the VDMO exhibits.

What is obtained at this point is a Predicate Transition Net where only the non trivial operators are indicated in the corresponding transitions. The fifth and last part of the methodology consists of the rigorous labeling of the nodes, connectors, and tokens of the net. It also gives precise meaning to what the individual tokens stand for (i.e., the list of their attributes), depending on the places which host them, and on what algorithm, or what set of algorithms, a particular transition models. The processing time of the different algorithms is also specified. The steps of that methodology are independent enough to allow changes in any subproblem, without threatening the functioning of the whole model. The modular architecture is also very convenient for the implementation of extensions of the model, which simply become new modules, or new well-defined subproblems.

This section focuses on the modeling of type 1 variable DMO's. An example of a three member organization with type 1 variability serves to illustrate the methodology. Examples of VDMO's exhibiting type 2 or type 3 variability are included in Monguillet (1988).

Interface with the Environment

The goal of this sub-problem is to achieve a representation of the input and output alphabets. In the modeling of decisionmaking organizations, the discrete representation of information sets is done in the form of lists of attributes, an instance of which is called a token. In the ordinary Petri Net representation of a DMO, the values of the attributes were of no importance, no matter what these values were, the treatment of the token was the same: the interactions between the components were the same.

In the case of type 1 VDMOs, the alphabet X of inputs is partitioned in r classes, namely X_i, for $i = 1,..,r$. All inputs x belonging to the same class are processed with the same resources used with the same pattern of interactions. A given input x cannot belong to more than one class, which implies that it can only be processed with one specific set of resources, and one specific kind of interactions. The identity of a token is the class X_i to which it belongs; it is denoted by the index number i. The variable "class of inputs" is denoted by x and has the following set of allowable identities:

$\underline{x} = \{1, \ldots, r\}$.

Since the environment is not modeled, the tokens which model the outputs of the organization need not have an identity. They are instances of the 0-ary variable ϵ.

Example: Step 1

The example consists of a three member organization with four possible interactions between the decisionmakers. The DMO consists of two field units, FU1 and FU2, and one headquarters HQ. The possible interactions are the following.

Int#1- FU1 and HQ (HQ fuses its assessment with FU1's, and issues a command to him).

Int#2- FU2 and HQ (HQ fuses its assessment with FU2's, and issues a command to him).

Int#3- FU1 alone (SA and RS stages).

Int#4- FU2 alone (SA and RS stages).

The alphabet of inputs X is therefore partitioned in four classes X_i, $i = 1, \ldots, 4$. The variable x representing the class of the inputs has a set of identities {1, 2, 3, 4}. The outputs are not partitioned. The model of the organization which is obtained at this point is shown in Fig. 4.

Figure 4 Example - Step 1.

Scarce Resources

Resource is a generic name which designates elements needed for the processing of a task. A resource is scarce when it cannot be allocated freely to the processing of any incoming input because of insufficient or limited supply. The scarcity of resources bounds from above the performance of the organization. Scarce resources are modeled in a convenient way in the Petri Net formalism. They are represented by places with multiple input transitions and multiple output transitions, and non-zero initial marking. Examples of scarce resources can be common databases with limited access, communication links with limited capacity, mainframes with shared processing time, or weapons platforms capable of handling a limited number of threats at a time.

In this modeling methodology, the decisionmakers are treated as scarce resources: they are assigned to an incoming input; once they have been assigned to a certain number of inputs, the other inputs have to wait in line to be processed. The pool of decisionmakers which implements the organization is partitioned in classes of DM's who have the same function within the organization, i.e., who possess the same kind of algorithms. Two decisionmakers who belong to the same class are then interchangeable. The DM's of a class are represented by individual tokens of a variable, and placed in the corresponding resource place. If there is only one class of DM's, then the DM's are represented by indistinguishable tokens. The other resources that the organization may need are partitioned and associated with variables and places in the same way.

Connectors are labeled with a formal sum of variables, which indicates the kinds of tokens they can carry. The input and output connectors of a given resource place R where the corresponding variable is x are labeled by elements of $L^+(x)$, the set of all applications from $\underline{x}$ to the non-negative integers.

Example: Step 2

In the example, two DM's are interchangeable as far as their interactions with the rest of the organization are concerned: these are the field units FU1 and FU2. HQ has a unique function in the DMO, and is the only one in that case. The three DM's are then represented by the following variables:

- Resource place FU: associated with the variable $\underline{s} = \{1, 2\}$. The individual token 1 models the decisionmaker FU1. The token 2 stands for FU2.

- Resource place HQ: since there is only one HQ, the place carries an indistinguishable token ϵ, shown as a dot in the place HQ.

The modeling of the DMO at this point is shown in Fig. 5.

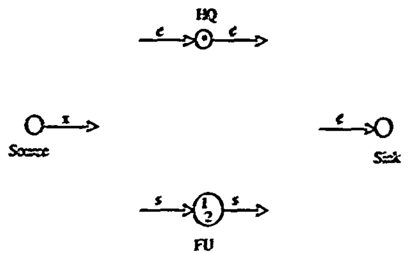


Figure 5 Example - Step 2.

Interactions

The allowable interactions between components are represented without considering the identity of the resources they involve. What is of interest, at this point in the modeling, is only the topology of interactions that can be found in the DMO. The typical model obtained at this point is shown in Fig. 6; it is a list of the possible patterns of interactions depicted in their most aggregated form. Had these interactions been considered alone as DMO's with fixed structure, the input and output places would have been the source and sink places.

The possible interactions can be partitioned in four generic types, as illustrated in Fig. 6:

- *Type (a)*: the pattern of interactions is that of an organization with a fixed structure which processes the inputs without resources. It is represented by an ordinary Petri Net which can be aggregated in a super-node Int#1.

- *Type (b)*. the pattern of interactions has the same characteristics as in type (a), but the net which models that pattern exhibits some properties of symmetry. A more convenient representation is obtained by folding the net The Predicate Transition Net which is obtained is aggregated in turn in a super-node Int#2.

- *Type (c)*: the pattern of interactions is the same as type (a), but the DMO with that pattern requires a resource R_1 for the processing of the inputs. This resource is used from the beginning of the processing until its completion. The ordinary Petri Net which models that pattern is therefore aggregated in a super-node and the resource place R_1 is both an input and an output places of that macro-transition Int#3 (the underlying Petri Net is still pure, however).

- Type (d): the pattern of interactions is similar as in type (c), except that resource R_2 is not used during the processing of the inputs. In the particular case of Fig. 6(d), it is only needed at the beginning. The ordinary Petri Net modeling that pattern is then aggregated in two super-nodes, (Int#4.1) and (Int#4.2).The former stands for the part of the processing that uses resource R_2, while the latter accounts for the remaining processing.

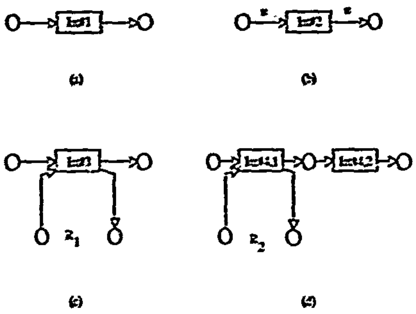


Figure 6 Allowable interactions.

Any other combination of type (a), (b), (c), or (d) can be encountered as well. In particular, the number and diversity of resources required and the lack of symmetry of the pattern of interactions may make aggregation in super-nodes inappropriate. In that case, the net which would appear in Fig. 6 would show in detail all the stages of the decisionmaking process.

No matter where the resource places are connected, the subnet which is subsumed in a macro-transition represents a decisionmaking organization where the internal processing of the input is modeled by the four stage representation that was described in Figure 1. That net stands, therefore, for an organization with fixed structure, which is to say, that it may contain some switches, but the setting of these switches does not affect the structure of the interactions between the decisionmakers (whose identity is not defined). If each switch is aggregated in a macro transition, then the ordinary Petri Nets which are obtained are all marked graphs, i.e., a place can have only one input transition, and only one output transition.

Example: Step 3

In the example, only two patterns of interactions are actually distinct: one where the HQ interacts with a FU, and one where the FU processes the task alone. The first part of the modeling consist of representing these patterns in detail (Fig. 7). Then an aggregated model comparable to Fig. 6 can eventually be produced.

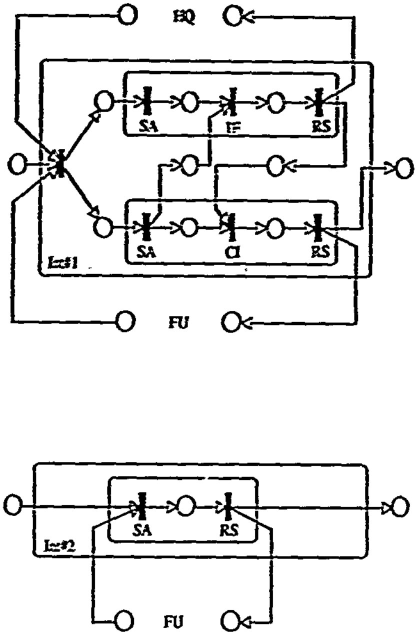


Figure 7 Example - Step 3.

For the first pattern of interactions, two resources are required, namely HQ and FU. The resource HQ is not used in the decision process until a response is chosen, and can be free before that. The resource FU, however, is needed from the beginning of the processing to the end. Finally, this pattern of interactions is such that no aggregation in super-nodes is possible. For the second pattern, the only resource used is FU, and it is needed during the whole processing of the input.

Switching Module

The objective of this module is the representation of the decision rule which determines, for any incoming input, what the actual configuration of the organization will be. The switching module is the part where the type of variability of the organization will be modeled. It supposes that the first three sub-problems have been already completed.

A switch is implemented as an output node of the source and the resource places. This switch consists of a set of transitions with operators, whose arguments are the individual tokens in the source and resource places. Recall that a DMO with type I variability is being modeled, and that it has been assumed that each class of inputs has associated only one possible pattern of interactions. Thus, if the number of classes of inputs is r, there are at most r branches in that switch.

A decisionmaking organization needs an interaction and some resources to process an incoming input. The type 1 variable DMO which has been considered so far adopts, for each class of inputs, a specific interaction and set of resources. The formal notation for the inputs, resources, interactions, and their relations is the following:

Inputs

- An input is an individual token of variable x.
- The source place SO is associated with variable x.
- The set of allowable identities for x is $\underline{x} = \{1, \ldots, r\}$.
- An input of variable x belongs to the class X_i, where x = i.

Resources

- The resources places are R_k for $k = 1, \ldots, K$.
- The resource place R_k is associated with the variable s_k.
- The set of allowable identities for s_k is $\underline{s}_k = \{1, \ldots, S_k\}$

Interactions

- The patterns of interactions are Int#(γ), for $\gamma = 1, \ldots, \Gamma$.
- There are J transitions t_j in the switch.
- t_j is associated with the Operator Op_j.
- t_j is associated with the pattern of interactions #$\phi(j)$, i.e., Int#($\phi(j)$).

Relations

- The input x requires a pattern of interactions #$\gamma(x)$, i.e., Int#($\gamma(x)$).
- The input x requires some resources from R_k, which are: $res(k, x) = \{s_{k,n}(x) \mid n = 1, \ldots, N(x)\}$.
- $\gamma(x)$ and res(k, x) for any k are functions of x.
- $\phi(j)$ is a function of j; ϕ is attached to the switch.

An incoming input, modeled as an instance of an individual token x, belongs to the class X_i. The organization is type 1 variable, and it adapts the pattern of its interactions to the class of the incoming input. The processing of the input x requires a specific pattern of interactions, namely Int#($\gamma(x)$). Since the same interactions can be adopted for different classes of inputs, the function γ is not bijective, and the number Γ of interactions is necessary smaller than the number r of classes of inputs. The processing of this individual token x also needs some resources of type R_k, given by the set of individual tokens res(k, x). The transition of the switch which corresponds to the pattern of interactions Int#($\gamma(x)$) is the transition t_j such that $\phi(j) = \gamma(x)$: there is only one j such that this relation is verified, which is denoted as $\phi^{-1}(\gamma(x))$.

If all the conditions stated above are fulfilled, then the input x is processed i.e., for $\phi(j) = \gamma(x)$, the transition t_j is enabled and fires. The operator Op_j associated with t_j expresses in logical terms the above conditions, and can be written as follows:

$$(\exists x \in SO) \wedge (\gamma(x) = \phi(j)) \wedge (\exists\, res(k, x), R_k \supseteq res(k, x)) \quad (1)$$

Since the transition t_j corresponds to Int#($\phi(j)$), and since this pattern of interactions may be needed for more than one class of input, the actual operator associated with t_j is the logical OR ($\vee$) of the operators (1) for the inputs x such that $\gamma(x) = \phi(j)$, i.e., for all the inputs x in the set $\gamma^{-1}(\phi(j)) = \{x \mid \gamma(x) = \phi(j)\}$. The operator Op_j associated with t_j is finally the following.

$$Op_j = \bigvee [(\exists x \in SO) \wedge (\exists\, res(k, x), R_k \supseteq res(k, x))] \quad (2)$$

$$x \in \gamma^{-1}(\phi(j))$$

The operators $\{Op_j\}_{j=1,\ldots,J}$ which are attached to the transitions t_j - the branches of the switch - are such that the following conflict resolution rule is verified: for any input x in the place SO, there is one and at most one transition in the set $\{t_j\}$ which is enabled, the one with the number $j = \phi^{-1}(\gamma(x))$. There is, therefore, no conflict and as soon as the required resources res(k, x) are available, t_j can fire.

The connectors from the place R_k to transition t_j are labeled by the set $L_{conn}(R_k, t_j)$ whose elements are the symbolic sums of the individual tokens in res(k, x). If the set res(k, x) is non empty, the connector from R_k to t_j has the following label:

$$L_{conn}(R_k, t_j) =$$

$$\left\{ \lambda \in L^*(s_k) \mid \lambda = \sum_{n=1}^{N(x)} s_{k,n}(x) \text{ and } \gamma(x) = \phi(j) \right\} \quad (3)$$

Example: Step 4

In the example, the switching module contains two transitions t_1 and t_2. Therefore,

Inputs: $\underline{x} = \{1, 2, 3, 4\}$.

Resources:

R_1 = HQ, associated to the 0-ary variable ¢.
R_2 = FU, associated to the variable s, with $\underline{s} = \{1, 2\}$.

Interactions:

Int#1, corresponding to transition t_1.
Int#2, corresponding to transition t_2.

Relations. For any input x, the pattern of interactions Int#($\gamma(x)$) is:

$\gamma(1) = 1$
$\gamma(2) = 1$
$\gamma(3) = 2$
$\gamma(4) = 2$

For any input x, the required resources are:

res(1, 1) = res(1, 2) = {1¢}
res(1, 3) = res(1, 4) = Ø
res(2, 1) = {1}
res(2, 2) = {2}
res(2, 3) = {1}
res(2, 4) = {2}

The operators Op_1 and Op_2 can then be written (without mentioning the quantifiers) as follows.

Op_1: $[(x = 1) \wedge (s = 1)] \vee [(x = 2) \wedge (s = 2)]$.
Op_2: $[(x = 3) \wedge (s = 1)] \vee [(x = 4) \wedge (s = 2)]$.

The operators can actually be aggregated into a more convenient form:

Op_1: $[[(x = 1) \vee (x = 2)] \wedge (s = x)]$.
Op_2: $[[(x = 3) \vee (x = 4)] \wedge (s = x-2)]$.

In the net obtained up to this point the patterns of interactions, the resources, the source, the sink, and the transitions of the switch are connected together, and the transitions show the operators assigned to them. The patterns of interactions, however, are still in their most aggregated form, and the connectors are not all labeled (Fig. 8). This net is not yet fully defined. The purpose of the next module will be precisely to make this net functional by completing its annotation.

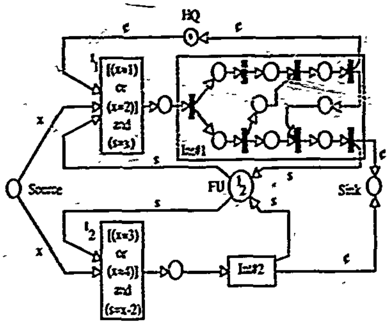


Figure 8 Example - Step 4.

Algorithm Implementation

This fifth module of the methodology deals with the labeling of the connectors, with the definition of the attributes of the tokens which can be found at different places, and with the algorithm that the various transitions represent. The rules of firing must also be established.

Labeling of connectors: The connectors from the source to the transitions of the switch are labeled x, i.e., with the variable designating the class of the inputs. Those from the input nodes of the sink to the sink itself are labeled ¢. The labels of the output connectors of the resource place R_k have already been given in Eq. (3). The input connectors of R_k are labeled accordingly.

Each pattern of interactions Int#(γ) is adopted whenever the incoming class of input x is such that $\gamma(x) = \gamma$. When $\underline{x}$ describes the set of classes of inputs $\underline{X}$, the number of times Int#(γ) is activated is equal to the number of times $\gamma(x) = \gamma$. The connectors which are involved in the representation of the organization with a pattern of interaction Int#(γ) can then be labeled with a variable ρ_j whose set of allowable identities is:

$$r_j = \{1, 2, ..., \gamma\}.$$

These labeling rules are the most general that can be presented, and can be applied to any case.

Firing rules: The firing rules are actually problem dependent, and can be revised at any time. However, they are generally the following:

- the transitions which constitute the switch are enabled and fire consecutively, i.e., with one input at a time.

- the transitions which are part of the subnets representing the possible interactions with ordinary Petri Nets are enabled and fire in the same consecutive manner. In other words, if a given place in one of these subnets contains more than one token, its only output transition ("only" because the subnet is an event graph) is enabled by more than one token. But it will fire them only one by one.

- the transitions which are part of the subnets representing the possible interactions with Predicate Transition Nets, i.e., when the original Petri Net has been folded, can allow simultaneous firing; depending on the circumstances, two tokens in the same place can enable the same transition at the same time and leave simultaneously the same place.

Depending on the identity of the individual token of variable ρ_j which enables it, a particular algorithm, or a particular switch, is activated, and processes the input that the token represents. Depending also on the organization that the net models, this transition can very well consist of only one algorithm, which is always activated and executed when the transition is enabled and fires, regardless of the identity of the individual token which has triggered that process. The rule that selects the algorithm which will process the token that enabled the transition is problem dependent, and as such, defined for each particular case.

Example: Step 5

The final representation of the example is given in Fig. 9. Since the organization is fairly simple, a simplified and self-explanatory labeling has been adopted.

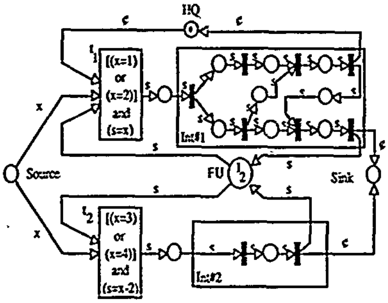


Figure 9 Example - Step 5.

4. EFFECTIVENESS OF A TYPE 1 VARIABLE DMO

In the previous section, a methodology for the modeling of VDMO's was presented, and it was assumed that the inputs were partitioned in classes, corresponding to specific patterns of interactions, before being processed by the organization. An example of a three member variable structure organization for which this assumption is relaxed is considered in this section.

The organization and its model

We consider an organization composed of three decisionmaking units, the Headquarters (HQ) and two Field Units (FU1 and FU2). Its mission is the defense of a given area against aerial threats, aircraft or missiles. Each incoming threat is identified by HQ, and its location determined by both Field Units. HQ communicates then the identity of the threat to the FU's who decide to fire or not to fire, depending on that information

DMO's with a fixed structure: FDMO1 and FDMO2

Different settings for the interactions between the DM's are possible. In the first case (FDMO1), the HQ and the FU's receive simultaneously the input and HQ sends its information on the identity of the threat to each of the FU's at the same time. They each fuse their assessement of the situation with that information, and give a response to the threat in a simultaneous way. In the second case (FDMO2), only FU1 receives information from HQ, which he fuses with his own assessment of the situation and sends to FU2. FU2 fuses in turn this information with his own assessment and produces the final response of the organization (Figs. 10 and 11).

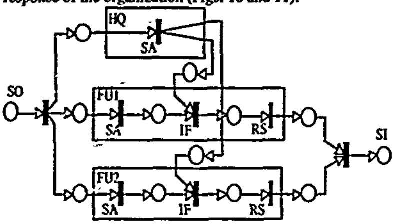


Figure 10 - Candidate #1: FDMO1.

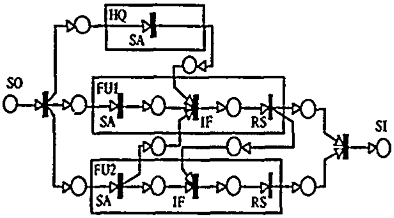


Figure 11 Candidate #2: FDMO2.

Type 1 VDMO

In general terms, it is legitimate to suspect that FDMO1 would take less time to respond than FDMO2, since the two Field Units have parallel activities in the first case, but have to interact in the second. However, the same reason may result in the response of FDMO2 being more accurate than the one of FDMO1.

An organization in which the three decisionmakers would concurrently and simultaneously assess the situation, and in which Headquarters would decide the type of interactions to be adopted between the FU's for their final processing, is likely to perform better; i e., lower processing delay and higher accuracy. The organization which would be obtained that way would be type-1 variable, and the Headquarters in that case would play the role of a preprocessor. The inputs arrive and are indistinguishable, then the HQ attaches to each of them an attribute, or class, which determines the type of interactions that are best suited for their processing. There are, therefore, three candidates for that air defense mission, two organizations with a fixed structure (FDMO1 and FDMO2), and a variable structure organization (VDMO).

PrTN model of the VDMO

The variable organization is modeled with a Predicate Transition Net using the methodology developed in the previous section. The Situation Assessment stage of the HQ acts as a source of information and associates an attribute u to the incoming token. What results is an hybrid representation, using the formalisms of both ordinary Petri Nets and Predicate Transition Nets. The VDMO is shown in Fig. 12. The variable controling the variability is called u, whose set of allowable values is {0,1}. The Situation Assessment stages of the Field Units are modeled with the conventional representation. After an input has been processed in these stages, the FU's are modeled with individual tokens of a variable x. The set of allowable values for x is {1, 2}, with token 1 (resp. 2) standing for FU1 (resp. FU2).

The Inputs

The three decisionmakers are geographically dispersed. They communicate with the help of wired links or radio. The threats are characterized by their radial distance, i e., they are modeled as occurences on a line. Their position on this line is measured by a variable x, x ∈ [0, 3]. They appear one at a time and they are independent. The line is divided in three sectors, namely [0,1],]1,2[, and [2,3]. Since the Field Units are placed close to the extreme sectors, they perform the same algorithm that determines the position of the target on the line, but with different accuracy, depending on the sector in which the target appears. For instance, FU1 is accurate when a threat appears in [0,1], less accurate when it appears in]1,2[, and even less when in [2,3]. The accuracy of FU2 is analogous to FU1's.

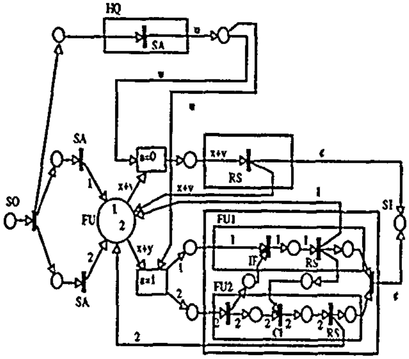


Figure 12 Candidate #3 VDMO.

The inputs are instances of elements x of an alphabet X. A given instance is modeled by the pair x = (z, Name), where z is a real in [0,3] and Name is a string in {00, 10, 01, 11}. The name of the input represents the identity of the threats. They can be thought as being types of aircraft, or types of behavior. The threats whose Name is 00, 01, or 10 represent Foes, and have to be destroyed. Only 11 is Friend.

The position of the threat on the line is denoted by z. This is the actual position, but the Field Units, who are in charge of determining it, only achieve their own measure [z] of z. In other words, each of them has an interval (of uncertainty) for the value of z. The accuracy of their measure decreases with remoteness. In order to keep the computations simple, the position z in [0, 3] is discretized such that only 30 different positions are allowed, namely 1, 2, ..., 30. Any input which appears actually in $[0.1*(i-1), 0.1*(i)[$ is called z_i, where i is an integer between 1 and 30. For completeness, the last interval is [2.9, 3.0]. Consequently, the alphabet X consists of elements $x = (z_i, Name_j)$, with:

$z_i \in \{1, 2, .., 30\}$
$Name_j \in \{00, 01, 10, 11\}$, for j = 1, ..., 4.

Strategies of the DM's and Cost Matrix

For any incoming input x_i, the Field Units determine the position of the threat, and the Headquarters identifies its Name.

Situation Assessement

Each FU's has the same set of two algorithms in the SA stage, called SA1(FU) and SA2(FU). SA1(FU) is more accurate than SA2(FU), and, as a result, takes more time to produce a response. Each algorithm yields a measure of the position of an input x_j with precision δ represented by an integer. A precision of 1 means that there is no uncertainty in the knowledge of z_i, and that the measure of its position $[z_i]$ is equal to z_i. The interval of uncertainty is reduced to $\{z_i\}$. A precision of 3 means that the measure $[z_i]$ can be at any one of hree different positions: $\{z_i - 1, z_i, z_i + 1\}$.

The algorithms used in the Situation Assessment of the Field Units are characterized by the precision they can achieve. In this model, precision is taken as a function of the sector to which the threat belongs. the precision δ is supposed to be a linear function of the remoteness, at least in this range of positions of the threat.

- Algorithm SA1(FU) for FU1 :

$$1 \le i \le 10 \Rightarrow \delta = 1$$
$$11 \le i \le 20 \Rightarrow \delta = 3$$
$$21 \le i \le 30 \Rightarrow \delta = 5$$

- Algorithm SA2(FU) for FU1:

$$1 \le i \le 10 \Rightarrow \delta = 3$$
$$11 \le i \le 20 \Rightarrow \delta = 5$$
$$21 \le i \le 30 \Rightarrow \delta = 10$$

The precision of measurements for FU2 are deduced from the above by setting $i' \rightarrow (30 - i)$. The values of δ are quantized so that they are the same wherever the threat appears in a given sector. Their dependence on the distance has been set to account for a rapid decrease in accuracy when the distance increases. The delay of the second algorithm has been set arbitrarily at one unit of time. At this point, we assume that if one obtains a measurement with precision δ but spends T units of time in that operation, then one will require more than 2T units of time to obtain a precision $\delta/2$. Since the first algorithm is twice as accurate as the second one, the processing delay of the first one is set to three units of time.

The Headquarters possesses a set of two algorithms in its SA stage. The first one, SA1(HQ), identifies the name of the threat by reading the two characters of the string. In that case, the threat is completely identified. The second algorithm, SA2(HQ), only reads the first character of the string and is less accurate than the first one. The same argument as above leads to a processing delay of two units of time for SA2(HQ) and four units of time for SA1(HQ).

Internal Strategies

The set of alternative algorithms that the decisionmakers possess leads to the definition of their *internal strategies*. The variables u_1, u_2, and u_3 are first defined to have their set of values equal to {1, 2}, and to correspond to the settings of the switch of the situation assessement stage of FU1, FU2, and HQ, respectively. The variable u_1 for instance is set to:

$u_1 = 1$ if FU1 processes its input with the algorithm SA1.

$u_1 = 2$ if FU1 processes its input with the algorithm SA2.

The variables u_2 and u_3 are determined accordingly. Now the internal strategy of FU1, D(FU1), is the probability distribution of the variable u_1, as indicated in the following:

$$D(FU1) = p(u_1) = \{p(u_1 = 1), p(u_1 = 2)\}.$$
$$D(FU2) = p(u_2) = \{p(u_2 = 1), p(u_2 = 2)\}.$$
$$D(HQ) = p(u_3) = \{p(u_3 = 1), p(u_3 = 2)\}.$$

A decisionmaker uses a *Pure Strategy* when he always processes the incoming input with the same algorithm Otherwise, he uses a *Mixed Strategy*. In the present case, each DM possesses two pure internal strategies.

Information Fusion stages

The time delay of the Information Fusion stages is a function of the number of inputs to be fused. If two inputs have to be fused, the processing delay is one unit of time If three inputs have to be fused, the delay will be two units of time. All other algorithms have associated a delay of one.

When the two Field Units fuse their measurements of the position of the threats, precision is increased, if these measurements are consistent. If two measurements of the same input with precision δ_1 and δ_2 are fused into a measurement with precision $\delta = Fus(\delta_1, \delta_2)$, then the results are as follows:

TABLE 1 Precision of Fused Information.

Fus(1, -)	= 1	Fus(5, 5)	= 3
Fus(3, 5)	= 2	Fus(5, 10)	= 5
Fus(3, 5)	= 2	Fus(10, 10)	= 10
Fus(3, 10)	= 3		

Response Selection Stage and Cost Matrix

The decisionmaker in each Field Unit can either allocate a missile to the target, or do nothing. If he sends a missile to the position where he has measured the threat to be located, then he can either hit the target or miss it, depending on the accuracy of his measure. The FU's response is denoted by y, the target coordinates. y can take the values x, if the missile is sent exactly where the target is,]x, if a missile is sent to a wrong position, and † if no missile is sent.

The ideal response for a Friend (Name 11) is to do nothing, whereas the ideal one for a Foe is to destroy it There is, furthermore, a penalty for an over-consumption of missiles The cost associated with any discrepancy between the ideal and the actual responses is indicated in the following cost matrix

TABLE 2 Cost Matrix.

x_1	x			$\neg x$			†		
x_2	x	$\neg x$	†	x	$\neg x$	†	x	$\neg x$	†
Foe: Y = D	1	1	0	0	6	6	0	6	6
Friend Y = ND	3	3	1	3	2	1	3	1	0

In that matrix, the left column corresponds to the ideal response of the organization. The top row labeled x_1 indicates the response of FU1, whereas the one labeled x_2 represents the response of FU2. The costs are adjusted to reflect subjectively the ranking of the actual responses of the organization. For example, the ideal response for a Friend input is for the Field Units to take no action, i.e., x_1 and x_2 to be inactive (†). If one missile is targetted to the wrong coordinates, in other words if x_1 = †, and $x_2 = \neg x$, (or the reverse), then the cost of wasting one missile is estimated to be one. The cost of targetting accurately a Friend is three. These values can be modified to account for any other set of beliefs.

The probability distribution of the occurences of the inputs is assumed to be uniform, unless otherwise specified. The probability for the input x of the alphabet X of having its Name equal to a given $Name_j$ is then 1/4, whereas the probability that this input has a position equal to a specific z_i is 1/30. We have then:

$$p(x = (z_i, Name_j)) = 1/120,$$
for all z_i in {1,..., 30} and all $Name_j$ in {00, 01, 10, 11}

Measures of Performance

Measures of Performance (MOP's) are quantities which describe the system properties. The MOP's are functions of the system parameters and of the organizational strategy adopted by the organization. The two MOP's considered here are *Accuracy* and *Timeliness*.

Accuracy, denoted by J, is a measure of the degree to which the actual response of the organization to a given input matches the ideal response for that same input. If we denote by

- X the alphabet of inputs x_i: $X = \{x_1, x_2, ..., x_n\}$,
- Y the alphabet of outputs y_j: $Y = \{y_1, y_2, .., y_q\}$,
- $p(x_i)$ the probability of occurence of the input x_i, with $\sum p(x_i) = 1$,
- $y_d(x_i)$ the ideal (or desired) response to x_i,
- $y_{aj}(x_i)$, $j = 1,...,q$, the response that the DMO actually produces,
- $C(y_d, y_a)$ the cost of the discrepancy between the ideal and the actual responses,

then a measure of Accuracy of the DMO is:

$$J = \sum_{i=1}^{n} p(x_i) \sum_{j=1}^{p} C(y_d(x_i), y_{aj}(x_i)) \;\; p(y_{aj}(x_i) \mid x_i) \qquad (4)$$

Timeliness, denoted as T, is the ability to respond to the input with a time delay T_d which is within the allotted time $[T_{min}, T_{max}]$, called the window of opportunity. If we denote by

$T_d(x_i)$ the average processing delay of x_i,
1_Ω the characteristic function on the set Ω,

then a measure of Timeliness of the DMO is the expected value of the processing delay:

$$T = \sum_{i=1}^{n} p(x_i) \cdot T_d(x_i) \qquad (5)$$

The performance of the organization is a function of the strategy of the organization as a whole, or organizational strategy, which is given by the triplet:

$$S = \{D(FU1), D(FU2), D(HQ)\}.$$

Since the three switches which are present in the organization are in the Situation Assessement stages, the internal strategies are not formulated with probabilities conditioned by the inputs There are, therefore, eight Pure Organizational Strategies, which are the triplets of the pure internal strategies These Pure Strategies S_i, $i = 1, .., 8$, can be defined by the algorithms the DM's are using, as follows (the order is FU1, FU2, HQ).

S_1 = (SA1, SA1, SA1)
S_2 = (SA1, SA2, SA1)
S_3 = (SA2, SA1, SA1)
S_4 = (SA2, SA2, SA1)
S_5 = (SA1, SA1, SA2)
S_6 = (SA1, SA2, SA2)
S_7 = (SA2, SA1, SA2)
S_8 = (SA2, SA2, SA2)

The application of Eqs. (4) and (5) gives immediately the values for Accuracy J and Timeliness T for FDMO1 and FDMO2, for each Pure Strategy S_i. The results are shown in Table 3, with T in units of time.

The type 1 VDMO beeing considered adapts the interactions between the Field Units to the inputs that they have to process We consider the case where the inputs are distinguished on the basis of the sectors in which they have appeared. HQ is assumed to be able to determine the sectors of occurence of the threat, which the FU's either cannot do, or can do but have to wait for the HQ's command. HQ, therefore, sets the interactions between the FU's to be as in FDMO1 when the threat occurs in the extreme sectors [0, 1] and [2, 3], and as in FDMO2 when the threat is in]1, 2[. In the former case, there is no real need for the Field Units to interact since at least one of them has an accurate measurement of the position of the threat. In the latter case, however, the precision of the measurement is increased because the FU's fuse their information, and, in doing so, reduce the interval of uncertainty of their respective measurements.

When compared to FDMO1, VDMO is likely to have an improved accuracy of response when the threat appears in]1,2[. When compared to FDMO2, VDMO will have a lower response time when the threat appears in the extreme sectors. The results for Accuracy and Timeliness for the VDMO are shown in Table 3, for the eight Pure Strategies.

A behavioral organizational strategy is constructed by considering the probability distributions of choosing a particular algorithm at each switch In the present case, such a strategy is completely defined by the triplet {p(u1), p(u2), p(u3)} The resulting strategy space for the organization is the set $[0, 1]^3$ The system loci for the two organizations with a fixed structure, i.e , FDMO1 and FDMO2, are depicted in Fig 13 They are disjoint, and no matter what Organizational Strategy is used in any of the two organizations, FDMO2 needs more time to respond. As indicated in Fig. 13, the whole locus for FDMO1 is to the left of the line T = 7 units of time, whereas the one for FDMO2 is to the right of the line T = 9 units of time.

TABLE 3 Accuracy and Timeliness for the Pure Strategies.

				FDMO1		FDMO2		VDMO	
	FU1	FU2	HQ	T	J	T	J	T	J
S_1	SA1	SA1	SA1	5.00	3.5900	9.00	2.4833	6.67	3.3944
S_2	SA1	SA2	SA1	6.00	2.8167	10.00	1.9583	7.67	2.6271
S_3	SA2	SA1	SA1	6.00	2.8167	10.00	1.9583	7.67	2.6271
S_4	SA2	SA2	SA1	6.00	2.1759	10.00	1.4167	7.67	2.000
S_5	SA1	SA1	SA2	7.00	3.0500	11.00	2.4167	8.67	2.8056
S_6	SA1	SA2	SA2	7.00	2.0833	11.00	1.1250	8.67	1.7292
S_7	SA2	SA1	SA2	7.00	2.0833	11.00	1.1250	8.67	1.7292
S_8	SA2	SA2	SA2	7.00	1.3056	11.00	0.5625	8.67	1.0625

The same methodology for evaluating the MOPs applies to the organization with a variable structure, the VDMO. The system locus of VDMO is shown also in Fig. 13. As expected, the variable structure organization is, on the average, faster to respond than the fixed structure organization in which the Field Units have to interact (FDMO2), precisely because they do not always interact in VDMO. VDMO is also, on the average, more accurate than FDMO1, since the FU's in the VDMO interact as needed to improve their measurements of the position of the target.

The computation of the performance of an organization for any behavioral strategy and the representation of its system locus are not sufficient to allow the designer to select the best organization among a set of candidates. The mission the organization has to fulfill has to be taken into account. This mission is described in terms of a pair (T^0, J^0) of constraints on performance. A convenient representation of the Effectiveness of a DMO is a three dimensional locus $(T^0, J^0, E(T^0, J^0))$, called *diagram of consistency*. In such a locus, $E(T^0, J^0)$ is the percentage of strategies for which the performance of the DMO (T, J) meets the requirements of the mission ($T \le T^0$, $J \le J^0$). The effectiveness measure $E(T^0, J^0)$ takes a value between 0 and 1, with 0 corresponding to no strategy at all satisfying the mission, and 1 meaning that all admissible strategies lead to satisficing performance.

The diagrams of consistency for the three candidate organizational structures are depicted in Fig. 14 (for FDMO1), Fig. 15 (for FDMO2) and Fig. 16(for VDMO). In these diagrams, the variables X, Y and Z correspond to J, T, and E, respectively. The figures show clearly that FDMO1 has a higher effectiveness than any of the other two organizations in the region of stringent constraints on Timeliness but not on Accuracy. Conversely, FDMO2 is the most effective when the mission requires high Accuracy.

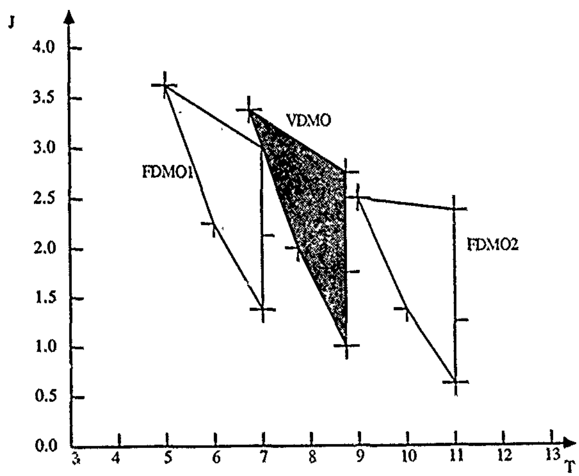


Figure 13 System Loci for VDMO.

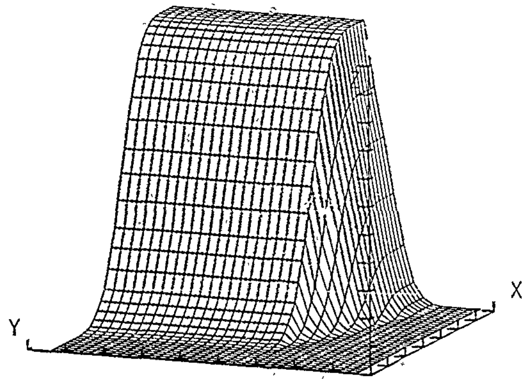


Figure 14 · Diagram of Consistency for FDMO1.

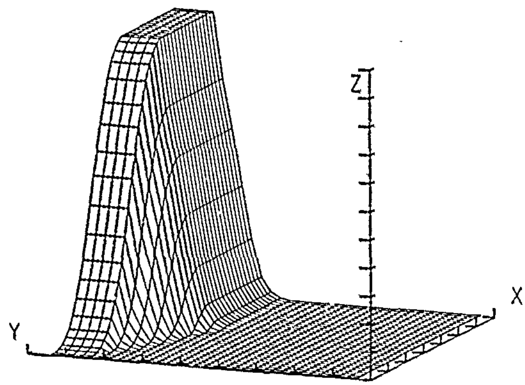


Figure 15 Diagram of Consistency for FDMO2.

Since the Effectiveness of each of the three design candidates has been computed, for any given mission defined by its requirements (T^0, J^0), the organization which has the highest effectiveness for a specific mission can be selected. More than one organization can, of course, achieve the same Effectiveness. Then each organization has associated a range of mission requirements (T^0, J^0) in the MOP space, such that for any mission requirements (T^0, J^0) within that subset, that organization will have higher effectiveness than all the other candidates. This defines a partitioning of the requirements space (T, J) in areas corresponding to each organization, or set of organizations, if the maximum effectiveness is obtained for several designs for the same mission requirements.

The computation of the measure of effectiveness E for each design candidate has been done for discrete values of T^0 and J^0. Thirty three values for the Timeliness requirement T^0, ranging from 4.00 to 12.C0, and thirty six values for the Accuracy requirement J^0, ranging from 0.50 to 4.00, have been used. This resulting grid of 33 x 36 values for the effectiveness of each candidate was then used to determine the ranges of mission requirements for which each candidate is the most effective. The precision of the determination of these ranges is of course a function of the size of the grid. This explains, for instance, the occasional piecewise linear border between zones.

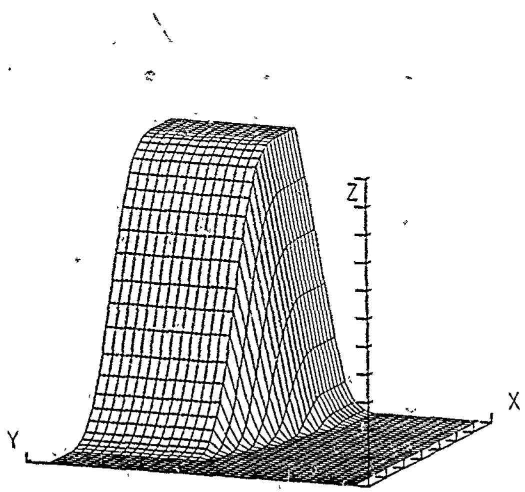


Figure 16 Diagram of consistency of VDMO.

Such a partitioning is represented in Figure 17. There are seven distinct areas. The first area, with no shading pattern, corresponds to the set of mission requirements for which all organizations have an effectiveness equal to 0, i.e., there is no organizational strategy that can meet the mission requirements. The area labeled FDMO1 is the one in which FDMO1 is the most effective; its non-zero measure of effectiveness is higher or equal to the measure of effectiveness of FDMO2. The areas labeled VDMO and FDMO2 are the ones for which VDMO and FDMO2 are most effective. In the fifth area, which is labeled FDMO1+VDMO, both organizations have an effectiveness of 1, which means that for both any organizational strategy will meet completely the requirements of the mission. There is no rationale in that case to select one organization over the other There is no region corresponding to FDMO1+FDMO2. In the region (FDMO1+FDMO2+VDMO), all three designs meet totally the requirements.

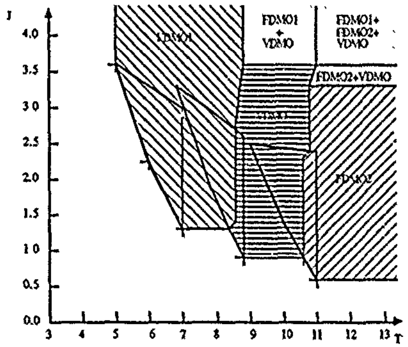


Figure 17 Partitioning of the requirements space for fixed and variable structure DMO's

5. CONCLUSIONS

In the previous sections, the need for variable structure organizations has been described and the concept of variability discussed. A methodology for modeling variable structure decisionmaking organizations that is based on Predicate Transition Nets has been presented. The approach was then used to model a variable structure organization and then analyze it with the tools that have been developed earlier for fixed structure organizations. It has been shown that one can not decide whether a VDMO performs better than an organization with a fixed structure, unless the specific mission requirements are taken into consideration. Then ranges of mission requirements have been identified for which specific organizational designs are most effective. If the requirements are such that the best design is the one with variable patterns of interactions, then the VDMO should be considered. If they are not, then there is no need to introduce variability, since a VDMO would not perform any better. A fixed organizational structure would require a simpler C^3 system to support it.

If the requirements are met both by a variable structure organization and an organization with a fixed structure, then other criteria may be used at this point, such as, for instance, the robustness of a design, which would favor a fixed structure DMO since it is less sensitive to noise or jamming. These criteria have not been addressed in this paper, but would constitute the next step toward the modeling of more realistic decisionmaking organizations.

6. REFERENCES

Boettcher, K. L. and A. H. Levis (1982). "Modeling the Interacting Decisionmaker with Bounded Rationality," *IEEE Trans. on Systems, Man and Cybernetics*, Vol. SMC-12, No.3.

Genrich, H. J. and K. Lautenbach (1981) "System Modeling with High-Level Petri Nets," *Theoretical Computer Science*, No.13, pp. 109-136.

Levis, A. H. (1984). "Information Processing and Decision-Making Organizations: a Mathematical Description," *Large Scale Systems*, No.7, pp. 155-163.

Levis, A. H., (1988). "Human Organizations as Distributed Intelligence Systems," *Proc. IFAC Symposium on Distributed Intelligence Systems*, Pergamon Press, Oxford.

Monguillet, J. M., (1988). "Modeling and Evaluation of Variable Structure Organizations," MS Thesis, Report LIDS-TH-1730, Lab. for Information and decision Systems, MIT, Cambridge, MA.

Remy, P. and A. H. Levis (1988). "On the Generation of Organizational Architectures Using Petri Nets," in *Advances in Petri Nets*, G. Rozenberg, Ed, Springer Verlag, Berlin.

ORGANIZATION THEORY AND C3

Dr. John P. Crecine
President
The Georgia Institute of Technology

Dr. Michael D. Salomone
Associate Dean
Carnegie Mellon University

Our approach to command and control assessment draws heavily on the Carnegie School of organization theory as developed in the classic works of Simon, Cyert, and March. This approach rests on a key premise: certain fundamental cognitive constraints severely limit the human capacity for rational action. Limits on the amount of information that can be attended to force people to respond to only limited aspects of their environment. And limits on the ability actively to manipulate conceptual information force people to rely on relatively simple mental strategies that frequently violate conventional notions of rational inference and choice. Organizations may be viewed as devices for overcoming these individual cognitive limits. Through specialization, individuals acquire the capacity to apply relatively complex cognitive strategies (based on very efficient information coding schemes) to narrowly defined task environments. Through division of labor, substantial cognitive resources can be simultaneously brought to bear on many tasks or information sources at a time.

Modern warfare represents the clash of two, rival organizations, each bent on containing or destroying the other. Adopting an organizational perspective for evaluating competing military organizations -- such as the NATO and Warsaw Pact organizations -- generates a different set of questions and focuses attention on different issues than the traditional forms of force balance calculations and threat assessment. In particular, the focus shifts from relative firepower, technological properties of weapons systems, and numbers of men and equipment to the organizational capabilities of military units and to the ways in which the various components of military units act in a coordinated way.

What can the complex of men and equipment do, how well, how quickly, and under what conditions? A central issue in determining the organizational capabilities of a large, military force concerns the aggregate division of labor in the force: what are the component units and what are their special capabilities? For a given division of labor, how do the component units work together and how are they coordinated so as to perform a set of functions beyond the capabilities of any single unit?

A second set of issues is the natural by product of organization structure; namely, formal organizational divisions or "seams."

Formal organizational boundaries are important in that they usually define areas where all aspects of a subunit's operations, "within boundaries," are organic to the subunit, whereas operations involving other subunits -- "across boundaries" -- involve explicit inter-unit coordination and planning. Organizational boundaries -- the "seams" in the organization -- are important to a military offense because they define what units require explicit coordination from a command point of view. Organizational seams are important to a military organization in a defensive posture because they represent points where the disruption of explicit, inter-unit communications in real time will do more to disrupt overall capabilities than the disruption of intra-unit communications. This is simply because inter-unit coordination more often involves explicit communication and coordination and intra-unit coordination more often involves tacit forms of coordination in addition to, but often in place of, explicit communication.

Coordination is the Achilles heel of the organizational strategy of overcoming individual cognitive limits through specialization and division of labor. The efforts of numerous individuals and separate groups of individuals must be structured to form coherent and useful patterns of activity. But this structuring task itself can easily become so complex as to overwhelm the cognitive capacities of those who must carry it out. Indeed, the high frequency of "coordination failures" suggests that such problems commonly occur.

The military command and control process embraces all of the information processing activities associated with monitoring the environment for problems and opportunities, formulating alternatives, deciding between alternatives, issuing mission orders and instructions and monitorng and controlling the execution of assigned missions. All of these tasks involve coordination among many parts of the command organization, both vertically within the chain of command, and horizontal beyond it into the chain of command of equivalent or parallel organizations. Adjacent army groups, corps, divisions are examples of this latter point. The additional coordination requirement for air forces to support ground forces through battlefield air interdiction or close air support missions adds another layer of complexity. Air forces within NATO not only have their own command structure, but they operate with radically different equipment, training, doctrine, requirements and ethos than do ground forces.

In adopting an organizational focus for C3I research, attention is immediately drawn to the question of what is being coordinated with what and the prominent role that the formal structure of the organization plays. An organizational approach to C3I is particularly important in two different types of situations: i) where the functioning of an organization depends crucially upon explicit coordination, communication, command and control, across significant organizational boundaries, and ii) where the functioning of a subunit of the organization is automatic or involves the execution of standard operating procedures.

THE ORGANIZATION OF NATO IN CENTRAL EUROPE

The current organizational structure of NATO evolved from political agreements about the post war structure of Europe, the formation of the Atlantic Alliance, and the sensitivities of individual member nations. While relatively stable concerning national assignments in the command structure of NATO military forces, the Alliance military structure also reflects the growing importance and military strength of the Federal Republic of Germany, and the worldwide economic and military strength of the United States. It is anything but conducive to the integration of the multi-national forces into a single, well-coordinated military force.

SACEUR (Supreme Allied Commander Europe is an U.S. general officer who is also the U.S. Commander in Chief, Europe, in peacetime) is the most prominent of three distinct major commanders in NATO. The Supreme Allied Commander Atlantic (SACLANT) and Commander-in-Chief Channel (CINCHAN) are the other two major commands. There are six separate commands on the level beneath SACEUR in the NATO hierarchy, each of these with two or three separate commands beneath it, and each filled by a military commander of a different nationality.

In the NATO Central Region, that includes the geographic territory of Belgium, the Federal Republic of Germany, Luxembourg, and the Netherlands, the joint forces commander is Commander-in-Chief Allied Forces Central Europe (CINCCENT). He is a German general officer, one level below SACEUR in the NATO structure. His command is further subdivided into: the Northern Army Group (NORTHAG, commanded by a British general officer who is also Commander, British Army of the Rhine, in peacetime), Allied Air Forces Central Europe (AAFCE, with a US commander who is also commander of U.S. Airforces, Europe in peacetime), and the Central Army Group (CENTAG, also commanded by a U.S. general officer who serves as CINC U.S. Army, Europe, in peacetime). AAFCE has within it the separate commands of Second and Fourth Allied Tactical Air Forces (TWO and FOURATAF). TWOATAF, to a great extent, reflects the operating style of the Royal Air Force although its headquarters organization reflects the multinational character of the forces (Belgique, Netherlands, German, U.S. and U.K.) it will command in wartime. FOURATAF, on the otherhand, is dominated by the Unites States Air Force and reflects U.S. organizational and operating style, attitudes and concepts of operations.

This highly disintegrated chain of command is largely a product of, and is further exacerbated by, political considerations . There is an absolute requirement within NATO, institutionalized through political reality and administrative procedure, to preserve a degree of national sovereignty in military affairs, a constraint that does not impede the Warsaw Treaty Organization. An additional, potentially problematic characteristic of the NATO deployment of forces in Central Europe is the division of the East-West border into territorial slices, "layer cake" fashion, with each layer defended by the forces of one nation. Thus, at minimum, the forces of five nations (the US, West Germany, the Netherlands, the United Kingdom and Belgium) will be arrayed along the front. It is possible that if the NATO position began to deteriorate, the French would also move forward to take a slice of their own, or be employed as a reserve coming in on top of any of the national corps sectors, or at a corps boundary.

Even from this superficial description of the organizational structure at echelons above corp, it is simple to conclude that the C3 and coordination requirements for for fighting a joint and combined conventional war with a multinational alliance in Central Europe are enormous. That these requirements be met is also of the utmost importance as more and more emphasis is placed upon the importance of conventional defense of Europe to support deterrence. C3 failures make cheap victories possible, as the fall of France in 1940 and the collapse of South Vietnam armed forces in 1975 point out.

THE CARNEGIE SCHOOL OF ORGANIZATION THEORY

The analysis of command and control presented here is based on the tenets of "organization theory" derived from an analysis of the behavioral limits and capabilities of individuals and of organizations. Drawn from the classic writings of Richard Cyert, Herbert Simon, and James March, the "Carnegie School" of organization theory is a set of propositions, inferences, and hypotheses drawn from research in cognitive psychology and human behavior and adapted to the description, analysis, and measurement of behavioral variance in collections of individuals, commonly known as organizations. This approach, well known in its application to problems of business and industrial management, has been applied to problems of command and control by John P. Crecine, Gregory Fischer, Robert Coulam and Michael Salomone at Carnegie Mellon University in the early 1980s, and most recently by the authors focusing on the organizational seams in the NATO Central Region from theater to division level.

BOUNDED RATIONALITY

The Carnegie approach rests on a key premise: certain fundamental cognitive constraints severely limit the human capacity for rational action. Limits on the amount of information that can be attended to force people to respond to only limited aspects of their environment.[1] And limits on the ability actively to manipulate conceptual information force people to rely on relatively simple mental strategies that frequently violate conventional notions of rational inference and choice.[2]

Organizations may be viewed as devices for overcoming individual cognitive limits. Through specialization, individuals acquire the capacity to apply relatively complex cognitive strategies to narrowly defined task environments.[3] Through division of labor, substantial cognitive resources can be simultaneously brought to bear on many tasks or information sources at a time.[4]

Central to this perspective on organizations is the concept of "bounded rationality"; that is, the idea that the limits of human capacity to generate alternatives, process information and solve problems constrains the organizational decision-making process. The five summary points which follow attempt to capture the direction of Simon's thinking on this concept.

First, because most problems which an organization faces are complex, they are factored, or split into quasi-independent parts and dealt with individually, usually by separate units of an organization.

Second, optimization, or finding the best alternative, is replaced by satisficing. This means choosing the first alternative which meets minimum decision criteria, that is acting on the first alternative that is "good enough." One does not look for every needle in the haystack, only for the first one that is sharp enough.

Third, because satisficing requires stopping at the first alternative that is good enough, the order in which organizations develop and propose alternative solutions to problems is a critical determinant of which alternative will be chosen.

Fourth, organizations seek to avoid uncertainty, particularly uncertain future consequences of present actions. Thus decisions which maximize short run feedback are preferred.

Fifth, repertories of responses or routines generally define the range of organizational choice in recurring or routine situations. Thus, the organization will search for analogs of the familiar when confronted with the unfamiliar.

Concentrating on the bounded character of human rationality, Cyert, March and Simon focus their attention on the consequences for the decision environment of less than complete information, limited information processing capabilities, uncertainty avoidance, sequential search procedures, adherence to routine and satisficing as an alternative selection mechanism. Thus, organizational structure and conventional practice become initial factors in the development of goals, the formulation of expectations and the execution of choice.[5]

An organization can be viewed as a complex system of parts each with specialized capabilities. When an organization is functioning in a coherent fashion, it is according to an implicit or explicit plan or strategy for coordinating the behaviors of the various organizational parts. The overall division of labor and coordination strategy for an organization is reflected in the structure of the organization. For an organization, its structure and division of labor defines what has to be coordinated with what. Once the boundaries between various subunits are identified and the functional interdependencies — division of labor between the various subunits — are known, it becomes clear where the key coordination points are.

THE COORDINATION PROBLEM

The potential accomplishments of human organizations are almost limitless. In practice, however, human organizations must overcome a fundamental difficulty — namely, the necessity of assuring that the pattern of activities being carried out by individuals in various subunits of the organization fit together in a relatively coherent fashion that results in progress toward the organization's fundamental objectives. Achieving such coherence defines what is termed the coordination problem. But this structuring task itself can easily become so complex as to overwhelm the cognitive capacities of those who must carry it out.[6] Indeed, the high frequency of "coordination failures" suggests that such problems commonly occur.

COORDINATION STRATEGIES

There are four basic mechanisms for achieving coordination in human organizations.[7] *Direct supervision* is the most obvious. Here, a supervisor exercises a degree of control over the behavior of subunit members by allocating resources to subunits, and by directly influencing certain details of the subordinates' behavior. Using the *mutual adjustment* method, two or more actors agree to share resources and to confer with one another concerning decisions that affect the activities of those involved. *Standardization*, or the creation of standard operating procedures (SOPs), offers a third mechanism for achieving coordination.[8] A supervisor can exercise control over the behavior of subunit members by creating standard procedures governing the behavior of subunit members under specified conditions. The supervisor is thus able to influence behavior even in situations of which he is unaware. Moreover, subunits can coordinate with one another purely on the basis of shared expectations. No direct communication is necessary if each subunit can anticipate which SOPs other subunits will implement and what the outcomes will be. Explicit *planning* is another way to achieve coordination. It is equivalent to direct supervision in advance. Achieving coordination via resource allocation processes or the creation of SOPs requires significant lead times to accommodate the planning process. Direct supervision and mutual adjustment rely on real-time coordination, which is far more difficult and costly than pre-planning.

Whatever the means chosen, achieving coordination is costly.[9] Human and physical resources are directly consumed by the activities of planning, monitoring, and communicating. Also, different mechanisms for achieving coordination produce unintended side effects that are costly to the organization. Direct supervision processes, for instance, place a heavy information-processing burden on those in supervisory roles, and as a consequence, information bottlenecks and delays are likely. Also, because supervisors cannot possibly access all of the information available to their subordinates, this mode of achieving coordination will generally fail to exploit detailed information concerning local circumstances.[10] Mutual adjustment processes, if carried to extremes, can result in extraordinarily high rates of information transmission with every actor communicating with every other actor. The resulting state of information overload can result in delays or total paralysis of the organization. Finally, coordination achieved via standing procedures can result in excessively rigid behavior, particularly in the face of circumstances not anticipated when the SOPs were formulated.

STRATEGIES FOR LIMITING COORDINATION COSTS

In light of the coordination costs mentioned above, organizations invariably devise strategies for limiting the degree of coordination attempted.[11] The strategies they may employ include the following:

Reliance on nearly decomposable task structures. A task structure is strictly decomposable if each subtask can be carried out without regard for how other subtasks are performed.[12] A task structure is nearly decomposable if, in the short run, subtasks can be performed without regard for how others are being performed, and in the long run, depend only on a few aggregate characteristics of how other tasks are being performed.[13] While a degree of monitoring, re-planning and adjustment is required in nearly decomposable task structures, the amount of information that must be gathered and transmitted is much smaller than in highly interdependent task structures.

Ignoring interdependencies. When interdependencies do arise, one crude strategy for reducing coordination costs is simply to ignore the interdependence.[14]

Ignoring interdependencies except in extreme cases. A slightly less extreme strategy for limiting coordination costs is to ignore interdependencies in all but extreme cases.[15] A supervisor might monitor for extreme cases of negative interaction between subunits or the subordinate units themselves might agree to inform one another when their behavior falls outside of agreed upon norms. In a standing procedure mode, SOPs themselves might specify conditions under which supervisors and subunits will transmit information concerning circumstances that fall outside the bounds of the anticipated or acceptable.

Creating buffer stocks of slack resources. Many coordination problems arise when one subunit requests resources or assistance that are to be provided by another subunit Absent detailed coordination between these subunits, there is a distinct likelihood that short-term shortages will result in delays in filling such requests. One crude solution to this problem is to hold large buffer stocks of slack resources to be used for meeting such demands.[16]

Reliance on flexible, general purpose resources within subunits. To the extent that subunits possess flexible, general purpose resources, they are less likely to require assistance from other subunits, thus less likely to need to coordinate with them.[17]

Reliance on standardization as the least-cost means for achieving coordination. Implementation of SOPs is a very inexpensive means for achieving coordination. To the extent that behavior is coordinated via mutual expectations rather than actual communication, coordination can be almost cost free, apart from the loss of flexibility inherent in standardization.[18]

Whatever the merits of the above strategies for limiting coordination costs, they are not without costs of their own. The planning processes involved in creating a decomposable task structure are not cost-free. The costs of ignoring interdependencies between subunit activities are obvious. At the extreme, ignoring interdependence may result in situations in which subunits carry out their tasks in manners that are directly at cross purposes. For example, the tragic Apollo 1 fire that killed three astronauts was facilitated by coordination failures between design groups. One chose a 100% oxygen environment for the capsule while another chose to use materials that were inflammable in most environments, but highly flammable, virtually explosive, in fact, in the 100% oxygen atmosphere of the capsule.[19]

The costs of ignoring all but exceptional cases of interdependence are similar to those of ignoring interdependence altogether, though if the exceptional cases are well chosen, the resulting costs may be considerably smaller in magnitude. Maintaining stocks of slack resources is costly in that the human and capital resources held in reserve are essentially wasted until they are utilized, and there may be inventory costs associated with holding capital resources. Relying on flexible, general purpose resources is also costly because such resources are likely to be more expensive to procure and maintain than specialized resources,[20] and, because of their complexity are likely to suffer from poor reliability. The recent difficulties associated with procuring multi-purpose fighter aircraft are instructive in this regard.[21]

Finally, despite the advantages of relying on standing procedures, this strategy for simplifying coordination processes may result in excessively rigid behavior, especially in the face of an uncertain and rapidly changing environment.[22] Attempts to overcome this shortcoming by devising finely-tuned SOPs, sensitive to changing environmental conditions, will result in substantial planning costs. And if the resulting SOPs are too complex, they are likely to be poorly implemented. The basic message here is to keep it simple. This strategy, however, runs the risk of nor planning for a sufficient range of contingencies. Once again, the key in organizational design is to strike a balance between competing goods.

THE VIRTUES OF HIERARCHY

Designers of organizational decision and control systems are faced with a difficult trade-off between coordination costs and the costs of attempting to limit coordination.[23] Almost invariably, organizations resolve this design dilemma by developing hierarchical task and control structures.[24] Hierarchical control systems are particularly advantageous when the task structure is decomposable, or nearly so. Hierarchical control systems require less communication than other types of control systems. If communication is strictly hierarchical, each actor communicates with only one superior and a small number of subordinates. Furthermore, the complexity of a hierarchical organization is nearly constant across all levels as far as the individual actor is concerned, which is advantageous since the ability to think about more than one complex task at once is not likely to vary significantly among individuals at different levels in the hierarchy. Nearly decomposable hierarchical systems, composed of stable subsystems, have an inherent evolutionary advantage since a change in one component will require little or no adjustment in the others, and damage to one component can be repaired without any alteration of the others. In short, hierarchy substantially simplifies planning and control processes, and it promotes survival in the face of a changing or hostile environment. Whatever the balance struck between the costs of coordination and the costs of limiting coordination, the optimal solution is likely to possess many attributes of a nearly decomposable hierarchy.

CENTRALIZATION AND DECENTRALIZATION: TRADE-OFFS AND MIXED ORGANIZATIONAL STRATEGIES

Discussions of what constitutes a good decision structure or process also frequently revolve around the issue of centralization of control. The degree of centralization is conceptually defined in terms of the number of alternatives available to subordinates that satisfy the basic resource constraints imposed by the superior, and that are compatible with the goals and instructions established by the superior.[25] The greater the number of alternatives available, the less the degree of centralization.

Most control systems are centralized in some respects and decentralized in others. The US Army's mission planning procedures, for example, involve centralized specification of mission objectives and resources, but decentralized planning and execution of mission details. The degree of centralization may also vary by task. Nuclear weapons maintained by the US Army operate under much tighter control than do conventional weapons under Army control.

Higher level control may be of two sorts.[26] The higher authority may physically constrain the options available to subordinates by means of resource allocation decisions. Or, the higher authority may influence the actual decisions of subordinates by transmitting to them goals, task instructions, or information about the internal and external environment of the organization. If effective, the second approach is likely to be more flexible and precise than the first. Invariably, the actual method of control will involve some combination of the two.

The primary advantage of centralization is the increased potential for coordinating subunit activities. The liabilities of direct supervision include delay and the failure to fully exploit local information and expertise. The centralization issue can also be framed in terms of organization based on mission versus organization based on function.

FUNCTION VERSUS MISSION DECOMPOSITION

An organizational hierarchy can be structured in many ways. One critical choice is whether to decompose by function or by mission. Functional decomposition is decomposition in terms of area of specialization. Immediately beneath the top level commander are specialties, then at the next level, subspecialties, and so forth. The US Army can be thought of as being structured in terms of major specialty areas such as infantry, armor, artillery and so forth. These specialties can be further divided into subspecialties; for instance, airborne infantry, air mobile infantry, mechanized infantry and light infantry.

Mission decomposition is decomposition in terms of task or purpose. Thus, the overall commander sits above a group of major mission commanders who in turn oversee a set of submission commanders, and on down the line. The armed forces of the US can be thought of in terms of a number of missions: strategic nuclear deterrence and retaliation; continental air defense of the US; defense of Europe against Soviet attack, to name a few. Each of these major missions can be repeatedly subdivided, of course, in terms of increasingly specific tasks.

In practice, most organizations use a mixture of organization by mission and organization by function. The most common pattern is to organize by mission at the top of the hierarchy and oy function nearer the bottom. In any organization that employs a mixture of mission and functional decomposition, the central design question is at what level the decomposition should proceed in terms of missions and at what level according to specialty and function. A second critical question is how the specialty subunits can be brought into the service of the mission-oriented subunits. There is, of course, no unique answer to either question. Differences of place and circumstance dominate.

The primary advantage of functional decomposition is that it permits the organization to exploit the economies of scale in herent in high degrees of specialization. The major liability of functionally-organized units is that there are very few missions that they can perform by themselves. To perform a novel mission, using functionally-organized subunits, a higher level decisionmaker must provide overall coordination and control between the units involved in the process, or SOPs may substitute if the task is a routine one.[27] Organizations that are functionally structured from top to bottom are likely to incur heavy coordination costs whatever the means employed to achieve that coordination. Because of the need for coordination, functionally structured organizations tend toward inflexibility and inability to adapt to rapidly changing situations.

The primary alternative to functional decomposition is to organize by mission at the top, and by function nearer the bottom of the hierarchy. An obvious cost of this approach is that it reduces the opportunity for exploiting economies of scale, since an organization cannot afford to include highly trained specialists or exotic technology in every major operational subunit.[28] Mission decomposition has other potential drawbacks as well. Mission-oriented subgroups may become parochial in outlook, overestimating the value of their mission relative to other organizational missions.[29] Finally, to the extent that mission-based decomposition results in a highly decentralized command structure, the resulting ambiguity and requirements for unstructured problem-solving by lower level personnel may result in degraded morale or performance if the task requirements exceed the capabilities of the individuals involved. But mission decomposition can substantially reduce coordination requirements (provided that the set of missions involved is decomposable or nearly so). Higher-level decisionmakers need only to assign tasks and allocate resources. Details of execution can be left to the discretion of mission and submission commands, permitting maximum exploitation of local information. It is clear, then that notions of centralization and decentralization are closely linked to whether an organization is structured in terms of mission or function. Mission-based decomposition encourages decentralization, provided that the missions are relatively independent. Function-based decomposition requires higher degrees of coordination between subunits, and hence tends to result in a highly centralized command structure.

ORGANIZATIONAL SEAMS

The various coordination strategies discussed in the preceding section are meant to coordinate organizations, or parts of organizations, despite structural, functional, institutional, or national divisions between them. These "seams" between organizations can be either minor or major impediments to attempts at coordination. Coordination within one unit of an organization is "organic," is often rehearsed, often involves tacit rather than explicit coordination procedures and is generally easier to accomplish for an organization. People within a subunit know each other, know what to expect, have a better ability to anticipate behavior, and often rely on implicit coordination. Coordination across subunit boundaries is often done explicitly and generally involves real coordination costs or detailed plans. It tends to be rehearsed less frequently than coordinated activities within subunit boundaries and is generally much more difficult for an organization.

Assuming that different subunits of an organization must act together, coordinating the activities of heterogeneous units — for example, a ground unit and an air wing— is more difficult than coordinating two similar units — two adjacent US battalions, for example. Coordination between similar units is inherently easier because each unit has a "conceptual model" for the behavior of their counterparts, in essence, "They behave just like we would.".

Illustrative of this approach is the provision of fire support, which can be provided from the ground by artillery that is organic to the ground forces, or from the air by air forces which exist in the U.S. armed forces as a different branch of service. The differences in artillery versus air support are representative of the differences between major and minor organizational seams. Artillery is indigenous to the ground forces. Artillery officers have the same basic and doctrinal training as the forces they support; their command structure is completely integrated with that of the ground commanders, and their communications equipment can easily access the main communications network. Tactical air support, in contrast, is provided by an external organization, namely the Air Force. Air Force officers come from a different systems of military education and background and, their attitudes and preferences reflect different incentives and belief systems. The command structures and planning cycles differ radically between the two organizations, and they have divergent views on allocation of resources among missions and acquisition of types of airframes and ordnance to support these mission priorities.

The difficulties in *coordinating the air and ground wars* further illustrate the coordination problem within NATO. The Air Force and the Army take fundamentally different positions on what form air support for ground forces should take. The Air Force believes that it cannot effectively support battlefield air interdiction and close air support along the forward edge of the battle area until it has established air superiority . The Army wants to see more direct applications of airpower in support of the troops. In addition to these philosophical differences, there are methodological ones. The Army and Air Force operate on entirely different planning cycles, with different time lines. The differences in time lines and planning horizons can have a significant and adverse effect on the interactions between the two services. For example, the ground commander's plans may be severely disrupted or even cancelled if his specific requests for air support or reconnaissance cannot be fulfilled within the time frame required.

Seams in an organization may be either "horizontal" or "vertical" in orientation. The seams between two units on the same level of a hierarchy are considered to be horizontal; seams between actors at different levels are considered vertical. The "height" of the seams, measured as the degree of difficulty in overcoming them, may also vary. The coordination of two infantry battalions in the same regiment, for instance, may be substantially easier than the coordination of two battalions in different divisions. The most insurmountable seams, perhaps, appear across multi-national boundaries. To illustrate this point, note that British aviators in TWOATAF operate under different assumptions and doctrine than their United States counterparts in FOURATAF. British air forces assume that during wartime, they will be operating for some time when they are out of contact with higher command levels. Their training reflects this assumption. In contrast, U.S. procedures assume a high degree of contact with flight controllers throughout a mission and that squadron coordination will be accomplished through realtime communication links.

In addition, the different national military organizations in NATO, the national corps, have nationally provided and therefore different intelligence and logistical support systems on which they will rely during wartime, while operating under the NATO command. These national forces use different communications systems and weaponry, and have different levels of readiness. It is clear that the seams between the elements of the NATO multi-national force are inherently "higher" than the seams that separate different units of the Bundeswehr.

COORDINATING THE OFFENSE VERSUS COORDINATING THE DEFENSE

The kind of coordination required generally varies with the task or function that the organization is trying to perform. For example, in coordinating the activities of various military units in an offensive operation, coordination itself is a command task. In a wartime situation involving two large, competing military organizations, the environment is likely to be uncertain and rapidly changing. Coordinating military organizations having highly specialized functions and equipment in a complex, hostile environment is inherently easier if achieved through pre-planning than if the complex of deployment, logistical, strategic and tactical decisions, mission assignments and force-tasking has to be done in real-time, "on the fly." Real-time coordination requires accurate real-time intelligence, assessment of a fluid battlefield environment, reliable and timely communications, and competent and continuous exercise of command.

The responsible military commander generally has a plan, reflected in the orders given to the various components of the offense. Especially if the military organization on the offense is able to control the pace of battle, coordination is accomplished simply by following the direction of the overall military commander or simply by following the offensive plan. Coordination is generally done through pre-planning or the sending of simple, easily understood orders.

In contrast, coordination of military units involved in defensive operations generally cannot rely on elaborate pre-planning in order to accomplish their task, unless they can predict with sufficient confidence, the exact location, size and timing of the enemy attack. Explicit forms of communication are generally required, and if the communications between various units can be disrupted, delayed, or if the link can be destroyed, the capability of the defensive military organization to operate in a coordinated capacity is severely diminished, if not eliminated. Coordination by explicit communication plays a much larger role in a defensive organization than an offensive one because defensive military operations are reactive, by definition.

The essence of Soviet military doctrine is to control the pace of battle through offensive operations that rely on pre-planning as the coordination strategy. By creating an environment that the defense has to adapt and react to, the Soviets hope to compound the coordination problem for their enemy. By keeping the pace of their offense rapid, the Soviets seem to be trying to force the defense to rely exclusively on real-time coordination strategies that are inherently more costly and difficult.

SIX KEY ATTRIBUTES OF EFFECTIVE COMMAND AND CONTROL

In addition to the aforementioned considerations in assessing effectiveness of command and control, Crecine, Coulam, Fischer and Salomone in their organization theory-based assessment of NATO command and control in the Central Region have defined six essential properties of a command and control system.30

The list of essential properties could certainly be expanded, but research has indicated that these six characteristics bound the irreducible minimum requirements for effective command and control. The properties were placed in two groups. The first group consisted of "performance characteristics" which included [1] processing capacity, [2] reaction time, [3] flexibility, and [4] coordination capability. The second group consists of the "continuity of command" characteristics of [5] physical survivability, and [6] robustness against attrition.

With respect to *processing capacity*, research shows that a C3 system should have as much cognitive, computational, and communications capacity as possible. This requirement is affected not only by numbers of men and levels of equipment and technological sophistication, but also by levels of training and rehearsal, communications discipline, well developed routines and standard operating procedures. Noting that organizations cannot do what they do not rehearse, the procedural becomes at least as important as the technical. For example, a courier on a motorcycle or in a jeep may be a much more effective communications device than a high speed communications link which can transmit great volumes of information to corps headquarters at such speed that the printers and displays which pull it off the wire there are backed up for twelve hours with incoming messages.

The second critical performance attribute identified in the study was *reaction time*. Defined as the amount of time that elapses between the onset of a change in the environment and the initiation of a response. Reaction time is a function of rehearsal and procedure, processing capacity and processing load, and centralization or decentralization of the organization with respect to both structure and decision locus. The greater the ratio of processing capacity to processing load the faster the reaction time and, perhaps less obviously, the more decentralized a C2 system, the greater its ability to respond rapidly to changes in local conditions that affect only one or a few subunits.[31]

The third property is *flexibility*, e.g., the capacity to adapt to rapidly changing conditions. Like the two performance attributes discussed above, a central key to flexibility lies in training and rehearsal. According to Crecine, et. al.:

> Flexibility is enhanced by the development and rehearsal of both tactical and C3 SOPs that cover a broad range of scenarios. This speculation rests on the assumption that the effectiveness of SOPs is inherently scenario dependent. Confronted by unforeseen contingencies, the command system will be forced either to engage in time consuming improvisation or to rely on SOPs that are inappropriate to the conditions of battle.[32]

Coordination capacity is the fourth critical performance characteristic of effective command and control identified by Crecine, Coulam, Fischer, and Salomone. Coordination capacity is determined by a mix of hierarchical and structural variables and procedures. The authors note, for example, that in the field, "C3 systems differ in their ability to plan and conduct 'combined arms' operations in which different types of forces — infantry, tanks, artillery, aircraft — work in conjunction with one another. At a higher level, say the whole European Theater, different C3 systems will differ in their ability to coordinate battles in different sectors, or to reallocate forces and supplies from one major sector to another."[33]

The two continuity characteristics identified in the study, *survivability* and *robustness against attrition* were found to be influenced and enhanced not only by physical means, such as hardening, but also by procedural actions, such as comunications disciplines that minimize the use of radios that can easily be detected by enemy forces. Again procedures, SOPs and organizational capabilities enhanced by realistic rehearsal can be effective substitutes for direct real-time communication if the objective is to avoid detection and possible destruction. Tacit communication, shared expectations, perceptions and beliefs can enhance survivability under conditions where intense hostile action is directed at first locating and then destroying headquarters and C3 systems. Mobility, the separation of command posts from transmission facilities, masking, hiding of facilities and deception are other means to enhance survivability. But as Crecine, et.al. point out: "No matter how great the efforts to enhance C3 survivability, a concerted attack on a C3 system is bound to be partially successful. Some communications links will be jammed, and some components of the system will be destroyed. Any truly survivable system must be able to continue to perform critical functions despite damage or attrition of C3 system components. Thus, the importance of the second property of continuity of command, robustness, is mainly a matter of organizational structure and procedures."[34]

Redundancy in human and physical C3 assets, and procedures for devolution of command, are essential if a damaged or degraded C3 system is to reconstitute itself or degrade gracefully. Since C3 failures make cheap victories possible, under conditions of massive damage to a command and control system, the "system must be robust in the sense of being able to reconstitute itself to form an effective fighting force from those C3 elements and fighting units that have survived."[35] This capability will be enhanced by structures and procedures that permit surviving units to fight on their own while the command structure reconstitutes itself. The study concludes that the many layers of command that exist in the NATO command and its constituent national corps and below "create a substantial potential for reconstituting at many levels, if the survivors can operate alone in the short run."[36] This need to be able to function independently for short periods of time mandates a nearly decomposable task structure that allows subunits to operate constructively on their own. Robustness is also enhanced by realistic exercise of operations under conditions of degraded communications and the physical loss of headquarters or other command facilities. It is in realistic training that NATO may have one of its greatest inadequacies.[37]

They conclude that although there are measurable performance attributes associated with the functioning of a command and control system along the six dimensions listed above, the effectiveness of a system of command and control can only be evaluated in the broader context of a specific scenario. The interplay of the battle scenario and the operational concepts employed by each side will be a prime determinant of the appropriateness and effectiveness of the respective command and control concepts, systems and capabilities of the combatants. For example, with respect to opposing military forces, they conclude that: "...the capacity for centralized coordination is less important to a military force that relies on independent action undertaken by isolated defensive units than to one that plans to establish and hold an unbroken line of defensive forces." 39 This tenet concerning scenario, operational concept and command and control structure and process is a condition of the interaction of organization and task environment. It is not only an aspect of military organization and command and control, but is germane to any organization and task environment.

Two themes clearly emerged from the study. The first is that actual rehearsal under realistic conditions provides the surest basis for effective action. Rehearsal serves to enhance the degree to which coordination can be achieved through reliance on shared knowledge and expectations. Simplicity is the second hallmark of effective action. Alternatives that create minimal coordination requirements are most likely to succeed. To the extent that direct means of coordination are essential, they should be based on coordination SOPs that are themselves simple and well-practiced.

CROSS DIFFERENCES IN NATO AND THE WTO FOR CONVENTIONAL WAR IN THE CENTRAL REGION

When one examines the composition of combat forces making up NATO and the WTO, it is clear that NATO consists of a multi-national force whereas the Soviet military organization dominates all planning for combat missions and combat forces in the WTO. NATO will have a minimum of five nationalities holding sections of the front line and the Central Region; the Americans, West Germans, British, Belgians, and the Dutch.

The multi-national character of NATO combat forces creates in itself some rather obvious command, communication and coordination problems that are different than those faced by the all-Soviet WTO combat force. Although there is a degree of cultural and linguistic diversity within the Soviet military, this is a fundamentally different problem for the Soviets than, for example, the coordination of adjacent US and West German forces or UK and Belgian forces. The Soviet forces are dominated and unified by the Russian language. Even among non-Russian Soviet officers, the language of their training is Russian.

Another gross difference apparent between NATO and the WTO is the degree of centralization in the command structure. The WTO tends to be more centralized than NATO, with key strategic and tactical decisions being taken one or two levels in the hierarchy above that of NATO. In addition, it is clear that the NATO command structure places far greater reliance on real-time communication as a way of coordinating the behavior of the various units making up the overall NATO force. In contrast, the Soviet/WTO organization tends to rely much more on detailed preplanning as a way of coordinating various units. This reflects partly the relative capabilities of the two sides, partly cultural as well as socio/political preferences, and partly different strategic concepts and approaches to conventional warfare. NATO quite clearly sees itself primarily as fighting a defensive war and finding itself in situations where elaborate pre-planning is of marginal utility because of the reactionary nature of the defense.

Another clear difference between NATO and WTO command structures lies in the way in which the air/land battle is coordinated. In NATO, the coordination takes place through parallel command structures, one for the allied air forces, and one for the ground forces. The primary level of integration for air and ground commands is at the army group level. In the WTO both air units and ground support units are organic to the ground forces, and utilize the same command structure. Soviet ground commanders at the division and army level each "own" appropriate air forces, whereas their counterparts in NATO may directly command ground forces but the associated air forces are "owned" by another command structure and necessary air support must be requested from the parallel structure.

Finally, if one looks at the echelons above corps in NATO one observes an overlapping and "messy" command structure with respect to intelligence, logistics, air/land battle coordination, and troop deployment/redeployment decisions. The echelons above corps in the Soviet military exhibit a much "cleaner" structure for assignment of responsibilities and chain of command. Much of the "messiness" of the NATO command structure in the echelons above corps are due to the multi-national nature of the NATO alliance and combat force and the difficulties of mixing different national forces in a single command structure.

ACKNOWLEDGEMENTS

Research that made this paper possible was sponsored by Mr. Andrew Marshall, Director of Net Assessment, Office of the Secretary of Defense, and Dr. Stuart Johnson, Director, Command and Control Research Program, Institute for National Strategic Studies, National Defense University. The authors are grateful to them for their support and encouragement. The authors also wish to thank research assistant Ms. Lisa McAnany for her assistance in the preparation of this manuscript.

REFERENCES

[1]Norman, D.A., *Memory and Attention*, 2nd edition, New York: John Wiley, 1976.

[2]Simon, H.A., *The Sciences of the Artificial*, 2nd edition, Cambridge: The MIT Press, 1981; Kahneman, D., Slovic, P., and Tversky, A., *Judgement Under Uncertainty: Heuristics and Biases*, Cambridge: Cambridge University Press, 1982; Tversky, A. and Kahneman, D., "The Framing of Decisions and the Psychology of Choice," *Science*, No 211, 1981.

[3]Chase, W.G. and Simon, H.A., "Perceptions in Chess" *Cognitive Psychology*, 1973.

[4]March, J.G. and Simon H.A., *Organizations*, New York: John Wiley, 1958.

[5]See Cyert, R.M. and March, J.G., *A Behavioral Theory of the Firm*, Englewood Cliffs, NJ: Prentice-Hall, 1963, Simon, H.A., *Models of Man*, New York: John Wiley, 1957. For an example of this schema applied to the problems of the president as chief executive officer in crises, see Paul Y. Hammond and Michael D. Salomone, *Constraints and Boundary Conditions Affecting C3I Survivability*, Marina del Ray: The Analytical Assessments Corporation, 1981.

[6]March and Simon, *Organizations*, 1958.

[7]Mintzberg, H., *The Structure of Organizations*, Englewood Cliffs, NJ: Prentice-Hall, 1979, Malone, T.W., *Organizing Information Processing Systems*, Working Paper, Palo Alto Research Center: XEROX, 1982.

[8]Emery, J.C., *Organizational Planning and Control*, New York: MacMillan, 1969.

[9]March and Simon, *Organizations*, 1958; Emery, *Organizational Planning and Control*, 1969; and Malone, *Organizing Information Processing Systems*, 1982.

[10]March and Simon, *Organizations*, 1958.

[11]Emery, *Organizational Planning and Control*, 1969.

[12]Simon, *The Sciences of the Artificial*, 1981.

[13]Ibid.

[14]Emery, *Organizational Planning and Control*, 1969.

[15]Ibid.

[16]Cyert and March, *A Behavioral Theory of the Firm*, 1963.

[17]Emery, *Organizational Planning and Control*, 1969.

[18]March and Simon, *Organizations*, 1958.

[19]This is one of the more tragic and striking examples of the dangers of optimizing within narrow boundaries.

[20]Emery, *Organizational Planning and Control*, 1969.

[21]Fallows, J., *National Defense*, New York: Random House, 1981; Coulam, R., *Illusions of Choice: The F-111 and the Problems of Weapons Acquisition Reform*, Princeton: Princeton University Press, 1977.

[22]Emery, *Organizational Planning and Control*, 1969.

[23]Ibid.

[24]Simon, *The Sciences of the Artificial*, 1981, and Emery, *Organizational Planning and Control*, 1969.

[25]Emery, *Organizational Planning and Control*, 1969.

[26]Ibid.

[27]Williamson, O.E., and Ouchi, W.G., "The Markets and Hierarchies Program of Research," in A.H. Vandeven and W.F. Joyce (eds), *Perspectives on Organizational Design and Behavior*, New York: John Wiley, 1981.

[28]The practice of holding specialty resources aside to be temporarily attached to mission subunits is an effective strategy for overcoming this problem. The main cost of this strategy is that it increases the degree of involvement by the higher level commander. See Emery, 1969 and Simon, 1981.

[29]It is not clear, however, that functionally structured organizations should be any less prone to subunits parochialism. Simon, *Administrative Behavior*, 1976.

[30]Coulam, Crecine, Fischer, Salomone, *Problems of Command and Control in a Major European War*, Marine del Ray: The Analytical Assessments Corporation in conjunction with Carnegie Mellon University, 1983, pp. 67-73, (hereafter referred to as Crecine, et.al); and Coulam and Fischer, "An Organizational Theory Approach to the Assessment of Military Command and Control Capabilities," in R. Coulam and R. Smith, (eds), *Advances in Information Processing in Organizations*, Volume II, Greenwich, CT: JAI Press, 1985.

[31]Crecine, et.al., p. 47.

[32]Crecine, et.al., pp. 72-73.

[33]Ibid., p. 72.

[34]Crecine, et.al., p. 50.

[35]Ibid., p. 74.

[36]Ibid., p. 75.

[37]Ibid., p. 51.

[38]Ibid., pp. 75-76.

PLANNING WITH UNCERTAIN AND CONFLICTING INFORMATION

Daniel Serfaty
Elliot E. Entin
Robert R. Tenney

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, Massachusetts 01803-4901

ABSTRACT

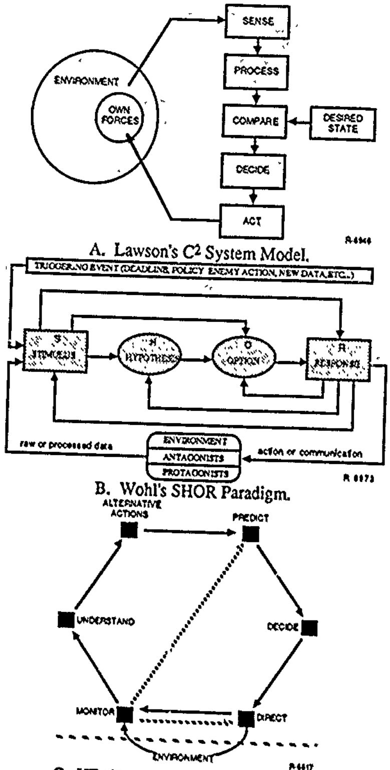
A fundamental function of commanders in a headquarters involves the sequential processing and integration of information from multiple sources. An experiment was designed to test hypotheses related to the way experienced commanders process information from different sources, update currently held beliefs in the light of new and conflicting information, and plan the realignment of troops to face possible enemy attacks. The experimental simulation was conducted in the context of the Headquarters Evaluation Assessment Tool (HEAT) project. Three army officers received sixteen experimental trials in a randomized order. Each trial presented an initial intelligence message surmising the enemy's attack direction and information reliability, a map displaying the tactical situation show information that either confirm or disconfirm the notions expressed in the intelligence message, and two subsequent intelligence updates that adjusted the enemy positions depicted on the map. The update information could confirm or disconfirm the initial attack direction estimate. Findings showed a strong recency effect in all experimental conditions due to the relative order of positive and negative evidence. Information received last had the strongest impact on the commander's attack direction estimates. It was also found that the commander's confidence level in the reliability of his information sources was a function of the consistency of the source. The commander's planning behavior followed a pattern different from the fluctuation of their beliefs. A primacy-like effect was observed in the commander's planning process: most of the troop movements and reassignment were done in the initial stage. Lastly, two models for the revision of beliefs were developed and validated: both the normative-descriptive Bayesian model and the contrast-inertia descriptive model predicted the commanders attack direction estimates as well as the strength of their beliefs in the estimate. The empirical evidence provided additional support to a "rolling" plan theory. It strengthened the assumption that commanders modify their estimates and plans in a continuous, evolving, and dynamic fashion rather than in a discrete, epochal, cycle-to-cycle manner.

I. INTRODUCTION

The last ten years have witnessed the appearance of several paradigms to describe Command and Control (C^2) processes. These include Lawson's Process Model [1, 2], Wohl's SHOR (Situation, Hypothesis, Options, Response) paradigm [3, 4], and the HEAT (Headquarters Effectiveness Assessment Tool) Adaptive Control paradigm [5]; these models are depicted in Figure 1. All three models view the decisionmaking process as a cycle comprising an assessment of the situation, a comparison of the current situation to the situation desired by the decisionmakers, and a selection of an option (or options) to maintain or achieve the desired situation.

An outstanding issue is whether the planning cycle is best viewed as an evolutionary process, supporting the "rolling plan" concept suggested by Cushman [6], or as a sequence of independent processes. The latter point of view implies that the - esses carried out at each planning cycle can be understood (and analyzed) without regard to those carried out in planning cycles that came before or after. This point of view does not imply that decisionmakers have no memory of the past, only that they do not actively reach decisions at one cycle to foster the decisionmaking process at a later cycle. Conversely, the "rolling plan" point of view implies that the understanding of the situation assessment process requires consideration of potential future the perceptions and decisions It also implies that understanding of the option selection process requires a consideration of plans and decisions that decisionmakers anticipate they may make in the future This point of view stresses learning in order to reduce uncertainty as an important characteristic of the decisionmaking and planning process. Lawson [2] and Wohl [7] propose rate of change of uncertainty as an indicator of effectiveness for C^2 systems. Indeed, probing and hedging strategies selected specifically to allow decisionmakers to learn more about their environment before selecting a course of action, can be understood only from the perspective of decisionmaking and planning as an evolutionary process.

While the Process and SHOR models proposed by Lawson and Wohl, respectively, view the planning and decisionmaking cycle as an evolutionary method, the current HEAT Adaptive Control paradigm decidedly views this cycle as a succession of independent processes. This conclusion is evident in the omission of measures for correlating observations from one planning cycle to the next in the current set of HEAT measures. This omission suggests a weakness in the current HEAT should the rolling plan concept prevail. The current HEAT methodology cannot reveal how decisionmakers learn about their environment, select options to facilitate learning (e g , probing), use learning



A. Lawson's C² System Model.

B. Wohl's SHOR Paradigm.

C. HEAT Adaptive Control Paradigm.

Figure 1. Three Models Describing the Command and Control Processes

in a feedback fashion to select options, and hedge decisions in anticipation of course of action they may pursue in the future.

The purpose of this study is to better understand the planning process conducted by a military headquarters. This study serves three objectives. First, to determine whether successive operations of the HEAT planning cycle are best viewed as an evolutionary process or as a sequence of independent processes. Second, to determine the nature of the correlation between successive planning cycles should the analysis support the suggestion that planning is an evolutionary process. Third, to determine how commanders incorporate both uncertainty and the confidence they assign to information they receive into the overall planning process.

II. EXPERIMENT HYPOTHESES

HYPOTHESIS I

Our first hypothesis seeks to confirm that measures taken of the planning process ('process' measures) are correlated over several planning cycles. The arguments for this hypothesis are found in normative theories for estimation as well as in descriptive theories from cognitive science. Estimation theory tells us that optimal estimators (e.g., Bayesian estimators, Kalman filters) combine prior estimates of the random variable under consideration with new measurements to yield posterior estimates. Indeed, the optimal estimators uses not merely the prior estimate of the variable, but the prediction of its value based on the prior estimate. Normative-descriptive theories of hypothesis generation and selection from cognitive science tell us people use prior hypotheses to adjust their beliefs in response to new data. In this respect, the Monitor and Understand processes conducted during a single HEAT cycle yield both estimates for the immediate cycle, and prior statistics for future planning cycles. Therefore we expect a correlation between process measures taken of these processes over several planning cycles.

The argument for this hypothesis is also found in normative theories for information-seeking that have been developed for the control of stochastic systems. Stochastic systems typically involve parameters that are only partially known to the controller or decisionmaker; this is known in military terms as the 'fog of war'. The control of these systems typically involves an element of active learning wherein information-seeking strategies (e.g., probing) are selected by decisionmakers in anticipation that they will better understand their system and use this understanding to advantage in future decisions. Active learning is sound only if the commander places a value on information that will be available to him in the future; it provides no advantage if the commander gives no weight to the future. (The complement to active learning is passive learning where decisionmakers learn about their system through feedback, but do not select options explicitly to facilitate learning.) The implications of active learning are that decisionmaking and planning is an evolutionary process; decisions made in the past are the means for making decisions in the future. Active learning is an element of all successful military enterprises, and is in evidence when a commander decides to conduct surveillance and reconnaissance ("Knowledge of the country is to a general what a rifle is to an infantryman," [8]). We expect that option selection activities (e.g., the Generate Alternative Courses of Action, Predict, and Decide processes) in the HEAT planning cycle should reveal a progression of decisions over several successive planning cycles.

HYPOTHESIS II

Our second hypothesis states that process measures are correlated to the confidence that commanders place in the information they receive. Arguments for this hypothesis are found in Estimation theory and in Stochastic Optimal Control theory. Estimation theory tells us that information should be given a weight that is inversely proportional to the confidence or certainty that can be attributed to that information. Stochastic Optimal Control theory tells us that the optimal control of a stochastic system is achieved through a control policy that is conditioned on the uncertainty surrounding the state of the system at any time [9]. For example, optimal control policies for arbitrary finite-state Markov processes are functions of the state probability distribution at any time. Our argument does not presume that humans are optimal estimators or optimal controllers, but that they are sufficiently rational to exhibit important characteristics found in optimal estimators and optimal controllers.

HYPOTHESIS III

Our third hypothesis states that decisionmakers are sensitive to the order of presentation of information that confirms or disconfirms prior beliefs when they assess a situation. The argument for this hypothesis is based on several models that have been proposed in cognitive science to describe human decisionmaking. We now consider briefly these models.

It is generally acknowledged that decisionmakers do not learn nearly as much from newly acquired information as the information warrants [10-12]. Tolcott, et al. [13] note that decisionmakers usually place undue confidence in the correctness of prior decisions and assessments (decisions and/or assessments hereafter referred to as 'positions'), supporting the idea that people do not readily learn from experience. Decisionmakers tend to anchor to prior beliefs by inadequately adjusting their position upon consideration of new information. Citing the earlier work of Wason [14] and Einhorn [15], Tolcott, et al., point out that decisionmakers tend to seek information that confirms their prior position, and ignore evidence that disconfirms their position. Lopes [16] and Tolcott, et al., attribute this cognitive bias to a process where a prior position is established, and new information is selectively used to adjust the position. This process produces a final position that lies somewhere between the prior position and the position that would be obtained only on the basis of the new information. Tolcott, et al., also offer the hypothesis that newly acquired information is not necessarily combined with old information, but is combined with judgments based on the old information.

Einhorn and Hogarth [17] offer a different view of the picture. Like Lopes and Tolcott, et al, they note that 1) people judge new information in the context of their prior beliefs, and 2) information received by a decisionmaker may support or refute his prior belief. However, their own research indicates that people tend to place greater emphasis on newer information than on prior information. They propose a theory of information integration that is based on three general principles describing how people process information. They hypothesize that:

- Information is updated and beliefs changed through a sequential anchoring and adjustment process.
- A decisionmaker classifies incoming information as either supporting or refuting the currently held position.
- The weight assigned to new information is dependent on the confidence maintained by the decisionmaker in the anchor.

Einhorn and Hogarth have developed a number of mathematical models of situation assessment and decisionmaking based on these principles. These models capture both the anchoring and adjustment process, as well as the process decisionmakers employ for dealing with confirming and disconfirming evidence. Section V will focus on the validation of these models using the experimental data collected in this study.

III. EXPERIMENTAL METHOD

EXPERIMENT OVERVIEW

Information integration and planning processes are studied in the context of a table-top exercise. The protocol is an elaboration of two pencil-and-paper games, one employed by Entin, et al., [18] and one described by Tolcott, et al, [13]. As a background to the exercise, commanders are told that the relations between the Soviet Union and the United States have greatly deteriorated and that a clash of forces in several parts of the world is imminent. The subjects are asked to assume command of the 52nd Mechanized Infantry Division, stationed either in the Fulda Gap, West Germany or near the DMZ in Korea. Their primary duty is to consider all the available intelligence information and predict the enemy's most likely main avenue of attack. Based on this estimate of enemy intent, they are to construct a plan allocating and reassigning resources to best thwart the attack.

The experiment passes through four phases.

First, the commander receives an intelligence brief describing the intelligence analysts' best estimate of the enemy's avenue of attack. This estimate is purportedly based on information that is from two to ten days old. Following a brief time to consider the message, the commander is asked to give his own estimate of the attack avenue and his confidence in the information supplied.

Second, the commander is shown a map (Fig. 2) of the Fulda Gap (or the Korean DMZ area) area depicting the deployment of their troops and the best estimate of the enemy's positions and strengths. The commander is told that Orange troop information displayed on the map is between 12-16 hours old. After a short time to review the situation map and his own force list, the commander is asked to give a second estimate of the enemy's most likely attack approach and his confidence in the information provided to date. Following these assessments, the commander has approximately 20 minutes to develop a plan adjusting troop positions and deploying reserve units (e.g., an calvary and assault helicopter units) to face the enemy's attack.

Third, the commander receives the first of two intelligence updates of the Orange force dispositions. The update may show no change to Orange force locations or may show some adjustments, corrections, additions, or deletions to Orange dispositions. The commander is told that this information is six to eight hours old. Given this new information, the commander has an opportunity to revise his attack approach estimate, confidence assessment, and plan.

THE FULDA GAP

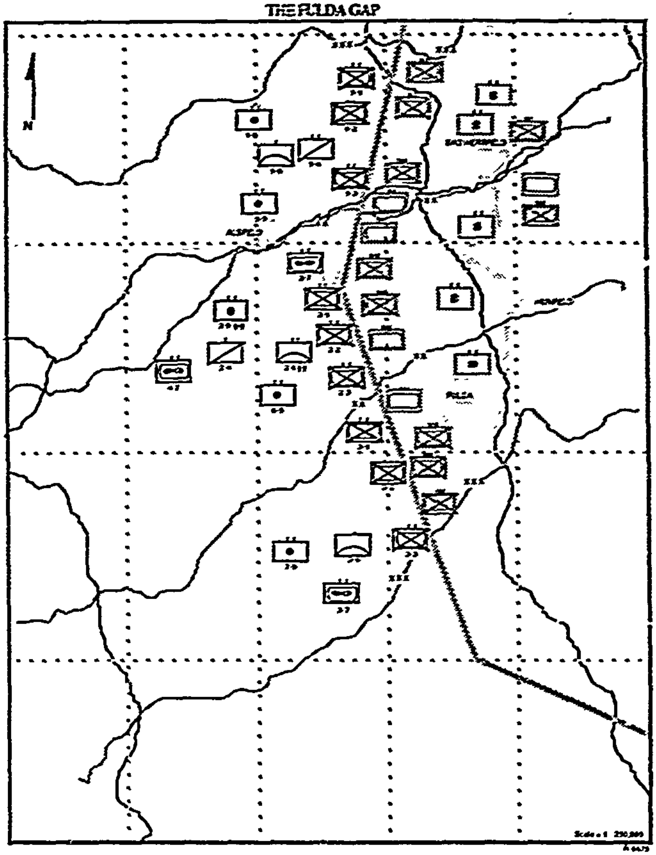


Figure 2. Map from the West Germany Fulda Gap Scenario Showing a Discernable Northern Attack Bias.

Fourth, the commander is given the second intelligence update of the Orange force dispositions. Once again, the information contained in the update may depict some alteration to Orange's force deployments, or show no change. The commander is told that this information is one to four hours old. The commander is then allowed one last time to revise his attack estimate, confidence assessment, and resource allocation plan.

The experiment structure allows the manipulation of three independent variables. The first independent variable is enemy avenue of attack, which refers to the enemy's most likely origin of attack (e.g., North or South) predicted by the intelligence analysts in the (first) intelligence report. The second independent variable is represented by the reliability (high or low) attributed by the intelligence analysts to the information presented in the intelligence brief. The third independent variable is the order of presentation of the briefings to the commander. Order determines whether the information depicted on the situation map (the position of Orange forces) and each intelligence update confirms or contradicts the attack approach estimate given in the first intelligence report.

SUBJECTS

One captain and two majors from the Army's 101st Airborne Division, Ft. Campbell, Kentucky, served as subjects. All the officers had at least 12 years of service, commanded at the platoon level, and the two majors had served as company commanders.

INDEPENDENT VARIABLES

The experiment manipulated three independent variables: 1) enemy avenue of attack; 2) information source reliability; and 3) information presentation order. We now describe each in turn.

Enemy Avenue of Attack — defined as the intelligence analysts' prediction of the enemy's main avenue of attack. This was presented to establish a particular prior position in the minds of the subjects of enemy's attack intentions. Information in the form of an intelligence brief was systematically varied to yield an estimate of the approach (north versus south in the Fulda Gap scenario and east versus west in the Korean-DMZ scenario) the enemy would likely use to launch a major attack.

Information Source Reliability — defined as the confidence or reliability the intelligence analysts (purportedly) attribute to the source of the enemy deployment information provided to the subjects. Theories from cognitive science suggest that individuals weigh information by its perceived reliability [17], [19]. Thus, information perceived as relatively high in reliability should have a larger impact on an individual's estimate of enemy position than information perceived as lower in reliability. The intelligence messages provided in the low reliability condition were given a low to moderate confidence evaluation by the intelligence analysts. The intelligence messages provided in the high reliability condition were given a moderate to high confidence evaluation.

Every written intelligence brief shown to a subject contained two parts: the first delineating a particular avenue of enemy attack surmised by the intelligence analyst , and the second indicating the analysts' confidence or reliability in the information used in their estimation.

Two different north (or east) and two different south (or west) avenues of approach were crossed with two different low reliability and two different high reliability confidence levels to produce the sixteen intelligence messages for each scenario required for the experiment.

Order — After reviewing the initial intelligence message, each subject received three information messages sequenced in time. The information in these messages either confirmed or disconfirmed information regarding the enemy's intended avenue of attack given in the initial intelligence message. Order was thus defined as the sequence of confirmatory and contradictory evidence presented in the three information messages. The manipulation of this variable provided the means necessary to test for the contrast effect predicted by the accretion and discount models of Einhorn and Hogarth [17]. There are eight (2^3) possible permutations of ordering within the three information messages provided to the commanders following the initial intelligence message. Two of these permutations are 'all confirming' and 'all disconfirming', and yield no data to confirm or refute the contrast effect predicted by the Einhorn and Hogarth model. This model also predicts that the contrast effect will be relatively weak if confirming information alternates with disconfirming information (i.e., confirm-disconfirm-confirm and disconfirm-confirm-disconfirm). In short, four orderings were not used. The information presented in the three information messages provided the commanders (referred to as Situation Map, Update One, and Update Two, respectively) was manipulated across four sequences as shown in Table 1.

TABLE 1. SEQUENCES OF INCOMING EVIDENCE

Level	Situation Map	Update One	Update Two
1	Confirm	Confirm	Disconfirm
2	Confirm	Disconfirm	Disconfirm
3	Disconfirm	Confirm	Confirm
4	Disconfirm	Disconfirm	Confirm

SCENARIOS

The experimental trials were nested within two scenarios. These scenarios were used to control for learning effects and to insure that results were not scenario specific. Eight of the experimental trials depicted force laydowns in and around the Fulda Gap, West Germany, while the remaining eight trials showed force laydowns at the border between North and South Korea where the Imjin-Gang River crosses the demilitarized zone (DMZ). In both scenarios, initial Blue troop positions were held constant (and subjects were informed of this fact), but Orange positions varied with experimental condition. For half the trials, Orange laydowns depicted a discernable northern bias for the Fulda Gap scenarios and an eastern bias for the Korean scenarios. That is, a large concentration of Orange forces in the Fulda Gap scenario appeared in the northern sector to give the impression that the enemy was preparing to attack using the northern corridor. Similarly, Orange laydowns in the remaining Fulda Gap and Korean scenarios depicted discernable southern and western biases, respectively. Figure 2 illustrates a schematic of the Fulda Gap Situation Map depicting a northern bias.

DEPENDENT VARIABLES

Three dependent variables were assessed after the initial intelligence message and after each of the three information messages for each trial:

- prediction of the avenue of the enemy's attack (i.e., North versus South for the Fulda Gap scenario or East versus West for the Korean scenario);
- estimation of the probability that the specified attack approach would be used;
- confidence in the information received at that point.

Following the introduction of the Situation Map showing Blue and Orange laydowns (i.e., the first information message), the subjects were asked to develop a plan for allocating (or reallocating) their resources to best counter the suspected avenue of the enemy's attack. After each of the updates that followed (i.e., information messages two and three), the subjects were given the opportunity to revise their plan to accommodate changes in their estimate of the enemy's intended avenues of attack. Subjects wrote out their plans on blank lined paper.

Four additional dependent measures were derived from this initial plan and its subsequent revisions:

- the number of units given orders to move to a new position;
- the number of units given orders reassigning them to a new authority;
- the number of future orders to move or reassign units (including conditional orders such as: "if the enemy attacks from the south, the 1-5 mechanized infantry battalion will be moved to such-and-such a position;") and
- the number of units reassigned or moved to a Division or to a central position. This measure was used to assess Blue's hedging tendencies.

EXPERIMENTAL DESIGN

The experiment design called for a 2 (Enemy Avenue of Attack) × 2 (Information Source Reliability) × 4 (Order) factorial design. Each of the two scenarios were used for half of the experimental conditions; pairing of scenarios and conditions was random. Each subject was randomly assigned to all of the treatment conditions.

IV. EXPERIMENTAL RESULTS

OVERVIEW: THE DECISION PROCESS

This section describes results obtained in the experiment. Then results are interpreted to identify behavior trends and test our three experimental hypotheses. The decisionmaking processes of the subjects-commanders in Figure 3 are summarized in this experiment.

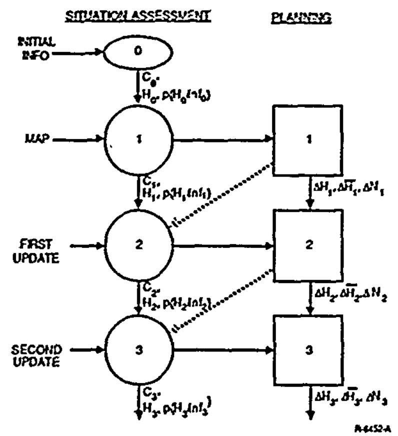


Figure 3. Commanders' Decision Processes.

We use the following notation:

$i = 0, 1, 2, 3$		stages in situation assessment process
$i = 1, 2, 3$		stages in planning process (no planning at stage 0)
C_i	:	subjective confidence in the incoming information [%]
H_i	:	current hypothesis (enemy's likely avenue of attack)
$p(H_i / inf_i)$	:	probability that hypothesis H_i is true given the evidence received at stage i (strength of belief in the current hypothesis) [%]
ΔH_i	:	number of units moved or reassigned in the direction of the current hypothesis
$\Delta\bar{H}_i$	:	number of units moved or reassigned in the direction opposite to the current hypothesis
ΔN_i	:	number of units moved or reassigned in a neutral area (division command or central region)

The decision process in question is made of two separate but interacting processes. First is the situation assessment process, in which the commanders combine, in a sequential manner, incoming information or evidence (e.g., map) with previously held beliefs to produce an updated hypothesis as to the likely enemy avenue of attack. Second is the sequential planning process in which the commander uses knowledge of his troops' positions and his current beliefs to make decisions concerning the resulting hypothetical movement of his troops to best defend against the possible enemy attack. Note that no actual movement of troops occurs, this is on-paper-planning only. We conclude that these decisions are made in a non-reactive hostile environment. Consequently we assume a "separation principle" between the two processes. The dual process of situation assessment and planning pertains to the "UNDERSTAND" and "DECIDE" nodes in the HEAT hexagon paradigm.

The results presented in this section describe the step-by-step evolution of all the variables previously described.

SITUATION ASSESSMENT RESULTS

At each stage in the situation assessment process, the commanders were asked to provide a hypothesis with respect to the enemy avenue of attack, a probabilistic assessment of that hypothesis (strength of belief) and a confidence in the evidence they are presented with. To simplify the data presentation, all probabilities refer to the initial hypothesis. Note that since the hypothesis is a binary one, i.e., North or South, a 0.7 probability of a northern attack is equivalent to a 0.3 probability of a southern one. Table 2 represents an aggregate of the probability data averaged across subjects, for each information sequence (order condition). Table 3 represents the average confidence data. For clarity of purpose, results of Tables 2 and 3 are plotted in Figs. 4-7. These graphs

TABLE 2. AVERAGE PROBABILITY (STRENGTHS OF BELIEFS IN CURRENT HYPOTHESIS)

	Average probabilities* and standard errors			
Order	P0	P1	P2	P3
CCD	68.3 (16.9)	75.8 (18.2)	82.9 (21.8)	41.7 (36.7)
CDD	65.4 (18.9)	80.4 (23.5)	52.5 (26.0)	25.8 (38.7)
DCC	70.8 (21.3)	55.0 (25.5)	75.0 (26.2)	85.0 (31.4)
DDC	66.7 (26.7)	50.0 (26.1)	49.6 (28.1)	57.1 (28.8)

* Probability that an enemy attack will occur from the initially assumed direction.

TABLE 3. AVERAGE CONFIDENCE LEVELS

	Average confidence levels* and standard errors			
Order	C_0	C_1	C_2	C_3
CCD	65.8 (13.5)	72.5 (16.5)	75.8 (17.3)	59.6 (13.8)
CDD	62.5 (15.3)	72.9 (20.1)	67.9 (17.8)	74.2 (19.7)
DCC	69.6 (13.0)	62.0 (13.5)	60.0 (12.8)	74.4 (11.1)
DDC	59.6 (16.1)	57.5 (16.9)	69.2 (17.3)	67.9 (18.1)

* Confidence level in estimated probability of attack as assessed by the subjects.

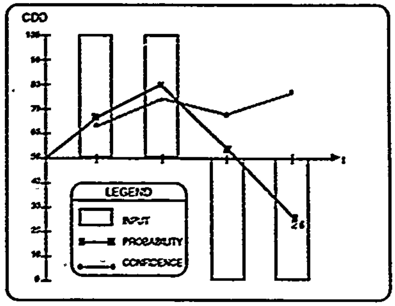


Figure 5. Situation Assessment Results (Probability and Confidence) for Sequence CDD.

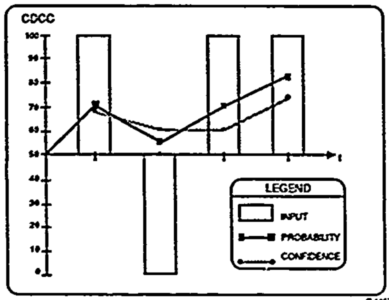


Figure 6. Situation Assessment Results (Probability and Confidence) for Sequence DCC.

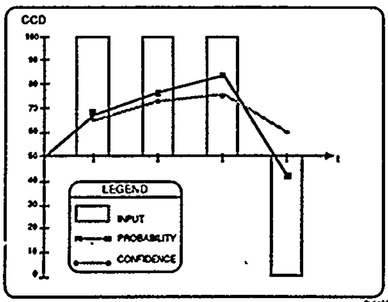


Figure 4. Situation Assessment Results (Probability and Confidence) for Sequence CCD.

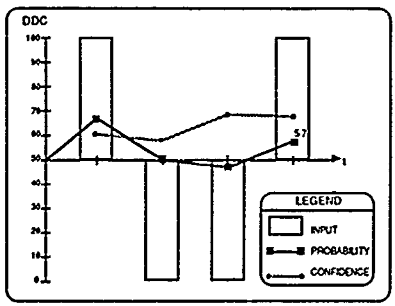


Figure 7. Situation Assessment Results (Probability and Confidence) for Sequence DDC.

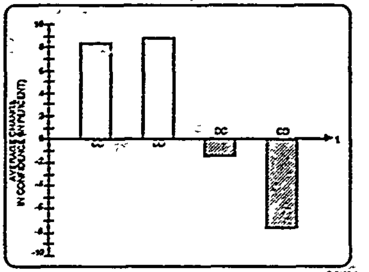


Figure 8. Change in confidence level as a function of the consistency of the source.

summarize all the input, probability and confidence data for each order conditions and provide useful insight into the dynamics of the commanders' information integration and belief updating processes.

Key observations can be made from the data presented so far. They indicate some essential trends in decision-making behavior. Most were advanced in our experimental hypotheses, but some surprised the experimenters nevertheless. The probability results are all significant at the $p < 0.05$ level.

Observation 1: Commanders, without exception, appear to behave rationally at each stage of the situation assessment process.

A quick look at the graphs will convince the reader that the subjects react to the new information by updating their beliefs in the direction of the incoming evidence. It does not necessarily mean that they change their current hypothesis. For example, in the CDD case, the first piece of evidence confirmed the initial hypothesis and brought up the probability from 65.4% to 80.4%. The first disconfirmatory evidence lowered the strength of the belief in the initial hypothesis to 52.5%, but not quite enough to make the commanders' switch to the opposite hypothesis. The subjects needed a second piece of disconfirmatory evidence to lower their probability assessment to 25.8%, thereby adopting the opposite hypothesis as their final belief.

These results although predictable are intuitively appealing. We are dealing with experienced commanders who understand the value of information and know how to integrate it with their current belief. But do they weigh correctly this new information with their current belief? The next observation attempts to answer this question.

Observation 2: The sequence in which the new evidence is presented makes a difference.

The order effect is omnipresent in the data. For instance, let us compare sequences CDD in Fig. 5 and DDC in Fig. 7 They both contain two pieces of disconfirmatory evidence and one piece of confirmatory evidence. However the final average probabilities (strength of belief) are drastically different: $p_3 = 25.8\%$ for CDD versus $p_3 = 57.1\%$ for DDC. These numbers not only reflect a quantitative difference but also represent an opposite final hypothesis as to the likely avenue of enemy attack. This result is remarkable in that it shows clear evidence of the phenomenon of recency. It validates the hypothesis advanced by Einhorn and Hogarth as to the presence of recency effect in any mixed evidence sequential information integration process. One should note that such effects are not predicted by classical Bayesian decision theory. Section 5 will be devoted to a more elaborate discussion on the subject of modeling. Our third experimental hypothesis was validated by the finding of a strong order effect ($p < 0.01$). It supports our conjecture that in addition to the quantity and quality of information entering a headquarters, the order in which this information is presented has a key effect on the judgment quality of the decisionmakers involved.

Observation 3: The commanders confidence level in the incoming evidence tend to increase slightly over time regardless of their initial and subsequent judgments.

Although not statistically significant, this long-term growth trend has been shown in other studies (see [13] for example) involving sequential information processing in a command and control environment.

Observation 4. The confidence level in the reliability of information sources is a function of the consistency of the source.

This is a short term trend pertaining to the confidence levels behavior over time. As shown in Fig. 8, commanders will upgrade their confidence levels in an incoming information source when the source provides two subsequent pieces of evidence pointing in the same direction and downgrade their confidence levels when the sources provides two subsequent pieces of evidence printing to opposite directions.

This finding supports our first experimental hypothesis stating that measures taken in subsequent planning cycles are interdependent. Confidence measures are a case in point. A one-cycle backtracking is necessary to be able to predict the level of confidence in information source expressed by the commanders in the present cycle.

Observation 5: The commanders' confidence in their information sources is remarkably stable.

The previous finding, although statistically significant, seems to hide the fact that fluctuations in confidence levels are rather weak. A 10% variation seems to represent an upper bound. Why is that? It appears that an across-conditions, across-subjects confidence level average is 66.9% with a standard deviation of only 6.1%. Almost independently of the reliability of the source or the strength of the current belief, subjects seem to assign an approximate 67% confidence level in the incoming information. In other words, for a binary hypothesis testing process, subjects assign a subjective likelihood ratio of about 2.1 that the incoming information is true. As surprising as it may be, this result is supported by previous studies of sequential information processing (Lopes, 1981; Ward, 1983). We will use this ratio as a baseline parameter in the modeling effort described in the next section.

Observation 6: Strength of belief is weakly related to the confidence in the incoming information.

We tested a more restrictive version of our second hypothesis: process measures (i.e. probability or strength of beliefs) are related to the commander's level of confidence in his information sources. This hypothesis can only be confirmed at the initial confidence level. A relatively low confidence level (55.2%) produced a relatively low average probability (61.2%) and a relatively high confidence level (73.6%) induced a higher average probability (74.4%). In subsequent stages this initial effect seems to fade away. This is due to a strong recency effect and to the relative stability (67.7% ± 6.1%) of the confidence level. A more rigorous explanation will be given in the next section.

PLANNING RESULTS

This subsection focuses on the experimental findings obtained in observing the planning portion of the commander's decisionmaking process. The planning decisions in this experiment are defined as the commitment of BLUE units to face a potential attack from the enemy forces. This commitment can take the form of a geographical unit movement, a unit reassignment to a different geographical command (without actual physical motion), or commitment for future actions. Recall that in this paper-and-pencil experiment, all planning decisions occur at the pre-engagement stage and therefore are not implemented in the field.

The main difficulty for the analyst was to find a way to quantify the written protocols the commanders produced at each planning stage. These were sometimes long and elaborate fragmentation orders (FRAGOS) — orders that did not follow specific, easily analyzable formats. They were nevertheless done in good faith, with real professionalism and are at present being used by the authors for future analysis. Table 4 shows two examples of such written protocols for different planning sequences.

To be able to have a basis for data analysis, we decided to assign a weight to each type of unit commitment.

- 1 point for actual movement of one unit
- 0.25 point for a reassignment of one unit to a different command
- 0.25 point for a future commitment of one unit

The data were then tabulated for each subject and each experimental condition and three scores were produced at each stage:

- ΔH : weighted number of units committed in the direction of the currently held hypothesis.
- ΔH : weighted number of units committed in the direction of the opposite hypothesis.
- ΔN : weighted number of units committed in the direction of the neutral direction (central command or division command).

TABLE 4. EXAMPLES OF WRITTEN PROTOCOL

Example A:

Frago #15-2

"Intelligence has shown the units in the west to be less than initially reported. Several large units located in the east now. Task organizations no change.

Continue to plan for main effort in the eastern sector now. 4-7 and 3-7 will attack to 2d Bde on order.

2-4 CAV. move to forward positions to support the 2d Bde.

Expect all % missions to be executed on next Frago.

I am waiting for one final Inf surv. before the final decision is made on attachment.

Cdr 2d Bde.

I expect you to hold the ground above the MDL and not need any further reinforcement."

Example B:

Frago #11-2

"Intelligence has been unable to confirm the presence of units in the southern sector and has confirmed the presence of units in northern sector. Therefore, we will assume that the effort will move in the north. Task Organization changes as follows:

2-7 Atk Helo DS to 1st Bde in 2 hours

1-4 CAV stays under division Control

2-4 CAV still under division control. Move to 1st Bde sector and link up unit 1-4 CAV.

4-7 Asslt Hel Bn move to positions where you can be quickly committed with 1st Bde.

%DS to 1st Bde.

Although intel appears to be finishing up the last InSurv, we have been fooled before. Be prepared to shift emphasis back to the north if that does occur (Remain Flexible)."

The following two observations were derived from empirical findings associated with the planning process:

Observation 7. Commanders tend to concentrate their planning activity in the initial planning stage, with little changes in the subsequent stages even in light of conflicting new evidence.

If we measure the total amount of unit movement and assignment at each stage irrespective of the direction of the troop commitment, we can construct a table (Table 5) representing the total quantity of activity at each stage for different sequences of incoming evidence. The pattern is quite significant ($p < 0.02$). most of the planning activity is concentrated in the first stage.

TABLE 5. AVERAGE NUMBER OF BLUE UNITS MOVED OR REASSIGNED AT EACH STEP OF THE PLANNING PROCESS

Order \ Step	1	2	3
CCD	5.9	0.5	1.6
CDD	4.5	0.5	1.4
DCC	4.8	1.9	1.0
DDC	4.6	0.2	1.6
Mean	5.0	0.8	1.4

This seems to represent a primacy-oriented planning behavior different from the one observed in the situation assessment part where recency was the central factor. It seems that the experienced commanders are quite willing to modify their opinions or beliefs, sometimes drastically, in the light of new evidence but are reluctant to move from the position they anchored in the initial stage [13]. Their elaborate mental model of the battle field probably includes the high cost of moving troops around as opposed to the relatively low cost changing a belief in a hypothesis. In other words, they recognize that the dynamics of action is different from the dynamics of information. This very interesting hypothesis should be further tested in realistic headquarters environments.

Observation 8: Commanders do not commit themselves to a single hypothesis but rather hedge against future uncertainties.

One can observe that most (82%) of the commitment of units follow one pattern: a movement or reassignment of units from the neutral area (central or division command) towards the two likely avenues of enemy attack, with most of the units assigned to the most likely avenue of attack. Figure 9 illustrates this finding.

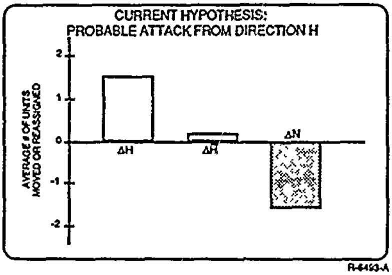


Figure 9. Representative Decomposition of a Commander's Planning Decision.

SUMMARY OF RESULTS

Our main findings demonstrate that even in the simplest C^2 environment, situation assessment and planning are dynamic processes. As indicated by Fig. 3, dependent variables of one cycle become the independent variables of the next and the derived measures are closely interrelated. Our experimental findings indicate that the order of the new incoming information has a strong effect on both processes. This order effect produces a recency-type behavior for the situation assessment process (with a higher sensitivity to negative evidence) and a primary-type behavior for the planning process. It was determined that the commander's confidence level in the incoming information is an essential variable in this decision process. Moreover, a surprising observation is that this confidence level tends to stabilize around the value of 67% (equivalent to a likelihood ratio of 2:1 for incoming evidence) regardless of the context of the information. The next section will build on these findings to develop models of the situation assessment process.

V. MODELING RESULTS

MODELING FRAMEWORK

This section focuses on the situation assessment process. We propose three candidate models to explain and predict the subjects' sequential belief updating process. We chose to model this aspect of the commanders' decision process for the following reasons. First, as explained in Section 4, the data collected for the planning behavior was quite difficult to quantify and, at this time, no simple, quantifiable models exist to describe dynamic planning processes. Comprehensive models would have to take into account complex Lanchester-type equations to quantify variations in tactical effectiveness due to the relative movement of military units. This approach is obviously beyond the scope of this study. On the other hand, the situation assessment/judgment data is more amenable to quantitative modeling. Moreover, the sequential updating of beliefs is an essential component of the commander's decisions. The process by which new evidence from multiple sources is integrated with current hypotheses and belief has been the focus of various recent studies in applied behavioral psychology ([13], [17], [20]). The ability of a commander to generate and compare alternate hypotheses concerning enemy actions is a central element in the HEAT hexagon ("understand" node) and the SHOR paradigm ("H" for hypothesis generation and evaluation). See Figs. 1B and 1C for details.

The results pertaining to the situation assessment process described in section 4 indicate rather complex phenomena. For example, the drastic effect that the order of incoming evidence had on the direction-of-attack assessment must be accounted for in a realistic model. We propose three models: a descriptive anchoring-and-adjustment model developed by Einhorn and Hogarth, called the contrast-inertia model, a normative Bayesian model, and a normative-descriptive modified Bayesian model.

DESCRIPTIVE CONTRAST-INERTIA MODEL

This model, a variation on Einhorn and Hogarth's contrast-inertia model, is based on three principles concerning the way information is processed. It assumes an anchoring-and-adjustment process, the evaluation of a single hypothesis (or an exclusive binary hypothesis, e.g., North-South) in the light of confirmatory (positive) or disconfirmatory (negative) evidence and the existence of a conflict between forces of adaptation and inertia. From Einhorn and Hogarth, 1987.

"Adaptation is modeled by assuming a contrast effect between the level of current opinion and the nature of evidence. Specifically, the larger current opinion, the more it is discounted by negative evidence and the less it is increased by positive evidence. Inertia is assumed to take two forms. One is a level of constant inertia that reflects attitudes toward negative and positive evidence. The other is the tendency for inertia to increase across time."

We use the following notation:

p_i : probability (degree of belief) at stage i that the attack will come from the initially assumed direction

p_{i-1} : probability (degree of belief) at stage i-1 that the attack will come from the initially assumed direction

p_o : initial degree of belief in the initial hypothesis

α : sensitivity to disconfirmatory evidence $0 \leq \alpha \leq 1$

β : sensitivity to confirmatory evidence $0 \leq \beta \leq 1$

C_i : subjective confidence in the i^{th} piece of evidence

The general form of the model is therefore:

$$p_i = \text{function}\ (p_{i-1},\ \alpha \text{ or } \beta,\ C_i) \tag{1}$$

Two forms of the model distinguish the processes for dealing with disconfirmatory and confirmatory evidence. Evidence is subject to contrast effects in that its impact reflects prior position. Specifically, the model supports the hypothesis that negative evidence will be discounted more when the preceding degree of belief is large, whereas positive evidence will lead to greater upward revision of belief when the degree of belief is small. The same negative evidence will induce greater discounting when prior opinion is high as opposed to low, and the same positive evidence will have more impact on belief when prior opinion is low as opposed to high. These contrast effects can be modeled by setting the adjustment weights for negative and positive evidence proportional to, respectively, the prior degree of belief and its complement. Thus,

$$p_i = p_{i-1} - \alpha \cdot C_i \cdot p_{i-1} \text{ if the } i^{th} \text{ piece of evidence is disconfirmatory} \tag{2}$$

and

$$p_i = p_{i-1} + \beta \cdot C_i \ \ (1-p_{i-1}) \text{ if the } i^{th} \text{ piece of evidence is confirmatory} \tag{3}$$

The force toward inertia can be captured in the above formulation by considering the meaning of the constants of proportionality, α and β. We define α and β to represent sensitivity toward negative and positive evidence, respectively. Small values of α and β imply low sensitivity to new information; large values of α and β imply high sensitivity. α and β model the force toward inertia by providing a mechanism for dampening the contrast effect.

The contrast-inertia model provides an elegant mechanism to predict the order effects evident in the experimental data. Why did information processed last in a mixed sequence of information updates (e.g., CDC) have more effect on final opinion than information encountered earlier? The answer can be easily proven in a mathematical manner by exercising the Eqs. 2 and 3. The model always predicts recency effects in the sequential evaluation of mixed evidence, a fact that was observed in all our data gathering trials.

In terms of overall performance the model predicted the human data for all four sequences very well. For reasons explained earlier we assumed an average C_i of 0.67. The best values for the sensitivity coefficients α and β were chosen to be 0.48 and 0.38 respectively. Note that the sensitivity to negative evidence, α, is larger than the sensitivity to positive evidence β. For this set of values, the model's predictions of updated probabilities stayed within 11.2% of the subject's average. This is a remarkable fit if one takes into account the relatively large inter-subject variability (20.4%). As an example, Fig. 10 shows the contrast-inertia model's predictions as compared to the subjects' data and the Bayesian model for the CDD sequence of incoming evidence.

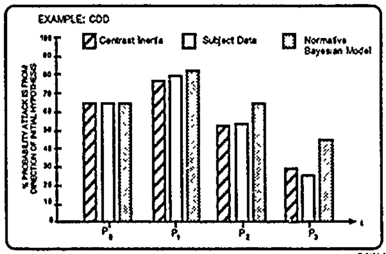


Figure 10. Models - Data Comparison for the CDD Sequence.

NORMATIVE BAYESIAN MODEL

The Bayesian framework is a normative model for the processes of sequential inferences, hypothesis updating or revision of beliefs. It provides a means to combine new information with an existing hypothesis and produces the resulting hypothesis in an optimal fashion. The literature is full of examples showing that humans are not truly "Bayesian" and are affected by various limitations and biases that prevent them to integrate information in an optimal fashion ([15], [16]). Nevertheless, Bayesian models provide a frame of reference for comparing actual human decisions with optimal behavior. According to Bayes theorem:

$$p(H / \inf_i) = \frac{p(\inf_i / H) \cdot p(H)}{p(\inf_i / H) \cdot p(H) + p(\inf_i / \overline{H}) \cdot p(\overline{H})} \tag{4}$$

where

H: initial hypothesis (a belief) (e.g. Northern attack)

$\bar{H}$: opposite hypothesis (e.g. Southern attack)

Since we are dealing with mutually exclusive binary hypotheses:

$$p(\bar{H}) = 1 - p(H) \tag{5}$$

p(H) = probability that hypothesis H is true before receiving the i^{th} piece of evidence (prior strength of belief);

= p_{i-1}

$p(H/inf_i)$ = probability that hypothesis H is true after receiving the i^{th} piece of evidence (posterior strength of belief);

= p_i

Therefore we can write Eq. 4 in its simplified binary form:

$$p_i = \frac{p_{i-1}}{p_{i-1}\,LR_i + (1 - p_{i-1})} \tag{6}$$

Where LR_i is the likelihood ratio of the incoming i^{th} piece of evidence:

$$LR_i = \frac{p(inf_i / H)}{p(inf_i / \bar{H})} \tag{7}$$

From Eq. 6, one notices that the Bayesian formulation, unlike the previous contrast-inertia model does not allow for recency factors that could produce order effects. The Bayes formula is a batch processor. At the end of a sequence of mixed evidence, only the amount and quality of information that went in is important, not the order. A graphical way to represent Eq. 6 is shown in Fig. 11. For example, one should upgrade a prior probability of 0.6 to a posterior probability of 0.75 if a piece of evidence confirming the hypothesis by a ratio of 2:1 is received.

How well did the Bayesian model predict the subjects' data? For reasons mentioned before and for comparability with the contrast-inertia model, we fixed the likelihood ratio to 2:1 for confirming evidence and 1:2 for disconfirming evidence. This reflects the empirical confidence level of 0.67 found in section 4. The Bayesian predictions were, in average, within 25% of the subjects' averaged data (as compared to 11.2% for the contrast-inertia model). The Bayesian model showed a "reluctance" to downgrade probabilities in the presence of negative evidence, while the subjects and the contrast-inertia model were oversensitive to it. Fig. 10 showed an example of the performance of the Bayesian model. The next subsection will show how to improve this prediction performance by modifying the Bayesian model.

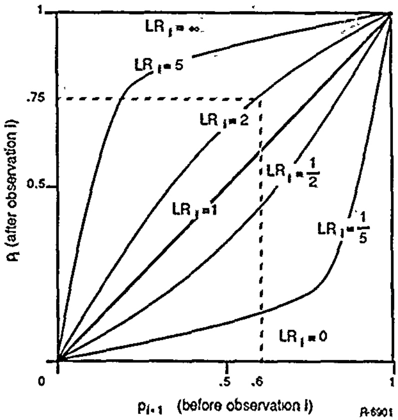


Figure 11. Bayesian Updating of Beliefs.

NORMATIVE-DESCRIPTIVE MODEL

This modeling approach follows principles described in [19] and is referred to as normative-descriptive. To begin, a normative model is developed, which in this case is a purely sequential Bayesian scheme, to optimally determine the probabilities (p_i). The normative model makes these determinations based on evidence derived from the decision environment (intelligence messages, maps, etc.). Following the formal delineation of the normative scheme, qualitative descriptive factors deduced from the statistical analyses of the subjects' strength of belief data are introduced into the normative formulation. The addition of these descriptive elements is designed to corrupt the normative model so it becomes more descriptive of the human subjects' data. The addition of the psychologically relevant descriptive factors to the normative formulation brings about the normative-descriptive model. The parameters of this normative-descriptive model are tuned until the best possible fit between the model's output and the subjects' data is achieved. The development of a normative-descriptive model (if it can be achieved) offers several advantages. Such a model yields a predictive tool of the subjects' decisionmaking. One of the most interesting and important advantages provided by the normative-descriptive model is the insight it provides of the subjects' decision processes.

In our experiment, the most salient descriptive feature is the recency factor that produced an order effect in the sequential version of probability. This recency factor is in part due to the asymmetry in the subjects' reaction to positive and negative evidence. Subjects tended to react to negative evidence more drastically and to positive evidence less drastically than the normative Bayesian model predicted they should (see Fig. 10).

Let us modify the Bayesian formula to reflect this phenomenon Equation 6 can be rewritten as the "odds" version of Bayesian theorem:

$$\frac{p_i}{1-p_i} = \frac{p_{i-1}}{1-p_{i-1}} \cdot LR_i \qquad (8)$$

Following Edwards et al [10], a normative-descriptive version of Eq. 8 is:

$$\frac{p_i}{1-p_i} = \frac{p_{i-1}}{1-p_{i-1}} \cdot (LR_i)^{\gamma}$$

Where

$0 < \gamma < 1$ for positive evidence $(LR_i > 1)$

$\gamma > 1$ for negative evidence $(LR_i < 1)$

This manipulation reduces the effect of positive evidence while enhancing the effect of negative evidence, thus reflecting human-like behavior. The values of $\gamma = 0.9$ for positive evidence and $\gamma = 1.3$ for negative evidence achieved the best data-matching scores. The resulting normative-descriptive model predicted the data with an average error of only 13.1%, comparable with the contrast-inertia model performance. Note that both models present on asymetry of the subjects sensitivity towards negative evidence.

SUMMARY

Different models were developed to predict and explain the situation assessment data gathered in the experiment. The descriptive contrast-inertia and the normative-descriptive modified Bayesian models predicted the data remarkably well. Both models included in an explicit fashion descriptive elements that could explain the recency phenomena and the resulting order effects due to an asymmetry in the perception of confirmatory and disconfirmatory evidence.

VI. CONCLUSIONS AND RECOMMENDATIONS

This study started with the premise that some basic characteristics of a headquarters planning process could be replicated in a focused experiment. By using a static (paper and pencil) task environment rather than a dynamic wargaming computer-based scenario, a small sample of single experienced commanders rather than hierarchies of commanders and their subordinates, and model-based experimental methods rather than exclusively empirical evaluation, we were hoping to be able to draw conclusions and formulate hypotheses for future larger-scale experiments and exercises. In other words we were asking the question: could such focused cost-effective mini-experiments be used as "rapid prototypes" to gain insight into the HEAT cycle process? We believe the present effort has shown that the answer to this question is positive. Three HEAT-related hypotheses were tested in this study.

The first hypothesis was confirmed. Process measures such as strength of beliefs or troop activity were found to be strongly interdependent from cycle to cycle. Empirical evidence as well as predictive modeling showed how the very nature of C^2 decisions is dynamic. For example the strength of one's current belief that one's hypothesis is true is a function of past beliefs, new evidence and the history of the information source providing this evidence.

The second hypothesis was only partially confirmed for some instances of the situation assessment process. Although all the models tested predicted that the commanders estimate of the situation should depend on his confidence on the information sources, we were not able to show a clear dependency with their understanding of the battle. More research will have to be done in that direction by investigating the planning activity as a function of a commander's current beliefs and his/her confidence in the understanding of the battle. Preliminary empirical data, shown in Fig. 12 indicated that this function may be a complex one.

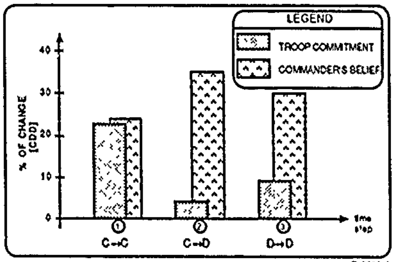


Figure 12. An Example of the Potential Complex Relationship Between Situation Assessment and Planning Processes.

The third and final hypothesis was strongly confirmed The order in which information was presented to the commanders had a very significant effect on their situation assessment behavior (recency-oriented) and their planning behavior (primacy-oriented). Two contract-inertia descriptive model and a modified Bayesian normative-descriptive model were able to predict the commander's situation assessment behavior remarkably well.

In conclusion, the first and third hypothesis provided strong support to our proposed view of the headquarters planning process as "rolling". The rolling plan concept introduced in Section 1 helps put the headquarters' decision processes in a dynamic perspective, where equivalent attention is being given not only to the amount and quality of the incoming evidence but also to the sequencing of that evidence. Training commanders, for a more balanced way of integrating this information, may produce more effective command and control planning decisions

REFERENCES

1. Lawson, J.S., "Command and Control as a Process," IEEE Control Systems Magazine, March 1981, pp. 5-12.

2. Lawson, J S, "The Role of Time in a Command and Control System," Proceedings of the Fourth MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by Command-Control-Communications Problems: Volume IV, Laboratory for Information and Decision Systems, LIDS-R-1159, MIT, Cambridge, Massachusetts, pp. 20-59.

3. Wohl, J.G., "Force-Management Decision Requirements for Air Force Tactical Command and Control," IEEE Transactions on Systems, Man, and Cybernetics, Vol. SMC-11, No. 9, September 1981, pp. 618-639.

4. Wohl, J.G., E.E. Entin, D.L. Kleinman, and K. Pattipati, "Human Decision Processes in Military Command and Control," in Advances in Man-Machine Systems Research, Volume 1 (A. Sage, Ed.), JAI Press, 1984, pp. 261-307.

5. Defense Systems, Inc., "Theatre Headquarters Effectiveness: Its Measurement and Relationship to Size, Structure, Functions, and Linkages, Volume I," McLean, Virginia, December, 1982.

6. Cushman, John H., Personal Communication, 1988.

7. Wohl, J.G., "Rate of Change of Uncertainty as an Indicator of Effectiveness of Command and Control Systems Effectiveness," Proceedings of the Fourth MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by Command-Control-Communications Problems: Volume IV, Laboratory for Information and Decision Systems, LIDS-R-1159, MIT, Cambridge, Massachusetts, pp. 1-17.

8. "The Instruction of Frederick the Great for his Generals," in Roots of Strategy, Brig. Gen. T.R. Phillips (Ed.), Stackpole Books, Harrisburg, Pennsylvania, 1985, pp. 301-400.

9. Bertsekas, D.P., Dynamic Programming: Deterministic and Stochastic Models, Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1987.

10. Edwards, W., and L D. Phillips, "Man as Transducer for Probabilities in Bayesian Command and Control Systems," in M.W. Shelly, II, and G L. Bryan (Eds.), Human Judgments and Optimality, New York, Wiley, 1964, pp. 360-401.

11. Peterson, C.R., and L.R. Beach, "Man as an Intuitive Statistician, "Psychological Bulletin, 69, 1967, pp. 29-46.

12. Donnel, M.L, and W.M. Ducharme, "The Effect of Bayesian Feedback on Learning in an Odds Estimation Tasks," Organizational Behavior and Human Performance, 14, 1975, pp. 305-313.

13. Tolcott, M.A., F.F. Marvin, and P.E. Lehner, "Effects of Early Decisions on Later Judgments in an Evolving Situation," Decision Science Consortium, Inc., Technical Report No. 87-10, Falls Church, Virginia, 1987.

14. Wason, P.C., "On the Failure to Eliminate Hypotheses in a Conceptual Task," Quarterly Journal of Experimental Psychology, 1960, 12, pp. 129-140.

15. Einhorn, H.J., and R.M. Hogarth, "Behavioral Decision Theory: Processes of Judgment and Choice. Annual Review of Psychology, 1981, 32, 53-88.

16. Lopes, L.L., "Averaging Rules and Adjustment Processes: The Role of Averaging in Inference" (T.R.), Madison, Wisconsin: Wisconsin Human Information Processing Program (WHIPP 13), 1981.

17. Einhorn, H.J., and R.M. Hogarth, "Adaptation and Inertia in Belief Updating: The Contrast-Inertia Model," Technical Report, Center for Decision Research, University of Chicago, Illinois, 1987.

18. Entin, E.E, R.M. James, D. Serfaty, and J. Forester, "The Effects of Cognitive Style and Prior Information on Multi-Stage Decisionmaking," Technical Report Number TR-277-1, ALPHATECH, Inc., Burlington, Massachusetts, 1987.

19. Bushnell, L., D. Serfaty, and D.L. Kleinman, "Team Information Processing: A Normative-Descriptive Approach", in The Science of Command and Control, Johnson and Levis, Eds, AFCEA International Press, 1988.

20. Anderson, N. H., Foundation of Information Integration Theory, Academic Press, 1981.

COMMAND AND CONTROL EXPERIMENT DESIGN USING DIMENSIONAL ANALYSIS *

Victoria Y. Jin
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

Dimensional analysis is a method used in the design and analysis of experiments in the physical and engineering sciences. When a functional relation between variables is hypothesized, dimensional analysis can be used to check the completeness of the relation and to reduce the number of experimental variables. The approach is extended to include dimensions pertinent to experiments containing cognitive aspects so that it can be used in the design of multi-person experiments. The proposed extension is demonstrated by applying it to a single decisionmaker experiment already completed. New results from that experiment are described.

INTRODUCTION

In the last few years, a mathematical theory for the analysis and design of organizations supported by Command, Control, and Communications (C3) systems has been developed based on the model of interacting human decisionmakers (DMs) with bounded rationality [1], [2]. While this model was motivated by empirical evidence from a variety of experiments, and by the concept of bounded rationality [3], there were no direct experimental data to support it. An experimental program has been undertaken to test the theory and obtain values for the model parameters [4].

One of the major difficulties in developing a model-driven experimental program is the specification of the large number of parameters that have to be specified and varied. The resulting problem has two aspects: (a) The parameterization of the experimental conditions leads to a very large number of trials, a situation that is not really feasible when human subjects are to be used, and (b) Not all experimental variables can be set at the values required by the experimental design because of the lack of direct controls on the cognitive variables.

Consequently, some orderly procedure is needed that will allow the reduction of the number of experimental variables and, more importantly, that will lead to variables that are easier to manipulate. Such an approach, called dimensional analysis, has been in use in the physical and engineering sciences [5], [6].

*This work was carried out at the MIT Laboratory for Information and Decision Systems with support by the Office of Naval Research under contract No. N00014-84-K-0519 (NR 649 003).

The purpose of this paper is to extend the approach to problems that have cognitive aspects so that it can be used for the design and analysis of experiments. The class of problems we are interested in are those that relate organizational structure directly to performance, as measured by accuracy and timeliness, and, more indirectly, to cognitive workload.

A special class of organizations will be considered a team of well-trained decisionmakers executing repetitively a set of well-defined cognitive tasks under severe time pressure. The cognitive limitations of decisionmakers imposes a constraint on the organizational performance. Performance, in this case, is assumed to depend mainly on the time available to perform a task and on the cognitive workload associated with the task. When the time available to perform a task is very short (time pressure is very high), decisionmakers are likely to make mistakes so that performance will degrade.

This class of organizations is a reasonable model for tactical distributed decisionmaking such as that in the Command Information Center (CIC) of a battle group, a team of well trained individuals receive information from a variety of sources, process the information to develop the situation assessment, generate courses of action (COA), select a COA, and produce the set of commands or orders that will implement the chosen COA.

Dimensional analysis will be introduced briefly in the next section. The approach is then extended to include cognitive variables and a completed experiment will be used as an example to demonstrate the approach Then, the application of dimensional analysis to the design of experiments for the analysis and evaluation of distributed tactical decisionmaking organizations will be described.

DIMENSIONAL ANALYSIS

Dimensional analysis is a method for reducing the number and complexity of experimental variables which affect a given physical phenomenon. A detailed introduction to dimensional analysis can be found in [5], [6].

Dimensions and Units. A dimension is the measure which expresses a physical variable qualitatively. A unit is a particular way to express a physical quantity, that is, to relate a number to a dimension. The dimension of a physical variable exists independently of the units in which it is measured For example, length is a dimension associated to physical quantities such as distance, height, depth, etc , while foot, meter, are different units for expressing length

Fundamental Dimensions. Fundamental dimensions are the basic dimensions which characterizes all variables in a physical system For example, length, mass, and time are fundamental dimensions in mechanical systems. A dimension such as length per time is a secondary or derived dimension.

Dimensionally independent variables. If the dimension of a physical variable cannot be expressed by the dimensions of others in the same equation, this dimension is independent. For example, distance, velocity and time are three physical quantities which are not dimensionally independent because the dimensions of any two variables can form the dimension of the third. They are, however, pair-wise dimensionally independent.

The foundation of dimensional analysis is the Principle of Dimensional Homogeneity, which states that if an equation truly describes a physical phenomenon, it must be dimensionally homogeneous, i e., each of its additive terms should have the same dimension.

For example, consider a moving vehicle with initial velocity v_0 and constant acceleration a. During time t, the distance traveled s can be described by the following equation:

$$s = v_0 t + at^2/2 \qquad (1)$$

where s has dimension of length, v_0 has dimension of length per unit time, t has dimension of time, a has dimension of length per unit time per unit time, and the constant 1/2 is a pure number which has no dimension. Expressing the terms of this equation dimensionally, we obtain:

$$[s] = L$$
$$[v_0 t] = LT^{-1}T = L$$
$$[at^2/2] = LT^{-2}T^2 = L$$

This shows all additive terms have dimension of length, therefore, Eq. 1 is dimensionally homogeneous.

The basic theorem of dimensional analysis is the π theorem, also called Buckingham's theorem.

π **theorem:** If a physical process is described by a dimensionally homogeneous relation involving n dimensional variables, such as

$$x_1 = f(x_2, x_3, \ldots, x_n) \qquad (2)$$

then there exists an equivalent relation involving (n-k) dimensionless variables, such as

$$\pi_1 = F(\pi_2, \pi_3, \ldots, \pi_{n-k}) \qquad (3)$$

where k is usually equal to, but never greater than, the number of fundamental dimensions involved in the x's.

Each of the π's in Eq. 3 is formed by combining (k+1) x's to form dimensionless variables. Comparing Eqs. 2 and 3, it is clear that the number of independent variables is reduced by k, where k is the maximum number of dimensionally independent variables in the relation The proof of the π theorem can be found in [5].

The π theorem provides a more efficient way to organize and manage the variables in a specific problem and guarantees a reduction of the number of independent variables in a relation Dimensionless variables, also called dimensionless groups, are formed by grouping primary variables with each one of the secondary variables. The procedure for applying dimensional analysis will be described now through an example:

Step 1 Write a dimensional expression.

Let the dependent physical variable be denoted by q and the set of independent variables on which q depends be represented by w, x, y, and z. Since all the variables represent physical quantities, the have appropriate dimensions.

Then, a dimensional expression can be written as

$$q = f(w, x, y, z) \qquad (4)$$

There are five dimensional variables in Eq. 4, that is, n = 5.

Step 2 Determine the number of dimensionless groups.

To illustrate this step, a physical system and real physical quantities have to be assumed. Assume q is energy, w is time, x is a mass, y is acceleration, and z is distance in some mechanical system. One set of fundamental dimensions of a mechanical system are mass (M), length (L), and time (T), i.e., there are three dimensionally independent variables, k = 3. The dimensions of the variables in Eq. 4 are shown in Table1.

TABLE 1 Dimensions of variables in Eq. 4

Variable		Dimension	Notation
energy	q:	force × length	$[q] = ML^2T^{-2}$
time	w:	time	$[w] = T;$
mass	x:	mass	$[x] = M;$
acceleration	y:	length per time per time	$[y] = LT^{-2};$
distance	z:	length	$[z] = L$

Since n = 5 from Step 1, there are,

$$n - k = 5 - 3 = 2,$$

so that three primary variables should be selected and two dimensionless groups can be constructed.

Step 3 Construct dimensionless groups.

While the choice of primary variables is essentially arbitrary, consideration should be given that the dimensionless groups be meaningful. If w, x, y are chosen as the three (k = 3) primary variables, two dimensionless groups are constructed on the basis of the remaining variables q and z. The first dimensionless group π_1 is formed by the combination of q, w, x, and y Using the power-product method, π_1 can be determined by the following procedure. Write π_1 as

$$\pi_1 = q^a w^b x^c y^d$$

where a, b, c, and d are constants which make the right hand side of the equation dimensionless so that the equation is dimensionally homogeneous. In terms of dimensions of q, w, x, and y, we have

$$[M^0 L^0 T^0] = [ML^2T^{-2}]^a[T]^b[M]^c[LT^{-2}]^d$$
$$= M^{a+c} L^{2a+d} T^{-2a+b-2d}$$

By the Principle of Dimensional Homogeneity, the following set of simultaneous algebraic equations must be satisfied.

For M: $a + c = 0$
For L: $-2a + b - 2d = 0$
For T: $2a + d = 0$

There are three equations but four unknowns. The solution is not unique. In general, it is convenient for the secondary variables, in this example q and z, to appear in the first power, that is, a is set equal to unity. Thus, by solving the set of algebraic equations, we obtain:

$a = 1,\ b = -2,$
$c = -1,\ d = -2.$

then

$$\pi_1 = q / (w^2xy^2).$$

Similarly,

$$\pi_2 = zw^2 / y.$$

The dimensionless form of Eq. 4 is

$$q / (w^2xy^2) = \Psi(zw^2/y),$$

or in terms of the dimensionless groups,

$$\pi_1 = \Psi(\pi_2) \qquad (5)$$

This is the result obtained by the application of dimensional analysis. The function Ψ is unknown and needs to be determined by experiments. The dimensional analysis reduces Equation 4, which has four (4) independent dimensional variables, to Equation 5 which has only one independent dimensionless variable. The complexity of the equation is reduced dramatically. Furthermore, in designing an experiment, it is only necessary to specify a sequence of values for the independent variable π_2, these values can be achieved by many combinations of w, y, and z.

APPLICATION OF DIMENSIONAL ANALYSIS TO PROBLEMS IN COMMAND AND CONTROL

To apply dimensional analysis to decisionmaking organizations, the fundamental dimensions of the variables that describe their behavior must be determined. A system of three dimensions is shown in Table 2 that is considered adequate for modeling cognitive workload and bounded rationality. An experiment conducted in 1987 [4] is used to demonstrate the application of dimensional analysis to Command and Control problems. The purpose of the single-person experiment was to investigate the bounded rationality constraint. The experimental task was to select the smallest ratio from a sequence of comparisons of ratios consisting of two two-digit integers. Two ratios were presented to a subject at each time. The subject needed to decide the smaller one and compare it with the next incoming ratio until all ratios were compared and the smallest one was found. The controlled variable (or manipulated variable) was the amount of time allowed to perform the task. The measured variable was the accuracy of the response, i.e., whether the correct ratio was selected.

TABLE 2. DIMENSIONS FOR C^2 PROBLEMS

Dimension	Symbol	Units
Time	T	sec
Information (uncertainty)	I	bit
Alphabet	S	symbol

The controlled variables were the number of comparisons in a sequence, denoted by N, and the allotted time to do the task, denoted by T_w. For each value of N, where N could take the value of 3 or 6, T_w took twelve values with constant increment in the following way:

$$T_w = \{ 2.25, 3, 3.75, ..., 10.5 \} \qquad \text{for } N = 3;$$
$$T_w = \{ 4.50, 6, 7.50, ..., 20.1 \} \qquad \text{for } N = 6.$$

The performance was considered to be accurate or correct if the sequence of comparisons was completed and if the smallest ratio selected was correct. The details of the experiment can be found in [4].

The hypothesis is that there exists a maximum processing rate for human decision makers. When the allotted time is decreased, there will be a time beyond which the time spent doing the task will have to be reduced if the execution of the task is to be completed. This will result in an increase in the information processing rate F, if the workload is kept constant. However, the bounded rationality constraint limits the increase of F to a maximum value F_{max}. When the allotted time for a particular task becomes so small that the processing rate reaches F_{max}, further decrease of the allotted time will cause performance to degrade. The performance drops either because all comparisons were not made or because errors were made.. It was hypothesized that the bounded rationality constraint F_{max} is constant for each individual DM, but varies from individual to individual. The bounded rationality constraint can be expressed as

$$F_{max} = G / T_w^* \qquad (6)$$

where T_w^* is the minimum allotted time before performance degrades significantly. G and T_w^* vary for different tasks, but F_{max} is constant for a decision maker, no matter what kind of tasks he does. Therefore, significant degradation of performance indicates that the allotted time approaches T_w^*. Observation of this degradation during the experiment allows the determination of the time threshold and, therefore, the maximum processing rate, provided the workload associated with a specific task can be estimated or calculated [4].

The retroactive application of dimensional analysis to this experiment will be shown step by step.

Step 1 Write a dimensional expression.

In the experiment, accuracy, J, of information processing and decisionmaking is defined as the number of correct decisions, that is, the number of correct results in a sequence of comparisons. Therefore, J has the dimension of symbol and depends on the following variables:

- N: number of comparisons in each trial;
- T_w: allotted time to do N comparisons;
- H: uncertainty of input, that is, the uncertainty of the ratios to be compared in a trial;

Then, the dimensional expression is

$$J = f(T_w, N, H) \qquad (7)$$

First, dimensional analysis checks whether this functional relation could describe the relation between J and other variables. The dimensions of the variables in Eq. 7 are the following:

$$[J] = S$$
$$[T_w] = T$$
$$[N] = S$$
$$[H] = I$$

Since the dimension of J is S, the right hand side of Eq. 7 has to be of the same dimension regardless of what the functional relation f is. However, all three fundamental dimensions are represented by the three independent variables. There is no way to combine these variables to obtain a term of dimension S only. Therefore, according to Principle of Dimensional Homogeneity, this functional relation is not a correct expression of the relation under the inv. tigation.

There are two approaches to obtain the correct relation. The first is to delete T_w and H from the relation. This is not acceptable because the allotted time is a critical factor in this experiment. The other approach is to add some variables or dimensional constants to satisfy the requirement for dimensional homogeneity. Dimensional constants are physical constant such as gravity, the universal gas constant, and so on. No such dimensional constant has been identified in C^2 systems as yet, therefore, some variables which have dimensions of time and information should be added to the relation. Moreover, the additional variables have to be relevant to the measurement of accuracy. Consideration of the nature of the tasks subjects performed and the data collected led to the observation that the entire allotted time period was not used to process information. This consideration led to a new variable, the actual processing time, T_f. Cognitive workload, denoted by G_a, is another significant variable affecting accuracy. Therefore, two variables are introduced to Eq. 7. The equation describing accuracy becomes

$$J = f(T_w, T_f, N, H, G_a) \qquad (8)$$

This equation is dimensionally homogeneous. There are six dimensional variables in Eq. 8, that is, $n = 6$.

Step 2 Determine the number of dimensionless groups.

The number of dimensionless variables is equal to n-k, where k is the maximum number of dimensionally independent variables in Eq. 8. Dimensions of the variables are

$$[J] = S, \qquad [N] = S$$
$$[T_w] = T, \qquad [T_f] = T$$
$$[H] = I, \qquad [G_a] = I$$

The maximum number of dimensionally independent variables is three. Therefore, k is equal to three. Then, the number of dimensionless groups is

$$n - k = 6 - 3 = 3.$$

There will be three dimensionless groups in the equivalent dimensionless equation.

Step 3 Construct the dimensionless groups.

The selection of primary variables is arbitrary as long as they are dimensionally independent. In this case, T_w, N, and H are selected as the primary variables. Using the power-product method, the π's are found to be

$$\pi_1 = J/N$$
$$\pi_2 = T_f/T_w$$

and

$$\pi_3 = G_a/H.$$

Now, we can write Equation 8 in a dimensionless form

$$J/N = \phi(T_f/T_w, G_a/H) \qquad (9)$$

or, in terms of the π's

$$\pi_1 = \phi(\pi_2, \pi_3) \qquad (10)$$

In Eq. 10, π_1 is the percentage of correct decisions; π_2 indicates that portion of the time window used to process information and make decisions, and π_3 represents the ratio of actual workload and input uncertainty. Equation 10 represents a model driven experiment in which π_1, π_2 and π_3 are the experimental variables to be measured or controlled. The function ϕ needs to be determined experimentally.

Comparing Equations 8 and 10, one finds that the number of independent variables is reduced from six to three. This reduction reduces the complexity of the equation and facilitates experiment design and analysis. Properly designed experiments using dimensional analysis provide similitude of experimental condition for different combinations of dimensional variables which result in the same value of π's. Similitude reduces the number of trials needed to be run in order to define ϕ. This is a major advantage when the physical (dimensional) experimental variables cannot be set at arbitrary values.

The experiment that has been described was not designed using dimensional analysis. The independent variables that were manipulated were not π_2 and π_3. Therefore, ϕ cannot be determined from the experimental data. The purpose of using this experiment is to illustrate the dimensional analysis procedure for the design and analysis of model driven experiments. Therefore, only new results from dimensional analysis will be shown

The model developed by applying dimensional analysis allows for more thorough analysis of the experimental data. In the original experiment, the allotted time was used to find the time threshold which was taken to correspond to the maximum processing rate. However, since the most obvious manipulated variable was the allotted time, the first priority of subjects seemed to be the completion of the comparisons within that time. The results from the experiment a[illegible]fect this observation. The rational expectation that the l[illegible] window would result in better performance does n[illegible] a[illegible]ere. Instead, actual processing time to complete a task [illegible]creasing with increase of the allotted time, but was clos[illegible] constant when the allotted time became larger than a certain value. Knowing the allotted time, subjects tried to finish the task as soon as possible. The experimental data show that in most case, subjects either used a portion of the allotted time to finish the task, or could not finish the task within the allotted time. The ratio of actual processing time and the allotted time is always less than one. Therefore, calculation of the processing rate using allotted time led to underestimating the actual value. The use of the actual processing time leads to a new time threshold that yields a more accurate estimate of the maximum processing rate.

To find the critical value of T_f, the relation between the allotted time T_w and the actual processing time T_f has been studied. Figure 1 shows scatter plots of T_f versus T_w for two subjects.

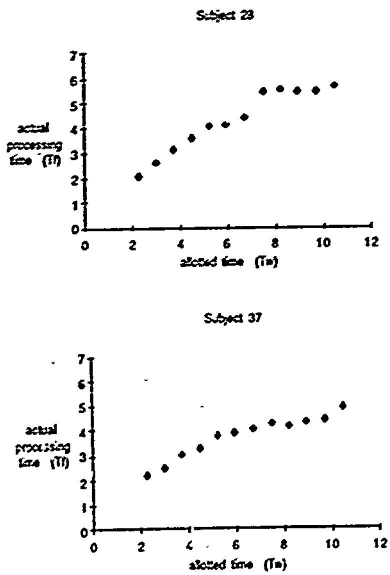


Fig. 1 Scatter Plot of T_f versus T_w for Two Subjects

The study of this relation res[illegible] in postulating the following functional relation between T_f and T_w:

$$T_f = a e^{bT_w} \tag{11}$$

where a and b are constant for each subject and vary among subjects. A least-squares fit was performed to determine the coefficients for each subject. Fig. 2 shows the results of curve fitting for the same two subjects.

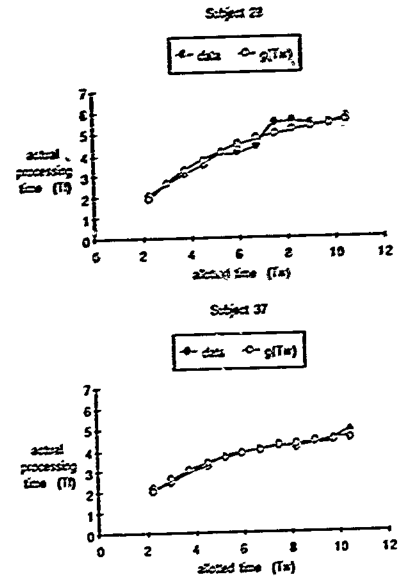


Fig. 2 Exponential Fit for Two Subjects

Since the critical value, T_w^*, of T_w has been found from the original analysis [7], the critical value of T_f, T_f^*, can be calculated using Eq. 11. Table 3 shows the statistics of the time threshold corresponding to T_f and T_w.

TABLE 3 Summary of Time Threshold for T_f and T_w over Subjects

	MEAN	ST. DEV	MAX.	MIN.
T_f^*	4.50	1.12	7.21	2.06
T_w^*	6.38	2.11	9.88	2.73

From Table 3, it is clear that the mean value of T_f^* is smaller than that of T_w^*. The standard deviation of T_f^* is less than that of T_w^* because T_f has to be less than or equal to T_w. Consequently, the calculation of information processing rate using T_f^* gives a higher value because F^* is computed by

$$F^* = G/T^*.$$

For a particular subject, G does not change regardless whether T_f^* or T_w^* is used.

The procedure for designing experiments to study the effect of organizational structure on performance measures, using dimensional analysis is:

1. According to dimensional considerations, determine independent variables which may affect the physical phenomenon, then form a general expression with an unknown function or functions;

2. Apply dimensional analysis to the expression to derive dimensionless groups and check for completeness;

3. Design experiments in which the values of the independent dimensionless groups are manipulated.

4. Run experiments to check the choice of independent variables and determine the hypothesized functional relation.

In the case of tactical decisionmaking organizations supported by C^3 systems, it is assumed that accuracy of a n-DM organization depends on the tempo of operations (which determines the allotted time to perform different tasks) and the cognitive workload of the individual decisionmakers, that is:

$$J = f(T, G^1, G^2, \ldots, G^n) \qquad (12)$$

where J is accuracy, T is a measure of time, and G^i is the workload of the i-th DM. The experimental model is established by augmenting Eq. 12. The measure of time is decomposed into the response times of individual DMs. The number of tasks is considered as a variable which affects accuracy. Uncertainty of the input can be controlled, and will also affect accuracy. For a particular task, cognitive activity varies among human decision makers because each DM may use a different approach to do the task. Let T^i the denote response time of the i-th DM, N denote the number of tasks, and H denote the input uncertainty. Equation 12 becomes

$$J = f(H, N, T^1, T^2, \ldots, T^n, G^1, G^2, \ldots, G^n) \qquad (13)$$

Equation 13 is an experimental model for an organization with n DMs. The unknown function f needs to be determined by experiment. There are (2n+2) independent variables in Eq. 12. Dimensional analysis will be used to reduce the complexity of the equation and organize the variables into groups amenable to manipulation in the context of experiments with human subjects.

CONCLUSIONS

Dimensional analysis has been introduced to the design of experiments that have cognitive aspects. An extension has been presented that makes it possible to include variables such as cognitive workload and bounded rationality of human decision makers. An existing single-person experiment has been used as an example to show how the methodology can be applied. A new result from the existing experiment has been presented to illustrate the possible advantages of using dimensional analysis. Note that dimensional analysis only determines possible relations between relevant variables; the actual functional expression has to be found from experimental data.

REFERENCES

[1] Boettcher, K. L., and A. H. Levis, "Modeling the Interacting Decisionmaker with Bounded Rationality," *IEEE Trans. on Systems, Man, and Cybernetics*, Vol. SMC-12, No. 3, May/June 1982.

[2] Levis, A. H., "Information Processing and Decision-making Organizations: A Mathematical Description," *Large Scale Systems*, Vol. 7, pp. 151-163, 1984.

[3] March, J. G., "Bounded Rationality, Ambiguity, and the Engineering of Choice," *Bell J. Economics*, Vol. 9, pp. 587-608, 1978.

[4] Louvet, A. C., J. T. Casey, and A. H. Levis, "Experimental Investigation of Bounded Rationality Constraint," in *Science of Command and Control*, S. E. Johnson and A. H. Levis, Eds., AFCEA International Press, Washington DC, 1988.

[5] Hunsaker, J. D., and B. G. Rightmire, *Engineering Applications of Fluid Mechanics*, McGraw-Hill, 1947.

[6] Gerhart, P.M., and R.J. Gross, *Fundamentals of Fluid Mechanics*, Addison-Wesley, 1985.

[7] Louvet, A. C., "The Bounded Rationality Constraint: Experimental and Analytical Results," SM Thesis, Report LIDS-TH-1771, MIT, June 1988.

EVALUATION OF EXPERT SYSTEMS IN DECISIONMAKING ORGANIZATIONS*

Didier M. Perdu
Alexander H. Levis

Laboratory for Information and Decision Systems
Massachusetts Institute of Technology
Cambridge, MA 02139

ABSTRACT

A class of decision aids that is receiving attention in the development community is based on artificial intelligence and especially expert systems. This paper presents a procedure for assessing to what extent the measures of performance of an organization are modified when an expert system is introduced. First, a model of symbolic computation with fuzzy logic, using Predicate Transition Nets, is presented to model the most common kind of expert systems: the consultant expert systems. This model allows to evaluate its response time for a given input. An Air Defense problem in which command and control involves a hierarchical two decisionmaker organization, where the expert system is used as an aid in the fusion of inconsistent information, is then presented. A strategy involving the use of the expert system is compared to two other strategies expected to be used by a decisionmaker facing this problem. Measures of performance (workload, timeliness and accuracy) are evaluated for each of these strategies. The results show that the strategy involving the use of the expert system improves significantly the accuracy of the organization, but requires more time and increases the workload of the decisionmaker using it.

INTRODUCTION

Decisionmaking processes require the analysis of complex situations and the planning, initiation and control of subsequent responses. These activities are done within some constraints such as time and accuracy and so that an acceptable level of effectiveness be reached. The amount of information handled by decisionmakers is often very large and, in order to maintain performance above a certain level, decisionmaking organizations use decision support systems to help them accomplish their mission. Among them, Expert Systems with their deductive capability and their ability to handle symbolic concepts have proved to be very useful. The aim of this paper is to show to what extent the use of an expert system modifies the measures of performance of a decisionmaking organization. To allow the use of the analytical framework developed for the study of these organizations, an expert system model using Predicate Transition Nets is first defined for the evaluation of the response time. Expert systems are then studied to assess their usefulness in aiding the fusion of possibly inconsistent information coming from different sources. This assessment is done through the analysis of an application involving a two-decisionmaker organization facing this problem of inconsistent information. Three strategies used to solve this problem are described, one of them involving the use of an expert system. Measures of performance reached for each of these strategies are finally evaluated and compared.

1.0 AN EXPERT SYSTEM MODEL USING PREDICATE TRANSITION NETS

Knowledge Based Expert Systems show properties of synchronicity and concurrency which makes them suitable for being represented with the Predicate Transition Net formalism (Genrich and Lautenbach, 1981). The rules of a knowledge base have to be checked in a specific order depending on the strategy used to solve the problem and on the current facts deduced so far by the system in the execution of previous rules. A model of an expert system using production rules to represent knowledge is presented. Some previous work (Gordona and Saitta, 1985) have addressed the modeling of production rules of a knowledge base using Predicate Transition Nets. The model presented here is different because it incorporates explicitly the control done by the inference engine. Fuzzy logic (Zadeh, 1965 and 1983; Whalen and Schott, 1983) is used to deal with uncertainty and Predicate Transition Nets are used to represent the basic fuzzy logical operators AND, OR and NOT that appear in this kind of rules. An extension of the standard inference net formalism is obtained by the combination of these operators which permits to represent the dynamical behavior of an expert system. The obtained net allows the identification of the rules scanned by the system to produce an answer to a specific problem and to deduce its response time depending on the number of rules scanned and on the number of interactions with the user.

1.1 Structure of the Expert System

Knowledge Based Expert Systems, commonly called Expert Systems, are - in theory - able to reason using an approach similar to the one followed by an expert when he solves a problem within his field of expertise. A net model for the most common kind of expert system, the consultant expert system, as described by Johnson and Keravnov (1985), is proposed. Most systems engage in a dialogue with the user, the computer acting as a "consultant," by suggesting options on the basis of its knowledge and the symbolic data supplied by the user. Moving from known items of information to unknown information is the vital process of a consultant system. The user of a consultant expert system has observed some particular state of affairs within the domain of the system's expertise and submits these observations to the system. Based

* This work was conducted at the MIT Laboratory for Information and Decision Systems with support provided by the Basic Research Group of the Technical Panel on C^3 of the Joint Directors of Laboratories through the Office of Naval Research under Contract no. N00014-85-K-0782.

on the observations, the system makes inferences and suggests new routes of investigation which will yield high grade information. Interactions continue until the system finds the most likely explanation of the observations. The formalism used to represent knowledge in consultant expert systems is the production system model.

There are three distinct components in an expert system, the Knowledge Base, the Fact base, and the Inference Engine.

The Knowledge Base contains the set of information specific to the field of expertise. Knowledge is expressed in a language defined by the expert. The knowledge base is a collection of general facts, empirical rules, and causal models of the problem domain. A number of formalisms exist to represent knowledge. The most widely used is the production system model in which the knowledge is encoded in the form of antecedent-consequent pairs or IF-THEN rules. A production rule is divided in two parts:

- A set of conditions (called left-hand side of the rule) combined logically together with a AND or a OR operator,

- A set of consequences or actions (called also right-hand side of the rule), the value of which is computed according to the conditions of the rule. These consequences can be the conditions for other rules. The logical combination of the conditions on the left-hand side of the rule has to be true in order to validate the consequences and the actions.

An example of a production rule is:

IF the flying object has delta wings AND
the object flies at great speed
THEN the flying object is a fighter plane.

The conditions "the flying object has delta wings" and "the object flies at a great speed" have to be true to attribute the value true the consequence "the flying object is a fighter plane."

The relationships among the rules of a production system can be represented with an inference net. The net shows graphically the logical articulation of different facts or subgoals, and identifies which rules are used to reach a specific goal. Let us consider the following production rules:

if A AND B, then C
if D OR E, then F
if NOT G, then H.

These rules are represented in the inference net formalism on Figure 1.

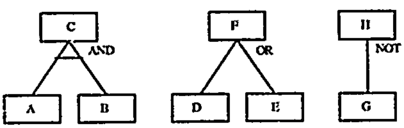


Figure 1 Representation of the logical operators in the inference net formalism

The Predicate Transition Net model developed in this paper is an extension of the inference net formalism and permits the explicit representation of the rules of a knowledge base and the relationships among them.

The fact base, also known as context or working memory, contains the data for the specific problem to be solved. It is a workspace for the problem constructed by the inference mechanism from the information provided by the user and the knowledge base. The working memory contains a trace of every line of reasoning previously used by memorizing all the intermediate results. Therefore, this can be used to explain the origin of the information deduced or to describe the behavior of the system.

The Inference Engine is used to monitor the execution of the program by using the knowledge base to modify the context. It uses the knowledge and the heuristics contained in the knowledge base to solve the problem specified by the data contained in the fact base. In the production system modeled in this paper, the rules are of the kind, A -> B, saying that, if A is valid, B can be deduced. The inference engine selects, validates, and triggers some of these rules to reach the solution of the problem.

In order to deal with uncertainty in items of evidence, fuzzy logic has been implemented in the model to combine logically the conditions of the left-hand side of the production rules. The value of a rule or a fact is either unknown or a number, pi, between 0 and 1, representing the degree of truth associated with it. The operators AND, OR, and NOT execute operations on these degrees of truth as follows:

$$p1 \text{ AND } p2 = \min(p1, p2)$$
$$p1 \text{ OR } p2 = \max(p1, p2)$$
$$\text{NOT } p1 = 1 - p1.$$

Among the strategies used by the inference engine to select the rules, forward chaining and backward chaining are the most common. In forward chaining, the inference mechanism works from an initial state state of known facts to a goal state. It finds first all the rules that match the context, then it selects one rule based on some conflict resolution strategy, and then execute the selected rule. Facts are inputs to the system. The most appropriate hypothesis that fits the facts is deduced. For backward chaining, the system tries to support a hypothesis by checking known facts in the context. If these known facts do not support the hypothesis, the preconditions needed for the hypothesis are set up as subgoals. The process for finding a solution is to search from the goal to the initial state, it involves a depth-first search.

In order to simulate the behavior of an expert system, the process of selection and firing of rules done by the inference engine has been modeled when a backward chaining strategy is used. A trigger is associated with every rule (or operator). A rule is selected by the inference engine when the trigger is activated. Only one rule at a time can be activated and the continuation of the selection and firing process is done according to the result of the rule:

- If the result is unknown, the rule is put in memory and the rule which gives the value of the first unknown precondition is selected.

- If the result is known, the last rule which was put in memory is selected again because the produced result is the value of one of its preconditions.

Let us consider the example where we have two rules.

$$B \Rightarrow C \quad (1)$$
$$A \Rightarrow B \quad (2)$$

and where the degree of truth of the fact A is known.

The inference engine selects first rule (1). The degree of truth of C is unknown because the degree of truth of B is unknown. Rule (1) is then de-activated and put in memory. Then rule (2) is selected. Since the value of A is known, the value of B is deduced. Rule (1), which is the last to have been put in memory, is selected again and the answer C is obtained.

The process of selection and firing of rules described above is repeated by recursion until the final answer is found; the process can last a long time. In the search for efficiency and performance, unnecessary computations must be avoided. In some cases, there is no need to know the values of all the preconditions of a rule to deduce the value of its consequence. For example, in Boolean logic, if we have the rule :

A AND B => C.

and we know that :

A is false,

then the consequence C is false and there is no need to look for the value of B to conclude that ; the set of rules giving the value of B can be pruned.

In systems using fuzzy logic, this avoidance of unnecessary computations is all the more important as computations are more costly in time and memory storage than in systems using Boolean logic. The problem is that little improvement in performance is obtained, if extra computation is avoided only in the case of complete truth (for the operator OR) or of complete falsity (for the operator AND). The solution lies in the setting of thresholds for certain truth and certain falsity. For example, in the case of the operator AND, if we have:

A AND B => C

and if we know that the degree of truth of A is less than the threshold of certain falsity, then we can deduce that the degree of truth of the consequence C is less than the degree of truth of A and, therefore, less than the threshold of certain falsity. There is no need to know the degree of truth of the precondition B. The thresholds for which no further search is required in the execution of the operators are set to 0.8 for certain truth in the operator OR and 0.2 for certain falsity in the operator AND. A rule or fact having a degree of truth larger or equal to 0.8 (resp. less or equal to 0.2) will be considered to be true (resp. false). Therefore, the logic takes into account the unknown rules or facts.

1.2 Characteristics of the Predicate Transition Nets Used in the Model

Predicate Transition Nets have been introduced by Genrich and Lautenbach (1981) as an extension of the ordinary Petri Nets (Peterson, 1980; Reisig, 1985) to allow the handling of different classes of tokens. The Predicate Transition Nets used in the model have the following characteristics.

Tokens. Each token traveling through the net has an identity and is considered to be an individual of a given class called variable. Each variable can receive different names. For this model, two classes of tokens are differentiated :

(1) The first class, denoted by P, is the set of the real numbers between 0 and 1, representing the degrees of truth of the facts or items of evidence. The names of the individual tokens of these classes will be p, p1, p2.

(2) The second class is denoted by S. The individuals of this class can only take one value. Only one token of this class will travel through the net and will represent the action of the inference engine in triggering the different rules.

Places. Places are entities which can contain tokens before and after the firing of transitions. Three kinds of places are differentiated:

(1) places representing a fact or the result of a rule and containing tokens of the class P or no token at all,

(2) places used by the system as triggers of operators and containing the token of the class S. These places and the connectors connected to these places are represented in bold style in the Figures and constitute the system net.

(3) places allowed to contain different kinds of tokens (P and S) and which are used to collect the tokens necessary for the enabling of the transitions of which they are the input places.

The marking of a place is a formal sum of the individual tokens contained in the place. For example, a place A containing a token of the class P, p1 and the token of the class S has the marking M(A) :

M(A) = p1 + S

Connectors and Labels. Each connector has a label associated with it which indicates the kinds of tokens it can carry. A special grammar is used on the labels to define in what way tokens can be carried. The labels of connectors linking places to transitions contain conditions that must be fulfilled for them to carry the tokens. The labels of connectors linking transitions to places indicate what kind of token twill appear in the places after the firing of the transition.

The following notation in labels is used :

When token names are joined by the sign "+" then the tokens defined by these names have to be carried at the same time. For example, the label "p + S" indicates that one token of the class P and one token of the class S have to be carried together at the same time by the connector.

When token names are joined by the sign "," then the tokens defined by these names can be carried at different times but not together. For example, the label "p, S" indicates that either a token of the class P or a token of the class S can be carried.

Mixing of notation is possible. The label "p+S, S" indicates that the connector can carry either a token of the class P and a token of the class S or only one token of the class S

A connector without label has no constraint on the kind of tokens it can carry.

In some cases, the connector has to carry the token of class S when there is no token of the class P involved in the firing of a transition. The statement "absence of token of the class P" is denoted by the symbol Ø. This symbol is used in the labels, as if it was a class of tokens, in association with the names of the other classes. The symbol Ø is used in the following cases

(1) The label "S+Ø" means that the connector can carry a token of the class S, if there is no token of the class P.

(2) The label "(S+p), (S+Ø)" means that the connector can carry either a token of the class S and a token of the class P, or a token of the class S, if there is no token of the class P.

Transitions. Transitions have attached to them a predicate which is a logical formula (or an algorithm) built from the operations and relations on variables and tokens in the labels of the input connectors. The value (true or false) taken by the predicate of a transition depends on the tokens contained in the input places of the transition. When the predicate has the value "true", the transition is enabled and can fire. In the model of the consultant expert system, predicates are conditions on tokens of the class P.

A transition without predicates is enabled as soon as all the input places contain the tokens specified by the labels of the connectors.

Transitions with predicates are represented graphically with squares or rectangles. The predicate is written inside. Transitions without predicates are represented with bars as in ordinary Petri Nets.

Firing Process. The conditions of enabling of a transition are: (1) the input places contain the combination of tokens specified by the labels of the connectors, and (2) the predicate of the transition is true. If these two conditions are fulfilled, the transition can fire. In the firing process, tokens specified by the input connectors are withdrawn from the corresponding input places and tokens specified by the output connectors are put in the output places. Let us consider the example shown on Figure 2:

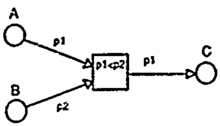


Figure 2 Example of a transition with a predicate

The condition "p1 < p2" written in the transition represented by a square is true when the value of the token named p1 coming from place A is less than the value of the token named p2 coming from place B, as specified by the connectors. In this case, the transition is enabled and can fire; the tokens p1 and p2 are withdrawn from the places A and B and a token p1 is put in place C.

1.3 Logical Operator Models

In order to construct the model of the expert system using Predicate Transition Nets, it is necessary to construct first models of the logical operators AND, OR, and NOT. The results are shown in Figures 3, 4 and 5. Let us describe now what happens in the operator AND (the operators OR and NOT behave in a similar way).

The operator drawn in Figure 3 realizes the operation :

A AND B => C.

AND

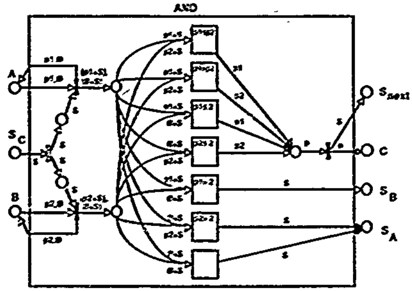


Figure 3 Model of the operator AND

It can be represented as a black box, having three inputs: A, B and S_C (the trigger) and six outputs : C (the result), A, B (memorizing of the input value) and three system places S_A, S_B and S_{next}. Only one of those system places (represented in bold style in the figures) can have a system token at the output. S_{next} will contain a system token, if the result of the operation is known, i.e., if C contains a token of the class P. This shows that the next operation can be performed. If the result is unknown, i.e., the two inputs are not sufficient to yield a result, the system token is assigned to S_A or S_B in order to get the values of these unknown inputs. A system token will be assigned to S_A if (i) C is unknown and (ii) A is unknown or if A and B are both unknown. The system token will be assigned to S_B if C is unknown and only B is unknown.

The execution of the operation will start only if there is a system token in S_C. We denote by S_C the trigger place of the operator computing C. As soon as there is a token in S_C, the two input transitions are triggered by the allocation of a system token (S) at the input places of these transitions. The values of A and B are therefore reproduced in A and B and in the output place of each of the transitions. These places contain also a system token, which will ensure the enabling of the following transition (i.e., that the two inputs are present). These two places are the input places of seven different transitions which have disjoint conditions of enabling. Only one of these transitions can be enabled and can fire. At the firing the result, if any, is given in the result place and then in C, while the system token is assigned either to S_{next}, or to S_A, or to S_B

OR

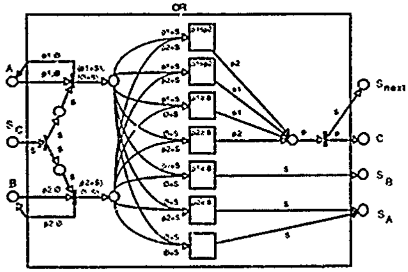


Figure 4 Model of the operator OR

These operators can be compounded in super-transitions. The model can be generalized to operators with more than two inputs by combining these basic operators.

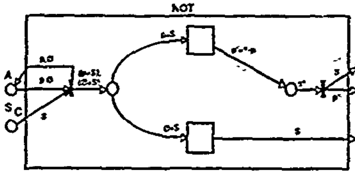


Figure 5 Model of the operator NOT

An example of the use of these logical operators is shown on the next section, where a simple inference net is modeled and the search process in this net is simulated.

1.4 Dynamic Representation of an Inference Net

The connection of the super-transitions representing the logic operators to places representing the items of evidence leads to a dynamic representation of an inference net. It allows to show explicitly how the inference engine scans the knowledge base. By running a simulation program, we can see in real time what the steps of reasoning are, the possible deadlocks, or mistakes. It allows one to identify the parts of the knowledge base where the knowledge representation is incorrect.

Let us consider the simple symbolic system containing the following rules:

if A and B => E
if C and D => D
if E or F => G

The standard representation of the inference net of this system (see section 3.1) is shown in Figure 6.

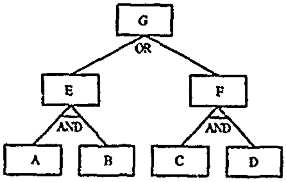


Figure 6 Standard representation of the inference net of the example.

The representation of the inference net with Predicate Transition Net is deduced from this representation by:

(1) replacing the rectangles representing the subgoals with the places of our model.

(2) replacing the formalism AND, OR, and NOT by the models of the operators aggregated in super-transitions, and linking these places to those transitions (including the self loops)

(3) linking the system places of each operator according to the rules described in section 4 for the scheduling of the checking of the unknown subgoals.

The representation of the inference net of the simple symbolic system in Figure 6, using the Predicate Transition Net models of the logic operators, is shown on Figure 7. The interface module with the user has been added through the places IA, IB, IC and ID, where the user can enter the degrees of truth of A, B, C and D.

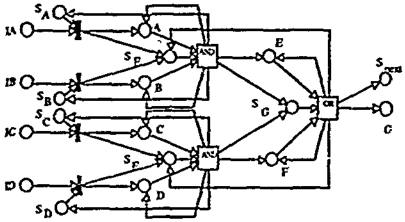


Figure 7 Inference net of a simple symbolic system, using the Predicate Transition Nets formalism

The simulation of the propagation of the tokens in this net allows one to observe the reasoning process followed by the system. The mapping of the different places of the net at each step of the process of the simulation is shown on Table 1.

TABLE 1 Mapping of the Places at the different steps of the simulation

	A	IA	B	IB	C	IC	D	ID	E	F	G	S_A	S_B	S_C	S_D	S_E	S_F	S_G	S_{next}
Step 1																		S	
Step 2																S			
Step 3												S							
Step 4		0 9										S							
Step 5	0 9															S			
Step 6	0 9												S						
Step 7	0 9			0 8									S						
Step 8	0.9		0.8													S			
Step 9	0 9		0 8						0 8									S	
Step 10	0 9		0 8						0 8		0 8								S

The search for the degree of truth of the goal G starts when the system token is put in the system place S_G, at the beginning of the search (step 1). The degree of truth of G cannot be evaluated when the operator OR is executed. The system token is therefore assigned to S_E for the checking of the subgoal E (step 2). The execution of the operator AND cannot lead to a result for E and the system token is allocated to S_A (step 3), which triggers an interaction session with the user to get the degree of truth of A. The user enters this value (say 0 9) through IA (step 4) which is assigned to A, while the system token is assigned to S_E (step 5). Since, the degree of truth of A is larger than 0 2, the result of the operator AND cannot be given in E and the system token is assigned to S_B (step 6) to get the degree of truth of B (say 0 8) through IB (step 7). The system token is then reassigned to S_E to trigger the operator AND (step 8), which can now be executed. The minimum of the degrees of truth of A and B, 0.8, is put in E, while the system token is assigned to S_G (step 9). Since the degree of truth of E is equal to 0.8, the operation OR can be performed to produce the result G equal to 0.8 The system token is allocated in S_{next} (step 10). The subgoal F has not been checked and all the part of the net which is used to evaluate F has been pruned.

1.6 Evaluation of the Response Time of an Expert System

The model allows the evaluation of the time needed to produce an output; this is then used to assess the timeliness of an organization using an expert system.

The response time of an expert system is related to the number of rules in the rule base scanned by the system to give an answer to a specific problem or goal, and to the number of interactions with the user. The model we have defined allows a quick identification of the parts of the rule base which have been scanned, given a certain set of inputs, to reach a specific goal, since each place contains the token symbolizing the value of the rule or fact it represents.

Let us consider an expert system being used to give a certain answer in a certain environment. We represent the input X_i to the system as a n-tuplet where n is the total number of questions which can be asked by the system. The answer to the questions are contained in this n-tuplet at the location corresponding to the question asked (this may not be listed in the order of appearance in time). The locations for the unasked questions are left empty. We denote by n_i the number of questions asked by the system. The number of X_is might be very large but it is bounded. Given a certain environment, we can define a distribution $p_i(X_i)$ for the occurrence of the input X_i.

For a specific input X_i, we can identify N_i, the number of places scanned by the system to reach its goal, since they still contain the degrees of truth of the subgoals they represent. If τ is the average time to check a rule and t is the average time taken by a user to answer a question asked by the system, then the time t_i to get an answer given an input X_i will be .

$$t_i = N_i \tau + n_i t$$

Therefore, the average time of use T of the expert system for the set of inputs X_i will be given by.

$$T = E[t_i] = \sum_i p_i t_i = \sum_i p_i N_i \tau + \sum_i p_i n_i t$$

which leads to :

$$T = E[N_i] \tau + E[n_i] t \quad ,$$

where E[X] denotes the expected value of the variable X.

The time T obtained is the average time needed to get an answer from the expert system. This model of a consultant expert system will be used to evaluate the effect that inconsistent information can have on the command and control process.

2.0 USE OF THE EXPERT SYSTEM FOR THE FUSION OF INCONSISTENT INFORMATION

An important problem faced by decisionmaking organization is the inconsistency of information which can degrade substantially their performance. This inconsistency can be attributed to different causes: inaccuracy in measured data, lack of sensor coverage, presence of noise, bad interpretation of data. In a military context, inconsistency of information can also be explained by the attempt by the enemy to mislead the organization about his actions through the use of decoys or jamming techniques. This presence of inconsistent information jeopardizes the successful execution of the mission of an organization.

Three strategies to fuse inconsistent information are considered in this paper . (1) ignore information sharing, (2) weighted choice among contradictory sets of data and (3) use of an expert system which has additional knowledge on the problem to be solved.

The first strategy occurs when the decisionmaker performing the information fusion uses only his own assessment and ignores the assessment of the other decision maker. This strategy is related to the way a human being assigns value to information which is transmitted to him, while executing a specific task. The study of Bushnell, et al. (1988) develops a normative-descriptive approach to quantify the processes of weighting and combining information from distributed sources under uncertainty. Their experimentation has shown that one of the human cognitive biases, which appears in the execution of a task, is the undervaluing of the communications from others, which occurs independently of the quality of the information received. The decisionmaker is, therefore, expected to have the tendency to overestimate his own assessment and to assign a lower value to the others assessments.

The second strategy is to perform a weighted choice among the contradictory assessments which are transmitted to him and compared to his own. This weighting strategy involves the confidence which can be given to the information and which depends on the manner this information has been obtained, or on its certainty. In many models of organizations facing this problem of inconsistent information and using the weighted choice strategy, measures of certainty are the basis for the weighting of different items of evidence. Among the methods used, the Bayesian combination has given valuable results.

The third strategy involves the use of an expert system. Expert systems can consider additional knowledge and facts which would be too costly in terms of time, effort, and memory storage to be handled efficiently by the decisionmaker on his own. For each instance of contradictory data, it can check if their values are consistent with the knowledge it has and give an indication of their correctness. With this additional attribute, the decisionmaker can perform a more precise information fusion.

In order to illustrate how these strategies modify the measures of performance of an organization and to emphasize the role of an expert system in the fusion of inconsistent information, an illustrative application will be used.

2.1 Command and Control in an Air Defense problem

Mission and Organization. The illustrative application involves an organization, the mission of which is to defend a set of facilities against attacking missiles. This set of facilities consists of three cities, two military bases and two production facilities located in a square with 30 mile sides, as shown on Figure 8. To destroy incoming missiles, the organization can either use a laser beam or send an antimissile rocket. The laser beam is used in case of urgency, when the time before the missile hits its target is less than a certain threshold The antimissile rocket is used when enough time is available. Both weapons require different targeting solutions. The performance of the organization is measured by its ability to send the right weapon at the right place for each incoming threat.

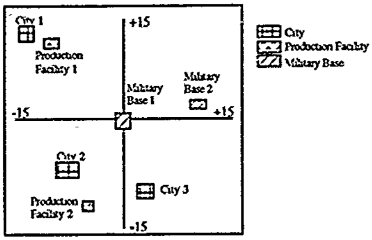


Figure 8 Location of facilities to be defended by the organization

The considered organization is a hierarchical two-decisionmakers organization with the Petri net representation (Tabak and Levis, 1984, Remy et al., 1987) shown in Figure 9. The two decisionmakers, DM1 and DM2, perform their own situation assessment producing the results Z1 and Z2. DM2 sends Z21, which is equal to Z2, to DM1 who is in charge of performing the information fusion with one of the three strategies available. One of them is to use an expert system Using the revised situation assessment Z'1, the response Y1 is selected and transmitted to DM2. DM2 takes into account this new information in his information fusion stage and realizes the final response selection of the organization, Y.

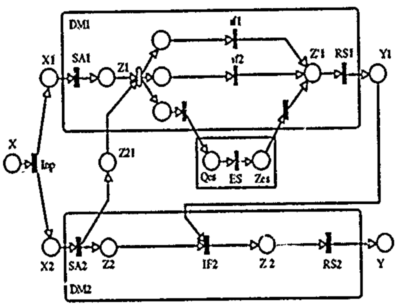


Figure 9 Petri net of the hierarchical 2-DM organization.

Inputs and situation assessments Each decisionmaker receives as input two points of the trajectory of the missile. The first one is its position at time t, which is the same for the two decisionmakers to make sure they are assessing the same missile The second point is determined by the tracking center of each decisionmaker. The tracking center is defined as the sum of the human and hardware means assembled to process the information. The use of decoys by the enemy and the presence of noise result in these positions being not the same for each of the decisionmakers When this is the case, we assume that one of the two is the actual one. In addition to these different coordinates, the input contains also the confidence factors associated with each position These confidence factors have been generated by a preprocessor (say, a tracking algorithm) and measure the quality that can be attributed to each set of data.

After receiving these inputs, the two decisionmakers, DM1 and DM2, perform the same situation assessment. DM1 (resp. DM2) computes the velocity of the missile and evaluates its impact point, according to the set of coordinates he has received, and produces the result Z1 (resp. Z2). DM2 sends Z21, which is equal to Z2, to DM1 who is in charge of performing the information fusion.

Information fusion of DM1: In his information fusion stage, DM1 makes first the comparison between Z1 and Z21. If they are equal, Z'1 = Z1 is produced. If they are different, DM1 has to choose from the three different strategies described in the previous section.

The first one is to ignore information sharing. In this case, DM1 produces Z'1 = Z1 without considering the situation assessment Z21, transmitted to him by DM2.

The second strategy is the weighting of the information according to the confidence factors associated with each set of data. DM1 considers the confidence factors Conf1 and Conf2 given with the input and which measure the quality of the information to choose Z1 or Z21 If Conf1 is greater than or equal to Conf2, DM1 produces Z'1 = Z1. In the opposite case, DM1 produces Z'1 = Z21.

The last strategy involves the use of an expert system The simple knowledge base system which has been developed for this application evaluates the degree of threat a missile represents as a function of the distance between the location of the different facilities and its impact point estimated by the user A more sophisticated system could make the assessment of the threat by taking into account the type of missile, the geographical aspect of the area, the direction of winds, the interest for the enemy to destroy the aimed facility, ... The threat assessment of the missile is done for the two possible trajectories, one after another. If the first threat assessment shows that the target is one of the facilities with enough certainty, the computer stops its search. In the opposite case, the computer evaluates also the threat that the missile would have if it followed the second trajectory. The answer of the expert system consists of two numbers between 0 and 1 representing the severity of the threat posed by the missile (according to each assessment). When the answer is given, DM1 does not use a strategy to make a comparison with a result from an internal algorithm, as shown by Weingaertner and Levis (1987). This is due to the fact that the decisionmaker has not enough data on his own to be able to double check the answer of the decision aid If the degree of threat according to the assessment of DM1 is greater than or equal to the one according to the assessment of DM2, the result is Z'1 = Z1.In the opposite case, the result is Z'1 = Z21

Response of the organization Having chosen the trajectory which seems to be the most likely, DM1, in his response selection stage, determines the type of threat the missile represents by computing the time before impact and sends it to DM2 with the fused information DM2, in his information fusion stage selects the weapon to use and performs the targeting solution in his response selection stage

2.2 Measures of Performance

The measures of performance considered in this paper are workload (Boettcher and Levis, 1982; Levis, 1984), timeliness (Cothier and Levis, 1986) and accuracy (Andreadakis and Levis, 1987). They have been defined for the two possible types of interaction between the computer and the user:

The user initiated mode when the decisionmaker enters all the data he has in a specified order and the machine produces a result. Not all entered data may be needed by the machine in its search process.

The computer initiated mode when the user enters specific data only in response to requests from the computer.

Thirty three equiprobable inputs to the organization have been considered. Twenty four inputs contain inconsistent information. We assume that for half of these inconsistent inputs, the tracking center of DM1 is correct (the tracking center of DM2 is correct for the other half because we assume that for each input, one of the two contradictory positions is correct).

2.2.1 Workload

The evaluation and the analysis of workload in decisionmaking organization uses an information theoretical framework (Levis, 1984). It allows to evaluate the activity of a decisionmaker by relating, in a quantitative manner, the uncertainty in the tasks to be performed with the amount of information that must be processed to obtain certain results.

The information theoretic surrogate for the cognitive workload of a decisionmaker is computed by adding all the entropies of all the variables used to model the procedures he uses to perform his task. The distributions of all the variables are generated by executing the algorithms for all the inputs. This process of generation starts with a probability equal to zero for all the values that each variable can take. When the execution of the algorithm is performed with the input X_j having a probability p_j, the internal variable w_i, if it is active, takes the value a_i. The probability mass function of this variable w_i is updated by adding the probability p_j to the probability this variable had to take the value a_i before the execution of the algorithm with this input X_j. The operation for all the variables w_i affected by the input is:

$$p(w_i = a_i \mid X_1, ..., X_{j-1}, X_j) = p(w_i = a_i \mid X_1, ..., X_{j-1}) + p_j$$

However, to take into account the effect of the different strategies, the workload of the decisionmakers has to be computed for all the mixed strategies. A mixed strategy is a convex combination of the three pure strategies, and is noted (p_1, p_2, p_3), where p_i, $\{i = 1,2,3\}$ is the probability of using strategy i in the mixed strategy. The quantities p_1, p_2 and p_3 verify:

$$p_1 + p_2 + p_3 = 1$$

To compute the workload of DM1 and DM2 for all the mixed strategies, the system of all the variables has to be divided in three subsystems.

The first subsystem is composed of the internal variables of the algorithms for situation assessment of DM1 and DM2. The execution of these algorithms and the values taken by their internal variables for each input do not depend on the strategy chosen in the information fusion stage. Therefore, these algorithms are executed only once for each input to generate the probability mass functions of their internal variables. This subsystem allows to compute the invariant part of the workloads of DM1 and DM2, G_{inv}^1 and G_{inv}^2.

The second subsystem is made of the variables of the different algorithms of the information fusion stage. This subsystem has for input (Z1,Z21) and produces the output Z'1 with three different algorithms. Each algorithm i is executed independently of the others for all the inputs and the sum of the entropy of its internal variables (Z'1 is considered to be an internal variable of each algorithm) gives the activity of coordination of the algorithm of the strategy i, g_c^i. The contribution of this subsystem to the workload of DM1 is evaluated by using the Partition Law of Information (Conant,1976).

The throughput, G_t, is given by:

$$G_t = T(Z1, Z21 : Z'1)$$

The blockage term, G_b, which represents information in the input not reflected in the output, is given by:

$$G_b = H(Z1, Z21) - G_t$$

We assume that the data are noiseless and that the algorithm are deterministic. This assumption is made only to simplify the presentation. The noisy case with stochastic algorithms leads to additional terms in various expressions. In this case, the noise, G_n, is only caused by the internal choice in the decisionmaking process and is simply given by:

$$G_n = H(u)$$

where u is the decision variable specifying the choice among the different algorithms. H(u) is equal to:

$$H(u) = H(p_1) + H(p_2) + H(p_3)$$

As stated by Boettcher (1981), the coordination term is given by:

$$G_c = \sum_{i=1}^{3} (p_i g_c^i + \alpha_i \mathcal{H}(p_i))$$

where:

$$\mathcal{H}(p_i) = p_i \log_2(p_i) + (1 - p_i) \log_2(1 - p_i)$$

and α_i is the number of internal variables of the algorithm i. We have therefore the activity of the subsystem, $G_{subsystem}$

$$G_{subsystem} = H(Z1, Z2) + H(u) + \sum_{i=1}^{3} (p_i g_c^i + \alpha_i \mathcal{H}(p_i))$$

Finally, since the entropies of Z1 and Z21 have been evaluated in the first subsystem, the contribution $G_{II}(p_1, p_2, p_3)$ of the second subsystem to the workload of DM1 for the mixed strategy (p_1, p_2, p_3) is:

$$G_{II}(p_1, p_2, p_3) = H(u) + \sum_{i=1}^{3} (p_i g_c^i + \alpha_i \mathcal{H}(p_i))$$

The third subsystem is composed of the variables of the algorithms used after the information fusion stage. These algorithms are the response selection of DM1, the information fusion and the response selection of DM2. The variables of these algorithms can take three values that are different for each input according to the pure strategy used. Therefore, for each variable of this subsystem, three probability mass functions are generated for all the inputs and for each pure strategy. To compute the entropies of these variables for the mixed strategies, a convex probability mass function is deduced from the probability mass functions determined for each pure strategy. By summing these entropies, the variable contribution to the workload of DM1 and DM2 is deduced, $G_{var}^1 (p_1, p_2, p_3)$ and $G_{var}^2 (p_1, p_2, p_3)$. The workload of DM1 and DM2 can now be evaluated:

$$G_{DM1} = H(\text{input}) + H_{inv}^1 + H(Z21) + G_{if}(p_1,p_2,p_3) + G_{var}^1(p_1,p_2,p_3)$$

$$G_{DM2} = H(\text{input}) + H_{inv}^2 + H(Y1) + G_{var}^2 (p_1,p_2,p_3)$$

2 2 2 Timeliness

The measure of timeliness considered in this application is related to the response time of the organization. A deterministic processing time has been associated with every algorithm. Again, each processing time can be described by a probability density function and the probability density function of the response time can be computed (see Andreadakis and Levis, 1987) The use of stochastic model does not add to the presentation of the example, but would be the model to use for an experimental investigation For the strategy involving the use of the expert system, the time to give an answer has been computed using the expert system model described in the first section of the paper. The response time of the expert system is function of the number of rules scanned by the system for each input to the organization and of the number of interactions with the user. This time is likely to vary with the mode of interactions used.

We assume that DM1 and DM2 perform their situation assessment concurrently and synchronously, and that the same amount of time is needed by the two to give an answer Therefore, only one of the two processing times is considered. T_{SA1}, (resp. T_{RS1}, T_{SA2}, T_{IF2} and T_{RS2}) denotes the time needed to execute DM1's situation assessment algorithm (resp Dm1's response selection, DM2's situation assessment, information fusion and response selection) $T_{IF1}(i)$ is the time needed to perform the information fusion using a pure strategy i (i = 1, 2, 3). $T_{IF1}(3)$ is a function of the average response time of the expert system computed from its response time for all the inputs. The response time for the strategy i, T(i) is therefore:

$$T(i) = T_{SA1} + T_{IF1}(i) + T_{RS1} + T_{IF2} + T_{RS2}.$$

The response time for each mixed strategy (p_1, p_2, p_3) is given by a convex weighting of the response time for each pure strategy If T (p_1, p_2, p_3) denotes the response time of the organization when the strategy (p_1, p_2, p_3) is used, we have :

$$T(p_1, p_2, p_3) = p_1 T(1) + p_2 T(2) + p_3 T(3)$$

2 2 3 Accuracy

Accuracy of the organization has been evaluated by comparing the actual response of the organization with the desired or optimal response expected for each input This desired response is known to the designer. A cost of one has been attributed when the incorrect type of weapon is used or when the target point is not accurate. For each input X_j having a probability $p(X_j)$, the use of the pure strategy i generates the response Y_{ij} which is compared to the desired response Y_{dj}. The cost function $C(Y_{ij}, Y_{dj})$ has the following characteristics :

$$C(Y_{ij}, Y_{dj}) = \begin{cases} 1 & \text{if } Y_{ij} \neq Y_{dj} \\ 0 & \text{if } Y_{ij} = Y_{dj} \end{cases}$$

The accuracy J(i) obtained for the pure strategy i is :

$$J(i) = \sum_j p(X_j)\, C(Y_{ij}, Y_{dj})$$

The accuracy for the mixed strategy (p_1, p_2, p_3), $J(p_1, p_2, p_3)$, is obtained by computing the convex combination of the accuracy for each pure strategy :

$$J(p_1, p_2, p_3) = p_1 J(1) + p_2 J(2) + p_3 J(3)$$

Consequently, J represents the probability that an incorrect response will be generated. The lower the value of J, the better the performance is. The next section provides an analysis of the results obtained by using these measures of performance

3.0 RESULTS AND INTERPRETATION

Using the method described above, measures of performance have been evaluated for the three strategies. For the strategy involving the use of an expert system, we have considered two different options for dealing with uncertainty in the firing of rules, Fuzzy logic or Boolean logic, and two modes of interaction between the user and the decision aid, user initiated mode or computer initiated mode. The results are summarized in Table 2.

TABLE 2. Measures of Performance for the three strategies

	Strategy 1 Ignoring other assesment	Strategy 2 Weighted Choice	Expert System Fuzzy Logic		Expert System Boolean Logic	
			user initiated	computer initiated	user initiated	computer initiated
J prob of error	0 360	0 270	0 210	0 210	0 240	0 240
T seconds	18 240	18 960	21 015	20 850	20 974	20 364
G1 bits/symbol	63 414	64 921	70 293	70 293	65,969	65 969
G2 bits/symbol	43 920	43 847	43 240	43 240	43 287	43 287

3.1 Measures of Performance

3.1.1 Pure Strategies.

The three first columns of table 2 display the measures of performance (MOPs) of the organization for each pure strategy These results show that the taking into account of more knowledge, either about the way data are obtained, in the case of the weighted choice strategy, or about the meaning of information, when the expert system is used, yields greater accuracy. Accuracy is an important issue for the kind of mission this type of organization is expected to carry out. The results show also that taking into account more knowledge requires the handling of more data Therefore, more time is needed and more effort, expressed in terms of workload, is required This increase in workload is caused more by the extra decisions which must be made when the knowledge is taken into account than by operations or manipulation done with the additional knowledge. These manipulations are done by the decision aids, out of the control of DM1

When DM1 ignores the situation assessment of DM2, very few operations are performed. The response time is the smallest of the three. If the measure of timeliness is the ability of the organization to give a response as fast as possible, this strategy leads to a more timely response than the two others. The simplicity of the algorithm results in low workload for DM1 in comparison with the other strategies which can be explained by the fact that DM1 handles fewer variables. This strategy has low accuracy in comparison with the other strategies, because the choice made on the information to be fused is arbitrary and has no rational justification. Thus, a clear assessment of the cost and valuse of coordination can be made.

For the weighted choice strategy, no operation on variables received is performed. DM1 makes only a comparison between the weights of the information. We have assumed that the weighting process was carried out outside the organization by a preprocessor and, consequently, DM1 performs only few operations more than in the first strategy. Therefore, workload and response time are slightly larger than for the first strategy because of the extra information obtained by comparing the confidence factors. An increase of 3.9 % in response time and of 2.4 % in the workload of DM1 is found. The measure of how the data have been obtained, given through the confidence factors, brings a large gain in accuracy. An improvement of 25 % in the accuracy of the organization in comparison to the first strategy is observed. These results show, as expected, that taking into account the quality of information plays an important role in the accuracy of the organization, without degrading substantially the other measures of performance.

When the expert system is used, the increase in workload of DM1 is about 8.3 % from the level of strategy 2, and 10.8 % from the level of the first strategy. This can be explained by the handling by DM1 of the assessments given by the expert system. These assessments are variables which have greater entropies and which require more processing. The increase in response time (of 10.8 % from the level of strategy 2 and of 15.2 % from the level of strategy 1) is mainly caused by the time taken by DM1 to interact with the system and the time needed to get the answer. This response time of the expert system can get larger as the size of the knowledge base and of the magnitude of the problem to solve increase. In the example, the simplicity of the expert system hides the real effect on timeliness which can be expected with the use of such interacting system. The gain in accuracy is very significant, about 22 %, in comparison with the accuracy reached with the second strategy and 41.7 % from the level reached when the situation assessment of the other DM is ignored. This shows the extent to which the accuracy is improved when additional knowledge is used to verify the correctness of information. By using the expert system to evaluate the threat and to estimate the severity of the threat for each possible trajectory, DM1 has a broader assessment which allows him to perform more accurate information fusion.

Finally, we note that the workload of DM2 remains almost constant for all the strategies. A variation of 1.5 % can be observed. He uses always the same algorithms, and only the different distributions of the variables of the algorithms obtained, when different strategies are used by DM1, explain this small variation in his workload

3.1.2 Mixed Strategies

The performance measures (accuracy, timeliness, and workload of DM1) reached by the organization, when mixed strategies are used by DM1 in his information fusion stage, have been obtained using CAESAR (Computer Aided Evaluation of System ARchitectures) Measures of Performance have been evaluated for all mixed strategies and have led to a surface in the space (J-T-G1) represented on Figure 10. The projections of this surface on the Accuracy - Workload (J-G1), and Timeliness - Workload (T-G1) planes are drawn on Figure 11. Measures of performance reached for each pure strategy are located at the three cusps of the figures. The convex combination of any two pure strategies gives a U-shaped curve (Boettcher and Levis, 1982) which can be explained by the fact that when a mixed strategy is used, there is an additional activity due to switching from one algorithm to another.

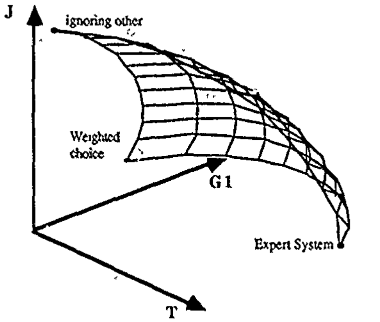


Figure 10 Locus of the Measures of Performance attained by the organization

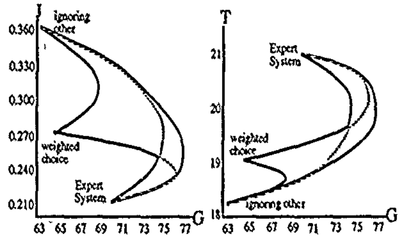


Figure 11 Mixed Strategies . Accuracy / Timeliness vs workload for DM1

The projection of the surface of the Measures of Performance on the Accuracy - Timeliness plane (J-T) gives the triangle shown on Figure 12, which shows the performance attained by the organization. The corners of this triangle indicate the level reached in accuracy and response time for each pure strategy. For all binary variations between pure strategies or for all successive binary combinations of mixed strategies, J and T are linear combinations of each other. Figure 12 shows clearly the trade-offs between response time and accuracy and how the requirements of the mission will justify a strategy. Thus, if the requirements in accuracy are too binding, the strategy of ignoring information sharing will not be acceptable. In the same way, if the time available to process each input is too short, the expert system would be useless because too much time will be needed to perform the information fusion.

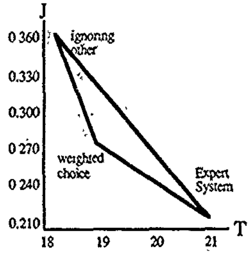


Figure 12 Mixed Strategies : Accuracy and Timeliness of the Organization

3.2 Effect o .ie mode of interaction

The effect of the mode of interaction on the measures of performance is shown on the last four columns of Table 2 There is no change in accuracy or workload; however, a slight change in timeliness is observed. This is caused by the fact that, in the user initiated mode of interaction, all the data which have a chance to be processed by the expert system are entered at the beginning of the session In the example, the position of the impact points according to the two different situation assessments are entered, even if the first set is sufficient to assess the threat. Therefore, more time is needed than in the computer initiated mo.; where data are entered at the request of the system during the search.

It is important to note that in the air defense, no workload have been assigned to the process of entering the information in the expert system. The process consists only of replication of the information the decisionmaker already has. If the inputs asked by the expert system do not correspond to the data the decisionmaker has, he would have to perform some operations to deduce these inputs from the information he has Let us consider an example where the decisionmaker has computed or received from another member of the organization the value of the speed of an object being analyzed. If the expert system asks the decisionmaker the question "speed of the object : {possible answer low, moderate, high}," the decision maker will have to deduce from the actual value of the speed the attribute asked by the system. A small algorithm will have to be executed, increasing his workload. It can be expected therefore that, in this case, a change in workload similar to the change in response time would be observed. This issue raises the problem of the adequate design of the expert system, or more generally, of the decision aid in which the mode of interaction has to be thought very carefully to avoid an unnecessary increase in the workload of the decisionmaker and in the response time.

3.3 Fuzzy Logic vs. Boolean Logic

For this illustrative application, the levels of performance reached when different expert systems are used have been studied. The performance achieved with an expert system using fuzzy logic as the means of inference, which has been developed for the example, has been compared to the performance obtained by using an expert system which does not deal with uncertainty and uses Boolean logic. This version of the expert system has been obtained by changing the mappng functions (only values 0 and 1 could be processed instead of the real numbers between 0 and 1). It has been assumed that a statement having a degree of truth greater (resp. smaller) than 0.6 was true (resp. false). Therefore, the assessment of the threat of the missile for each trajectory has only the values true or false. The different measures of performance obtained for the two systems are summarized in the last four columns of Table 2.

The organization has a response time slightly lower with an expert system using Boolean logic than with the expert system using fuzzy logic (2.3 %). This is due to the fact that by assigning the value true or false to the severity of threat, the system can reach a conclusion (which is not always the best one) by examining fewer possibilities. It can prune a larger part of the knowledge base than the fuzzy logic system when it reaches the conclusion that the missile is threatening a specific facility. When this conclusion is reached for the first possible trajectory, the other trajectory is not examined. This results in a shorter time to produce the answer and in fewer interactions with the user and therefore in a shorter response time.

Since the expert system with Boolean logic assesses the threat only with the value true or false, the answer of the expert system has a lower entropy The workload of the decisionmaker is therefore lower (about 6.8 %) when he uses the expert system with Boolean logic than when he uses the expert system with fuzzy logic.

By pruning a larger part of the knowledge base when it reaches a conclusion, the system has more chance to make the wrong assessment of the threat. The results show that, indeed, the system with Boolean logic exhibits lower accuracy than the system with fuzzy logic. The level of accuracy is, nevertheless, better than for the two other strategies expected to be used in the information fusion stage and is explained by the fact that more knowledge is taken into account in the information fusion process.

4.0 CONCLUSION

In this paper, a model with fuzzy logic as a means for dealing with uncertainty has been developed using the Predicate Transition Net formalism. A method to make time-related measures from this representation has been introduced, taking into account the portion of the rule base scanned by the system and the number of interactions. Then, the assessment of the role of an expert system has been made through the study of an example which involves a two decisionmaker organization facing the problem of fusion of inconsistent information The decisionmakers must identify the trajectories of threats that they then have to destroy to protect a set of facilities. In the example, the expert system helps the decisionmaker to clarify the contradictory situation assessment he has to fuse. This strategy has been compared to two others expected to be used in this situation : (1) ignoring the assessment of the other decisionmaker, (2) making a weighted choice among the two contradictory situation assessments, by taking into consideration the way the data used to produce these assessments have been obtained by each decisionmaker. Measures of performance (workload, timeliness and accuracy) have been evaluated The results show that the use of the expert system improves significantly the accuracy of the organization, but requires more time and increases the workload of the decisionmaker using it The comparison of the two modes of interaction between the user and the system has shown variations in workload and in response time the computer initiated mode requires less workload and less response time for a same level of accuracy This result tends to show that the design of an interacting decision aid must take into account not only the characteristics of the problem to be solved, but also the way the decisionmaker would use it.

REFERENCES

Andreadakis, S. K. and Levis, A. H., 1987, "Accuracy and Timeliness in Decisionmaking Organizations," *Proc. 10th IFAC World Congress*, Pergamon Press, New York.

Boettcher, K. L., 1981, *An Information Theoretic Model of the Decisionmaker*, S.M. Thesis, LIDS-TH-1096, Laboratory for Information and Decision Systems, MIT, Cambridge, MA.

Boettcher, K. L. and Levis, A. H., 1982, "On Modeling Teams of Interacting Decisionmakers with Bounded Rationality," *Automatica*, Vol. 19, No.6 (1983), pp. 703-709.

Bushnell, L. G., Serfaty, D. and Kleinman, D. L., 1988, "Team Information Processing: A Normative-Descriptive Approach," *Science of Command and Control: Coping with Uncertainty*, Edited by S. E. Johnson and A. H. Levis, AFCEA International Press, Washington, D.C.

Conant, R. C., 1976, "Laws of Information Which Govern Systems," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-6, No.4, pp. 240-255.

Genrich H.J., Lautenbach, K., 1981, "System Modeling with High Level Petri Nets", *Theoretical Computer Science*, *13*, pp. 109-136.

Giordona A. and Saitta L., 1985, "Modeling Production Rules by Means of Predicate-Transition Networks", *Information Sciences*, *35*, pp. 1-41.

Johnson, L. and Keravnov, E. T., 1985, *Expert Systems Technology, A Guide*, Abacus Press, Tunbridge Wells.

Levis, A. H., 1984, "Information Processing and Decisionmaking Organizations: A Mathematical Description," *Large Scale Systems*, Vol. 7, pp. 151-163.

Cothier P. H. and Levis A. H. "Timeliness and Measures of Effectiveness in Command and Control", *IEEE Transactions on Systems, Man and Cybernetics*, SMC-16, No 6.

Peterson, J. L., 1980, *Petri Net Theory and the Modeling of Systems*, Prentice Hall, Inc., Englewood Cliffs, NJ.

Reisig, W., 1985, *Petri Nets, An Introduction*, Springer Verlag, Berlin.

Remy P., Levis A. H., and Jin V. Y. Y., 1988, "On the Design of Distributed Organizational Structures," *Automatica*, Vol. 24, No.1, pp 81-86.

Tabak, D. and Levis, A. H., 1984, "Petri Net Representation of Decisions Models," *IEEE Transactions on Systems, Man and Cybernetics*, Vol. SMC-15, No.6, pp. 812-818.

Weingaertner, S.T. and Levis A. H., 1987, " Evaluation of Decision Aiding in Submarine Emergency Decisionmaking," *Proc. 1987 Symposium on C^2 Research*, National Defense University, Fort Mc Nair, Washington DC.

Whalen, T. and Schott, B., 1983, "Issues in Fuzzy Production Systems," *International Journal of Man-Machine Studies*, Vol. 19, pp. 57-71.

Zadeh, L. A., 1965, "Fuzzy Sets," *Information and Control*, Vol. 8, pp. 338-353.

Zadeh, L. A., 1983, "The Role of Fuzzy Logic in the Management of Uncertainty in Expert Systems," *Fuzzy Sets and Systems*, Vol. 11, pp. 199-227.

MULTIDISCIPLINE MISSION PLANNING AND MANAGEMENT

Dr. Alicia J. Browner
Lockheed Austin Division
0/T4-20, B/30F, (512) 448-5159
6800 Burleson Road
Austin, Texas 78744

Laura C. Morrison
Lockheed Austin Division
0/T5-40, B/30F, (512) 448-5126
6800 Burleson Road
Austin, Texas 78744

BACKGROUND

The current technique for the mission tasking of a multiple sensor suite on an airborne platform lacks the efficiency and coordination between sensors that is needed to maximize the effectiveness of the platform assets for a battlefield situation.

The state of the art in mission tasking for multisensor systems involves independent and often manual scheduling of each sensor within the flight-profile constraints and schedule provided for the platform. This is a time-consuming process that generally fails to provide for integrated, multisensor observations of critical targets in a fashion that facilitates their multispectral exploitation. New concepts and automated tools are required to plan and manage reconnaissance missions utilizing multidisciplined sensor suites in an integrated way. Solutions to this problem are needed that treat the complex tradeoffs between the conflicting goals of collection efficiency and survivability, which often arise during intelligence collection.

This paper presents work in progress whose goal is to develop new concepts, methodologies, and tools for the *planning* and *management* of multisensor intelligence and reconnaissance missions. The *planning* function begins with requests for intelligence collection and culminates in a detailed plan that specifies the platform flight path, sensor schedules, cue lists, self-protection rules, and historical information required by mission management. The *management* function monitors mission execution, performs multisensor fusion and cueing on processed sensor data, and modifies the mission plan in response to the immediate situation.

PROJECT OVERVIEW

This section focuses on the overall scope of the project, presenting the top-level concepts for developing mission planning and management tools. The next section describes the work to date that has been integrated into software.

The project approach is to integrate and evaluate the models and algorithmic solutions developed from these concepts in the framework of a software-based analysis system. Figure 1 depicts the three components of the analysis system. The *plan development* phase results in the complete specification of a mission plan. It includes flight path planning, sensor scheduling, and cue rules generation. This portion of the system has been the focus of this work to date. The *system simulation* phase provides a detailed time-history of the aircraft flight, sensor collection, threat responses, and target movements. The simulated flight will differ from the original mission plan as the mission manager responds to information gathered during execution. The *evaluation* phase collects and analyzes the statistics generated from the simulation.

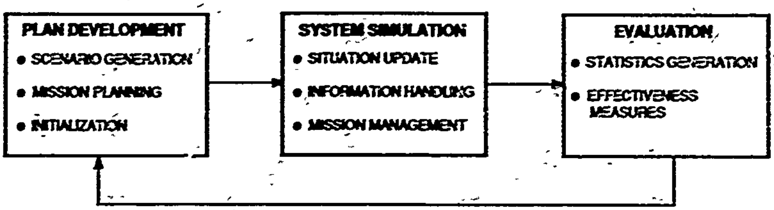


Figure 1. The analysis system for the development of mission planning and management algorithms provides a complete testing environment for new concepts in multisensor tasking. The system becomes the actual decision aid for a mission planner after the design and verification of the three components: plan development, system simulation and evaluation.

Plan Development

The functional flow for multisensor mission planning (Figure 2) interprets the planning process in terms that allow automated implementation. This corresponds to the plan development phase of the analysis system (Figure 1) and results in the initial mission plan depicted in Figure 2. Details on the software implementation of this functional flow are provided in the Impact System section of this paper.

The setup phase is initiated by submitting a request for collection, which consists of an object of interest, its associated area of interest (AOI), timeliness information, and desired information on the object. This includes information on quantifiers (e.g., location), acquisition level (detection, classification, recognition, identification), and activity status. The system then determines specific targets for IMINT and SIGINT sensors that will satisfy the collection request. At this time, threats to the survivability of the platform are added to the list of target objects on which to collect information, and a history of observables is referenced for known locations and densities. Next, an interactive feedback loop between the selection of the flight path and an evaluation of exposure to threats determines and refines the platform route. Also in the setup phase, tasks that can only be satisfied by a near-real-time management function (such as high-resolution location accuracy or identification of a target) are allocated to management and converted into a planning task (such as lower resolution accuracy or detection) with an associated cue stored for the management tasking.

The operating parameters of the sensors and mission then are used to determine each sensor's capability to collect the information in order to satisfy the requests. Each of the sensor models calculates expected measurement accuracies and probabilities of acquisition based on sensor capabilities, target parameters, geometric considerations (range, field of

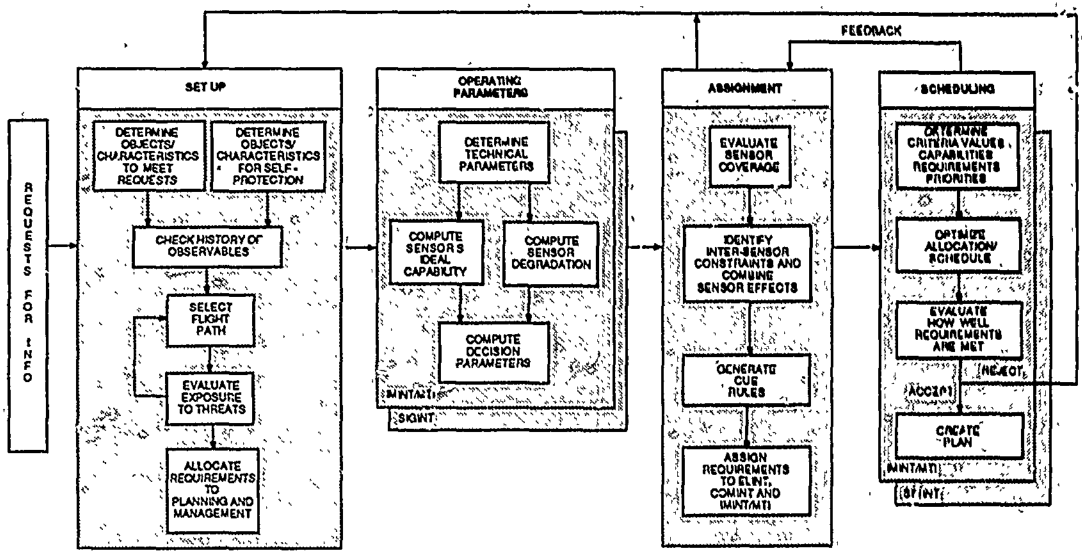


Figure 2. The functional flow for mission planning decomposes the process into terms that allow efficient, coordinated implementation of automated multisensor planning.

view), atmospheric conditions, and weather conditions.

These decision parameters are passed to the assignment phase, where sensor coverage is displayed to the user. In this phase, intersensor constraints are noted. These can be exclusive constraints (e.g., SAR may interfere with ELINT collection in a frequency range) or inclusive constraints (e.g., EO and IR may share optics so that they must be directed at the same area). Combined sensor effects, also calculated here, determine the cumulative probability of acquisition by more than one sensor. Each element of information with a probability of detection by a sensor above a minimum threshold is assigned to be collected by that sensor.

In the scheduling phase, optimized, detailed sensor tasking is computed. The algorithms for these computations consider criteria values that are based on the priority of the request, the capability of the sensor to collect the information, and the contribution of the information to satisfying the overall request. Tradeoffs for employing SIGINT or IMINT collection opportunities are made at this time. Initial requirements for the intelligence collection mission are then compared to the expected performance of the mission for the scheduled sensor tasking. The candidate plan and the evaluation of it are then presented to the user who can accept the plan, modify it or return to components of the setup or assignment phase to create a new plan.

System Simulation

The purpose of the simulation is two-fold: (1) to provide the developer with a vehicle for creating and analyzing new mission management algorithms, and (2) to aid the mission planner in evaluating courses of action as an element of the mission planning decision aid.

The system simulation encompasses two primary components: the modeling of intelligence collection and the responding to the perceived situation through the mission manager. The simulation of intelligence collection will be performed by the Lockheed Interactive Simulation for Intelligence Collection (ISIC), an Ada-based system that resides on a VAX. ISIC simulates the activity and movement on a tactical battlefield, and models the sensors' collection of intelligence information.

Because the primary goal of this project is to develop mission planning and management approaches, the simulation efforts will focus on the mission manager, the second component of the simulation. Figure 3 shows the architecture for a management function which will be implemented into the simulation. The figure depicts a baseline functional flow for information generation, processing, and mission management in a multisensor environment. Once the information has been received by the system, its handling can be segmented into three major categories: single-sensor processing, multisensor processing, and mission replanning. Sensor processing is the analysis of the input data itself and includes the individual processing functions of each sensor and processing for self protection. The mission management component is responsible for multisensor fusion and the subsequent generation of cues for more data when the information is insufficient to meet the collection goals. These cues, along with any external requests, update the collection objectives. The mission replanner monitors changes to the collection objectives and dynamically retasks the mission by reallocating sensor utilization, reconfiguring the flight path, and employing means for self-protection.

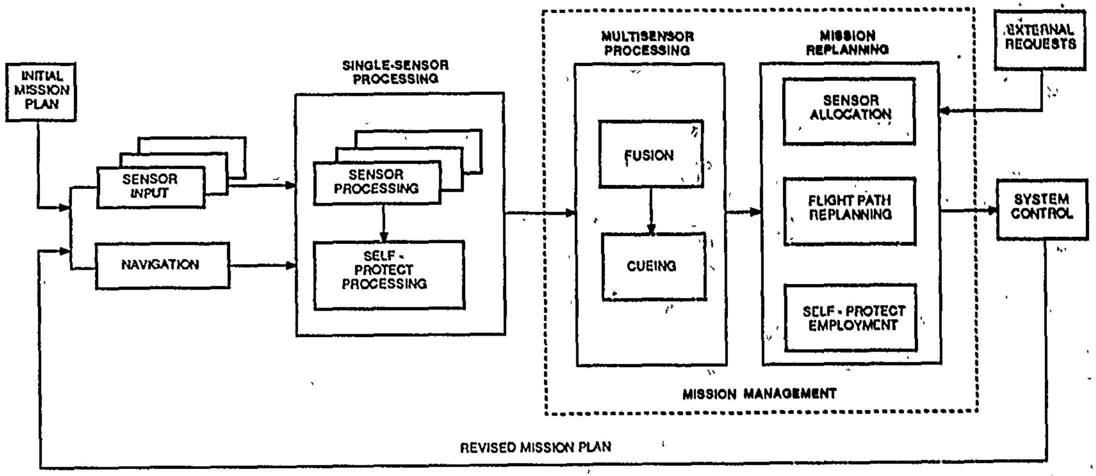


Figure 3. The functional flow for information-handling accepts an initial plan and dynamically performs single- and multisensor processing. Mission management components are multisensor processing and mission replanning.

System Evaluation

The effectiveness of a mission is determined by conflicting goals of survivability and collection efficiency. Three techniques for generating measures of effectiveness (MOEs) have been identified for consideration: (1) neural networks, (2) an expert system that generates rules from examples, and (3) utility functions. These MOEs can be used to evaluate and rank various planning and management options.

MOEs corresponding to platform survivability include minimizing platform observability to threats and active and passive responses to threat activity. It also includes the ability to execute the mission plan in the presence of nonlethal threats, including jamming and deception.

Collection efficiency encompasses several factors that relate to the operational usefulness of the information gathered, processed, and transmitted. The components of collection efficiency are:

- Number of desired objects detected/identified
- Quality/accuracy/relevance/timeliness of information
- Time required to collect desired information
- Volume of excess data collected.

A composite measure of effectiveness corresponding to collection efficiency attempts to maximize the first two of these components while minimizing the last two. These factors and their relative importance depend upon the operational scenario, the mission goals, and the time criticality of the information, and may cause conflicting goals.

IMPACT SYSTEM

The mission planning function described in the preceding section is realized in the Intelligence Mission Planning and Collection Tasking (IMPACT) system, a computer-based decision-aid that converts the currently time-consuming process of generating independent sensor tasking from mission directives into an interactive session that allows the user to rapidly generate, evaluate and modify candidate plans for multiple sensors.

Along with the collection requests, the planner specifies the platform characteristics, sensor capabilities, weather conditions and flight path way points. These are stored in data bases and are used to construct missions. A mission consists of a platform, flight path, sensor suite, AOIs (with associated objects of interest) and mission-specific parameters. The IMPACT system then develops optimized sensor schedules that integrate and coordinate the collection of information to satisfy the user's requests.

Sensors integrated into the IMPACT system are electro-optical (EO), thermal infrared (IR), synthetic aperature radar (SAR), moving target indicator (MTI), communications intelligence (COMINT), and electronic intelligence (ELINT).

The IMPACT system is written in Ada and resides on a VAX 11/780. The graphics is based on the Graphical Kernal System (GKS), using a Chromatics CX1536 colorgraphics engine. The textual displays use the DEC forms Management System (FMS), and the user interface is the Lockheed Guided Input (GI) menuing system, both of which run on a VT-100 (compatible) terminal.

Overview of Functions

The IMPACT system performs the following functions:

- Determines targets for IMINT and SIGINT sensors which will satisfy collection requests.

- Calculates a flight profile
- Evaluates platform survivability
- Evaluates sensor capabilities to provide desired information
- Allocates requirements to the different sensors
- Develops cueing rules to be used by the mission manager
- Creates an optimal schedule for the sensors
- Assesses how well collection requirements will be satisfied.

The following sections concentrate on analytic approaches to components of the IMPACT system.

Target Determination

IMPACT includes a data base of collection targets, each of which is decomposed, or flowed down, in a tree whose lowest-level elements can be associated with collection by a specific sensor type (IMINT or SIGINT). For example, Figure 4 describes an object flowdown for a SAM. When information on a SAM is requested, the planning system for a multisensor mission determines what the specific tasking assigned to each sensor will be: e.g., communications networks and acquisition radars to SIGINT, associated vehicles to IMINT. All leaves on the tree in Figure 4 become potential targets when a high-level request for information on SAMs is made. The requests can be made at any level on the tree, such as locating the missile system itself (e.g., SAM-x) or locating its acquisition radar.

In addition to the typical templating of targets as illustrated above, a logical structure is imposed on the targets in the tree. This refinement brings the object flowdowns to the level suited for consideration by a planning function. The three relationships between branches of a tree are *OR*, *AND*, and *XOR* (exclusive or). Simplified examples of these relations, which characterize these concepts are shown in Figure 5. The *OR* relation is a typical decomposition of an object into two components, e.g., a SAM can be identified by collecting its radar emission with ELINT or by imaging a specified chassis with an IMINT sensor as is shown in Figure 5(a). Two branches are related by *AND* when the concurrent collection of both components is required. For example, detection of both RADIO1 and RADIO2 is necessary for the detection of the communications network in Figure 5(b). An exclusive or relation, *XOR*, arises when one category of targets is segmented into two subcategories [Figure 5(c)] such that an object falls into one category or another, but not into both. When locating moving vehicles, it is desirable to locate both trucks or tanks (as it is for *AND*), although the collection can be partially satisfied by locating either trucks or tanks (as it is for *OR*).

This detail on the object flowdown is necessary to the planner of a multisensor intelligence mission since assessing the probability of satisfying a higher-level request depends on the logical relation between the branches under the object requested. In addition, the value of collecting an element of information, a critical element of the scheduling criterion, depends on the relationship of the information to the requested object.

Sensor Capabilities

Target acquisition and accuracy models for each of the sensors were developed to apply specifically to the unique needs of the planning process. The models accept available environmental data as input and are based on parameters which drive sensor tradeoffs. A brief overview of each of the models developed is given below.

Infrared (IR) model. The thermal infrared acquisition parameters for

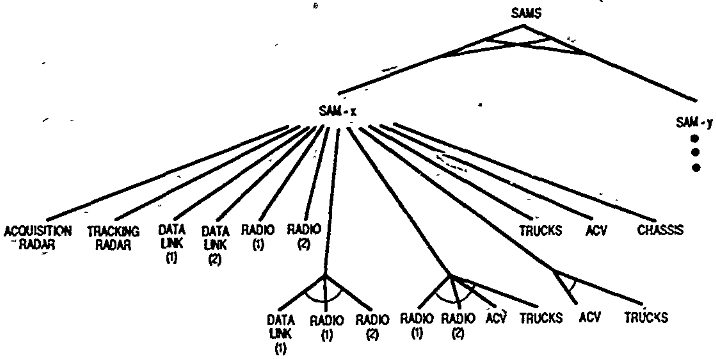


Figure 4. This example tree depicts the object flowdown for a SAM. When a request for information on a SAM enters the IMPACT system, the bottom elements in the figure are automatically identified as targets of interest to the IMINT and SIGINT sensors.

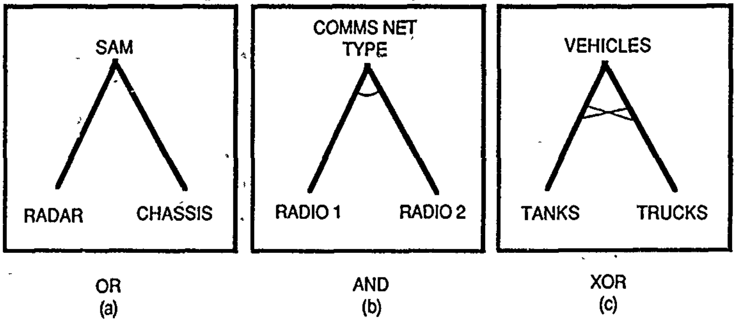


Figure 5. Logical relations between objects in a flowdown in the IMPACT system implement realistic scenario components. (a) Traditional OR relation: detection of either element of information, the chassis or radar, satisfies collection request. (b) AND relation: both pieces of the communication network must be detected to verify detection of the network. (c) XOR relation: detection of tanks or trucks satisfies requests for information on vehicles while detection of all tanks and trucks satisfies the request fully. Each moving vehicle is either a tank or a truck, but not both.

detection, classification, recognition, and identification are computed from two sets of calculations. The first is the probability that the signal-to-noise (S/N) ratio is sufficient for detection, P(S/N), and is based on the radiant temperatures of the object and background, sensor performance parameters, and transmission loss. Transmission loss depends on precipitable water, aerosol content, rain, and clouds. The second calculation is the probability that the object resolution is sufficient for acquisition, given detection of the signal, P(acq|S/N). These probabilities are multiplied for overall probability of acquisition: P(acq) = P(S/N) • P(acq|S/N).

Electro-optical (EO) model. The EO model is similar to the IR model in that probability of acquisition is computed by P(S/N) • P(acq|S/N). Signal-to-noise is computed as a contrast ratio from the radiances of the object and background, solar illumination parameters, and atmospheric transmittance. The latter term, P(acq|S/N), is calculated in the same way as the corresponding IR term. For both P(S/N) and P(acq|S/N) computations, the shadow of the object is considered separately and has its own parameters and dimensions as seen by the sensor.

Synthetic Aperture Radar (SAR) Model. The SAR model is similar to the IR and EO models in that probability of acquisition is computed by P(S/N) • P(acq|S/N). The latter term, P(acq|S/N), is calculated similarly to the corresponding IR term, except that the computation of resolution elements per target dimension is dependent upon range and azimuth resolution of the SAR sensor, as opposed to angular resolution for EO and IR. For P(S/N) we assume that S/N is constant for objects outside a minimum depression angle and within the maximum range of the sensor.

Moving Target Indicator (MTI) Model. The MTI detection model performs an initial screening to test whether the object's radial velocity is within the minimum and maximum endoclutter threshold of the sensor. If it passes the screening, signal-to-noise and signal-to-clutter ratios are calculated from the radar range equation, radar losses (see SIGINT model), subclutter visibility, and the apparent object cross section. These ratios then determine the overall probability of detection of the object.

In addition to these calculations is the computation of the maximum time between MTI searches in order to maintain coverage except for wedges of length specified by the user. This maximum time is used to interleave SAR and MTI when they are performed by the same radar.

SIGINT Model. The SIGINT model treats collection of signals in the traditional COMINT and ELINT ranges, and above. The detection model initially screens a signal to test if its parameters are within a sensor's acquisition capabilities, e.g., rf range, demodulation. An additional screening determines if the signal position is within the azimuth and elevation fields of view of the sensor. A signal-to-noise ratio is calculated based on the radar range equation and additional transmission losses. The losses depend on transmitter frequency and polarization, terrain, and atmospheric/weather conditions. The signal-to-noise ratio then determines the probability of detection.

Accuracy Models. Accuracy models were developed that compute precision parameters which affect the mission planning process. A linear error analysis was performed to calculate target-state error covariances and utilities designed to integrate these covariances into the mission planner.

Covariance analyses were performed for target position determination by EO, IR, MTI, SAR and SIGINT sensors and for velocity determination using MTI data. Significant input error sources that affect sensor tasking that have been modeled are sensor measurement errors, aircraft navigation errors (position and velocity), pointing errors of the sensor and aircraft, and height errors in digital terrain map data.

The standard covariance analysis technique employed permits the mapping of the covariance of errors encountered in the sensor's data collection process into a covariance of the errors in the target's position and/or velocity. Because the analysis is linear, it permits rapid parametric study of the effect of data collection errors on knowledge of the target state and therefore, on the mission planning process. The technique also allows the quantification of accuracies of multiple sensors.

Sensor Scheduling

This section describes the algorithmic approaches to detailed sensor scheduling that were developed.

The value of a given sensor collecting an element of intelligence is computed based on the priority of the request for that information, its contribution to satisfying the overall request, and the capabilities of other sensors to satisfy the request. After calculating value factors, the schedules are computed sequentially, beginning with IMINT since it is more constrained than SIGINT collection. IMINT is initially scheduled assuming that the SIGINT sensor schedule will satisfy all SIGINT objectives. A SIGINT schedule is then computed and any required tradeoffs are made, based on the actual IMINT schedule. If the resulting SIGINT schedule varies significantly from the assumption, iterations of this sequence are performed until convergence to stable IMINT and SIGINT schedules.

IMINT Scheduling. The solution to IR, EO, and SAR scheduling specifies where, on the ground, the sensors should be directed. The positions that optimize the amount of desirable information collected are those that maximize

$$V_o \cdot \int \int f_o(x,t) \cdot P_o(x,t) \ dxdt$$

where

V_o = value of the object

$f_o(x,t)$ = ability to collect information

$x_o(x,t)$ = distribution of the object.

At every point on the flight path, an optimized combination of IMINT sensor placement is computed by first discretizing the points on the ground perpendicular to the platform velocity and assessing the individual sensor coverages at each point. Then the ground positions that maximize the total effectiveness (according to the above equation) are found by a branch and bound search or by enumeration.

The schedules for SAR and MTI sensors are interleaved to satisfy continuous MTI coverage, where required. The time period required to achieve this coverage is based on geometry, sensor characteristics, and probability of acquisition. SAR and MTI usage may be limited if survivability considerations restrict the platform from actively radiating.

SIGINT Scheduling. The objective of SIGINT scheduling is to produce a frequency scan pattern that determines the order and repetition of frequency ranges that the sensor searches through as it listens for signals of interest. SIGINT scheduling is achieved in three stages: signal segmentation, allocation of frequency search slots, and sequencing of the frequency bands.

The *segmentation* process begins by calculating a metric, or distance, between each pair of signals of interest

(SOIs). The distance is based on the value of the signals, the differences between the frequency ranges of the signals, and between the requirements on the bandwidth and timing for intercepting the signals. It also reflects the signal density of the environment, in order to avoid unwanted signals as well as to provide work-off time for dense signals. A clustering algorithm then joins nearest (as defined by the metric) signals into frequency segments that will be included in the sensor schedule.

An optimum *allocation* of the total number of (not necessarily unique) segments allowed in the sensor schedule is then made by first calculating a measure of merit for each segment as a function of the number of occurrences. A linear program (LP) then globally optimizes the number of times each segment occurs within one frequency scan cycle. The nature of the constraints imply that a real-valued LP will provide integer solutions to the problem (a necessary constraint).

The third stage, *sequencing* the segments into a final frequency scan pattern, maximizes the separation between occurrences of the same segment. A heuristic approach has been developed that appears well-suited to finding the optimum or near-optimum solution. However, an approach that combines number theory with simulated annealing has been identified and will be investigated in the future.

Because each of the three stages depends on the others, the implementation will include an iterative scheme that converges to an improved solution.

CONCLUSION

Concepts have been developed for the three components of an analysis system for evaluating multisensor mission planning and management: plan development, system simulation, and evaluation. Software has been developed for the planning component through the IMPACT system.

The current status of IMPACT is an operational system that allows the development of missions, object requests and database elements. Acquisition probabilities for the targets which are identified as satisfying requests are calculated, displayed and passed to the scheduling processes. Initial scheduling algorithms have been integrated into IMPACT and are undergoing testing and enhancement.

Future work will continue to refine the architecture for a real-time mission manager and will identify algorithms for implementing the architecture.

INTEGRATION AND EXPERIMENTATION WITH COOPERATING EXPERT SYSTEMS, NATURAL LANGUAGE PROCESSING, AND ALGORITHMIC TECHNIQUES TO PERFORM CORRELATION AND DATA FUSION IN A TACTICAL ENVIRONMENT

James B. Nelson

Lockheed Austin Division
O/T0-10, B/30F, 6800 Burleson Road, Austin, Texas, 78744
(512) 448-5458

INTRODUCTION

Many existing tactical command, control, communications, and intelligence centers are severely limited in their ability to process automatically the large volume of multiple-format, multisource message data they receive. Currently fielded systems lack the processing sophistication that is needed to reduce this voluminous amount of data into useful, manageable information. This shortfall in processing capability of current systems has stimulated the development of new, improved, state-of-the-art processing techniques that are more accurate, save time, and provide the battle manager with useful, reliable, automated tools.

Purpose

The purpose of this project was to investigate, develop, demonstrate, and evaluate the application of advanced processing techniques (automatic message processing, algorithmic correlation/data fusion, natural language processing, expert systems, and relational data base interfaces) to the domain of tactical intelligence processing and analysis and to examine the timely use of those data by a tactical battle manager.

Experimentation

Five specialized vignettes were created (by domain experts) and added to a 1990s Fulda Gap war scenario. These were input to the Testbed system (1) to determine the effectiveness of the algorithmic and heuristic approaches to correlation/data fusion, (2) to observe how well the Natural Language Text Line Processor (NLTLP) and Expert Systems (ES) performed, and (3) to ascertain if a dynamically created and updated intelligence Order-of-Battle (OB) could be used automatically by a Weaponeering (Targeting) function of a tactical C3I center.

Vignette I. Delineation of Unit Boundaries. This vignette is structured to capture the knowledge an analyst applies when trying to determine the area of interest of a threat division, the forward line of troops, the boundaries between forces adjacent to the division, and the boundaries of subordinate regiments. Recognition precepts include inferences on air defense assets, critical node location, and artillery command and control.

Vignette II. Recognition of Air Defense Artillery Assets. This vignette focuses on tactical deployment of division air defense assets, their electronic signatures and communications links. It deals with the thought processes that an analyst uses to identify those assets, locate them, and track them. Recognition precepts include target acquisition battery inferences, echelon and organizational inferences, relative location inferences for emitters, and overall deployment inferences.

Vignette III. Identification and Location of Command Posts. This vignette is designed to activate the thought process of the analyst when he/she identifies, locates, relates, and interprets the significance of Command Posts. Recognition precepts include signatures, geolocations, organizational data, interactions, and tactical inferences.

Vignette IV. Recognition of Attack Activities. This vignette provides the stimulus necessary to activate those ES rules that deal with recognition of the type of combat activity about to unfold. Recognition precepts include communications activity, special equipment, frontage size, and asset deployment inferences.

Vignette V. Enhancement of Algorithmic Correlation. This vignette is designed to stimulate the refinement, assessment, and accuracy of OB Database files of emitters and units. It permits the ES, which contains multiple hypotheses about emitters or units, to recorrelate questionable decisions made by the algorithmic correlator.

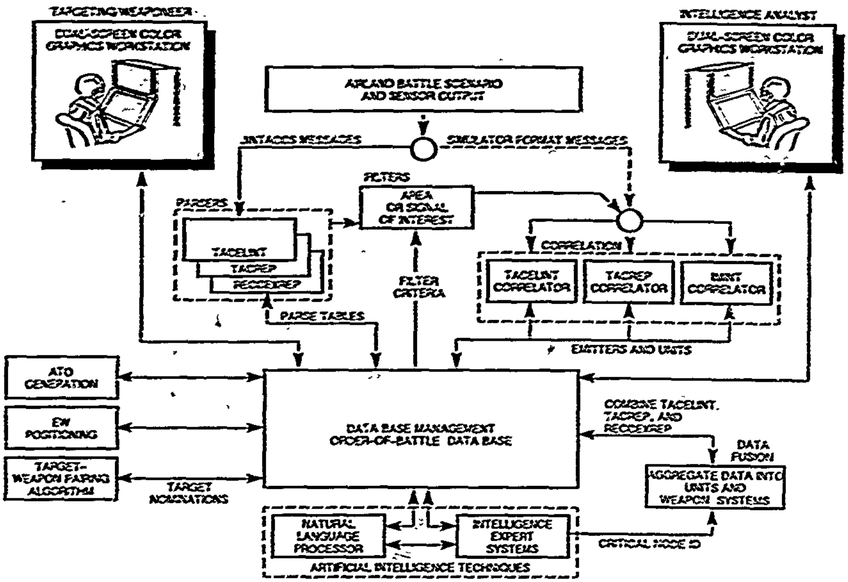


Figure 1. Integrated Testbed Architecture

SYSTEM ARCHITECTURE

The overall system (C3I Testbed) is a fully concurrent, multiuser information processing system where processing is divided into functional blocks with well-defined interfaces between modules. The Testbed functional architecture (Figure 1) permits experimentation and demonstration of integrated technologies as applied to the intelligence and weaponeering functions of a tactical C3I center.

Fully formatted JINTACCS messages are input to the Testbed by an AirLand Battle simulator, which uses a 1990s Fulda Gap war scenario (enemy events and OB) and a variety of airborne sensor models. The Testbed automatically parses (decodes) and filters the JINTACCS messages according to tables that are set and easily changed by the Intelligence Analyst. Two algorithmic correlators (statistical and figure-of-merit based) perform automatically to read the message data and to create and update emitter files within an OB Database.

A Data Fusion function examines the emitter files created by the correlators, adds the information produced by the artificial intelligence (AI) techniques, then combines the data to form units and weapons systems within the OB that represents the most complete information about entities on the battlefield.

The AI techniques include the NLTLP and distributed cooperating ES. The NLTLP reads and interprets the English language text lines of the JINTACCS messages, reformats the data, and passes the results to the ES. The ES examines the OB Database and the NLTLP output to determine the presence of certain types of units on the battlefield, their geographic operating boundaries and activity state. Additionally, the ES recognizes and corrects certain types of errors not handled by the algorithmic correlators.

The Weaponeering/Targeting function automatically accesses the intelligence OB Database, compiles target lists, and calculates target-weapon pairing solutions based on a global optimization algorithm.

An imagery processing capability is in the planning stages at this time.

An Air Task Order (ATO) generation capability is in the development stage.

The system architecture permits a wide range of experimentation possibilities for the testing of specific modules, algorithms, or fully integrated components.

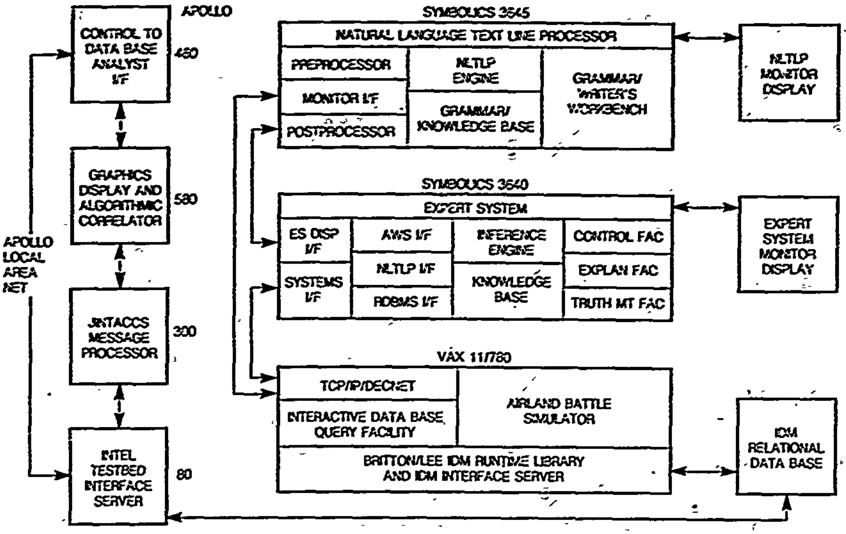


Figure 2. Hardware/Software Modules and Interfaces

Hardware/Software Implementation

The approach used to implement the C3I Testbed System is characterized as a balanced integration of conventional processing technology with AI processing techniques in a distributed environment. Each technology is applied to the intelligence analysis and targeting problems in the manner deemed most appropriate. This approach does not attempt a total AI solution nor does it attempt to force AI techniques on aspects of the problem that have already been solved by algorithmic methods. Figure 2 shows the hardware connectivity and the software major module integration of conventional algorithmic processes with AI heuristic processes.

Distributed Processing Interfaces

This project emphasized the use of distributed processing techniques as a way to achieve fast and useful interfaces between the various workstations and processes These interfaces permit the ES, NLTLP, algorithmic correlator analyst displays, Weaponeer, and data base management functions all to reside on separate processors, yet be accessible by any. Two primary classes of interfaces are used: RS232 software interface and TCP/IP software In addition to vendor-developed TCP/IP system software for specific machines, we developed a remote procedure-call applications layer for interface between the Apollo and Symbolics machines.

Software Tools and Programming Languages

The Testbed System uses a variety of software techniques to make the system work. These are summarized as follows:

Simulator: Ada, Vax 11/780

Expert Systems: Common Lisp, Automatic Reasoning Tool (ART) ES Shell, Symbolics. ART was chosen for its graphics, rule definition and compilation, confidence levels, and viewpoints (blackboard) capabilities

Natural Language. Common Lisp, Language Craft NLP Shell, VAX, and Symbolics. Language Craft was chosen because it is a general-purpose, natural-language processing tool, it permits the developer to concentrate on vocabulary and semantics rather than mechanics of parsing and semantic interpretation, it has a well-defined interface to Common Lisp, and runs on both VAX and Symbolics.

Map Graphics: C, Apollo (also runs on Silicon Graphics, Sun, MicroVAX) Digital map graphics display technology (SoftCopy Map Display System) contains DMA data (DTED, DFAD) and World Data Base II and has user friendly software that permits extensive map manipulation and control.

User Interface Manager. C, Apollo (also runs on Silicon Graphics, Sun, MicroVAX) System permits creation/change of displays in minutes by a nonprogrammer

Correlators: FORTRAN, Apollo

Data Base: IDL and Progress Relational DB Access, Britton-Lee.

Message Processing: FORTRAN, Apollo.

ALGORITHMIC CORRELATORS

Both the Tactical Electronics Intelligence (TACELINT) and the Tactical Report (TACREP) correlation algorithms function automatically within the Testbed message-handling architecture. Each has a set of specialized interactive tools that permit an analyst to control automatic correlation variables: review, edit, or change the results of automatic correlation and make manual correlations.

TACELINT CORRELATOR

The TACELINT Correlator performs the following primary functions:

- Attempts to correlate signal-of-interest messages to existing emitter OB Data Base files automatically
- Updates emitter OB Data Base files automatically
- Sends graphic updates to the map graphics display (analyst workstation).

The TACELINT algorithm is based on a scoring and weighting system that calculates the statistical differences between the emitter parameters contained in a message and the parameters of emitters held in the data base that fall within a tolerance gate set by the intelligence analyst. Those OB Data Base emitters that fall within the tolerance gate are considered to be reasonable candidates for correlation. Each parameter of each emitter is scored and weighted (by any criteria as determined by the analyst, e.g., sensor measurement, time delay, etc.). Next, each candidate is tested for geopositional feasibility and scored. Total scores are then calculated for each candidate versus the message emitter. Finally, the algorithm compares each candidate. If a single best candidate (heuristically determined by the analyst and input to the algorithm) is identified, a correlation is made. Otherwise, candidates are ranked and filed in the OB Data Base as potential (ambiguous) correlations. Each ambiguous emitter is reconsidered as new messages are input to the TACELINT correlator.

TACREP CORRELATOR

The TACREP correlator uses parametric association and attribute-matching techniques to correlate message data with emitters held in the OB Data Base. Automatic correlation occurs when certain combinations of data elements in the message match exactly or within tolerances with the data elements contained in the data base. If a unique identifier is present in both the message and the OB Data Base, if the candidate meets a geopositional feasibility test, and if the candidate meets a scoring threshold (set by the analyst), a correlation is made. Those attempts at correlation that do not meet the minimum score result in the message data being returned to a message file for future attempts at correlation. The TACREP correlator does not create ambiguities.

DATA FUSION

The Data Fusion function uses the emitter OB Data Base files, previously created by the TACELINT and TACREP correlators, to create and maintain weapons systems and units in the OB. It also uses the data provided by the ES and the analyst workstation.

Data Fusion performs automatically, making inferences about the battlefield. At the first level, inferences proceed directly from a single fact, e.g., the existence of a weapons system can be inferred directly from the existence of certain emitters. Some unit types can also be inferred in this manner. A second level of inferences occur when inferences are made based on several facts that do not imply anything separately, but when considered together imply something about the OB. For example, the existence of several different entities, located within a certain distance of each other, or the determination of a parent-child relationship between two or more entities, can imply much more than the existence of a single entity.

The Data Fusion function performs the following functions:

- Groups entities into systems
- Determines battlefield positions for systems based on locations of component parts
- Deduces the presence of units from the deployment of specialized equipment
- Assigns systems and equipment to the controlling unit
- Determines battlefield unit positions
- Determines unit organizational hierarchy.

The Testbed System uses tightly coupled OB Data Base, Correlation, and Data Fusion functional modules as a complete OB management system. Each of the automatic processes is responsible for maintaining its specialty data in the central data base and on the map display. All changes to the OB, whether initiated by the analyst, ES, Data Fusion, or Correlator, maintain the integrity of the OB Data Base. For example, the parametric details of TACELINT correlation may be unknown by Data Fusion, but if the emitter had moved, the communication of that fact to Data Fusion would cause the Data Fusion algorithm to recalculate the position for the unit containing that emitter.

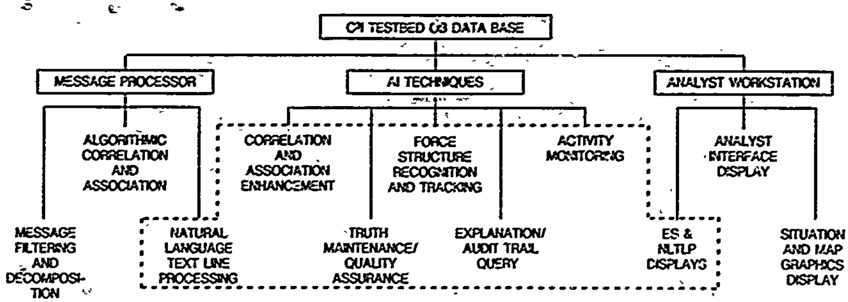


Figure 3. AI Techniques Integrated with Testbed System

ARTIFICIAL INTELLIGENCE TECHNIQUES

Seven component subsystems using AI techniques were integrated into the Testbed System. These components are organized as illustrated in Figure 3. The ES includes three cooperating subexperts and two global support functions.

Correlation and Association Enhancement

The Correlation and Association Enhancement (CAE) subexpert looks for constraint violations in the output of the algorithmic correlators, directs the correlators to recorrelate the associated messages, and communicates with the correlator in a manner functionally similar to that of the human analyst. The CAE works in conjunction with the FRT to perform constraint-based correlation and data fusion. Constraint-based correlation is performed using constraints along one or more of several different problem space dimensions (or classes). Parametric, Geopositional, Doctrinal, Terrain, Activity, and Multisource. CAE is closely coupled with the algorithmic correlator and has a feedback loop with it to pass information on recorrelation and change of OB Data Base.

Force Structure Recognition and Tracking

The Force Structure Recognition and Tracking (FRT) subexpert accomplishes the following: hypothesizes force structures from the association of emitter reports; determines the types of units represented using doctrinal, table of organization and equipment (TO&E), and analytical knowledge, tracks the movement and confidence associated with the force structures and units. The FRT formulates and compares the positional clusters formed by individual emitters, determines subclusters, and examines the relationships between clusters to form an opinion of the existence of certain types of units. This subexpert also compares the total OB known to exist within an area of interest to doctrinal data to determine if there is agreement between the perceived and doctrinal OB. If discrepancies are found, they are reported to the analyst. In addition, the FRT performs specific unit identification, including echelon and subordination.

Truth Maintenance

The Truth Maintenance Facility maintains the interrelationships between objects and permits the system to backtrack whenever a given object characteristic or occurrence is found to be in error or is contradicted. These interrelationships are derived from the endorsements of each object maintained for reasoning under uncertainty and allow for chaining from one object and fact dependency to another.

Explanation Facility

The Explanation Facility provides an English-like explanation of the reasoning process behind each decision made by the three subexperts. This facility responds to analyst requests for explanation regarding any emitter, unit, or activity indicator. The explanations are derived from the endorsements of each object maintained for reasoning under uncertainty and include an audit trail of the inferences performed rather than just a list of rule names. Each rule included within any of the ES subexperts is coded with a brief English-language explanation of the rule function, and when a rule fires, an endorsement is created for the object of interest and is stored with the object. By using the accumulated endorsements as the basis for explanations, a separate audit trail does not have to be maintained and overall system consistency is ensured

Activity Monitoring

The Activity Monitoring (AM) subexpert recognizes activity indicators associated with the units in the OB, accumulates warning signs, and posts alerts to the analyst for indicators of attack. It uses the output of the Natural Language Processor and the output of the algorithmic correlators. Activity indicators are accumulated by activity category and by unit type involved. The AM uses a hierarchical structure of indicators, warnings, and alerts so that any type of activity can be included. If multiple indicators occur for a given activity or unit type, a warning is provided to the analyst. If multiple warning are created regarding a particular activity type, an alert is created and issued to the analyst.

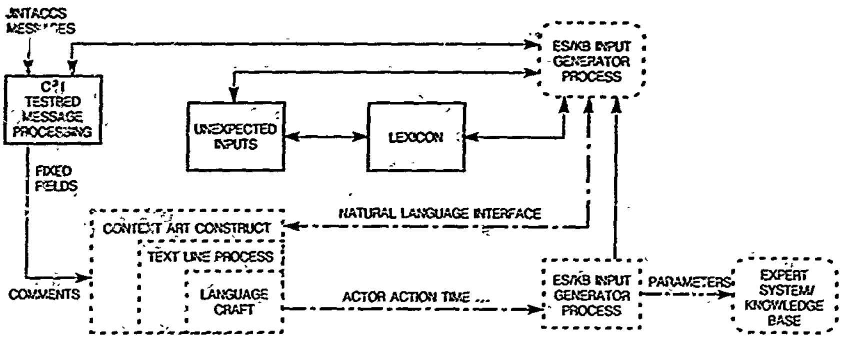


Figure 4. Natural Language Functions Integrated with Testbed System

Natural Language Text Line Processor

The NLTLP was designed to combat the difficulties of interpreting free text and creating a usable form of data for the ES. The most important goals were to examine the techniques needed for processing free text and to create an automatic process to accomplish this. The techniques explored were the caseframe grammar used, along with discourse analysis techniques such as using a context mechanism, to complete missing values. These techniques were written in Common Lisp, along with the grammar written for use with the Language Craft parser. The techniques were initially implemented as a automatic process, however, an entirely automatic process does not appear to be the best solution because ambiguous (multiple) parses can occur unless the grammar is constructed very carefully. Ambiguous parses can be presented to the analyst to resolve, saving computer time. The integration of the NLTLP within the Testbed System is shown in Figure 4

NLTLP Functions and Design

The Natural Language Text Line Processor accepts the fixed fields and free-form text lines of both TACELINT and TACREP messages, parses the free-form text, and interprets the meaning of this text in the context of the fixed fields and other text lines. The results of this processing are records that can be processed and interpreted by the ES. The NLTLP includes preprocessing functions to handle abbreviations, acronyms, and simple misspellings It handles a variety of activity types, verbs, nouns, and modifiers. It can distinguish tenses and can correctly interpret time and place information The NLTLP uses a caseframe grammar approach to parsing, and semantic interpretation makes the NLTLP easy to extend to new reports, activities, objects, and modifiers.

Preprocessing of Free Text

Preprocessing is done on text lines before they are parsed by grammar. Several preprocessing steps are used

- Message handling with and without text
- Breakdown of multiple text lines into individual text lines
- Preparation of each text line (acronym expansion, separation of numbers and characters in time and location fields, addition of case markers to enhance parsing, and handling of abbreviations not allowed by Language Craft.

Components of Grammar

The grammar used for the NLTLP processing is the caseframe grammar that is used with the Language Craft parser (Plume), consisting of the four main components caseframes, rewrite rules, irregular morphology rules, and abbreviation rules.

Postprocessing Rule

Postprocessing rules operate on the instantiated caseframe generated by the parsing of a text line Three forms of postprocessing rules occur

- Mapper Rules. These are used to transform the caseframe instances output by the parser into caseframes for input to the LISPifier rules
- LISPifier Rules These are used to convert mapper output into LISP forms suitable for a particular application.
- Context Resolution Rules A context schema is created from analyzing each text line, and if a resultant schema is missing information, previous schemas are checked to fill the missing values

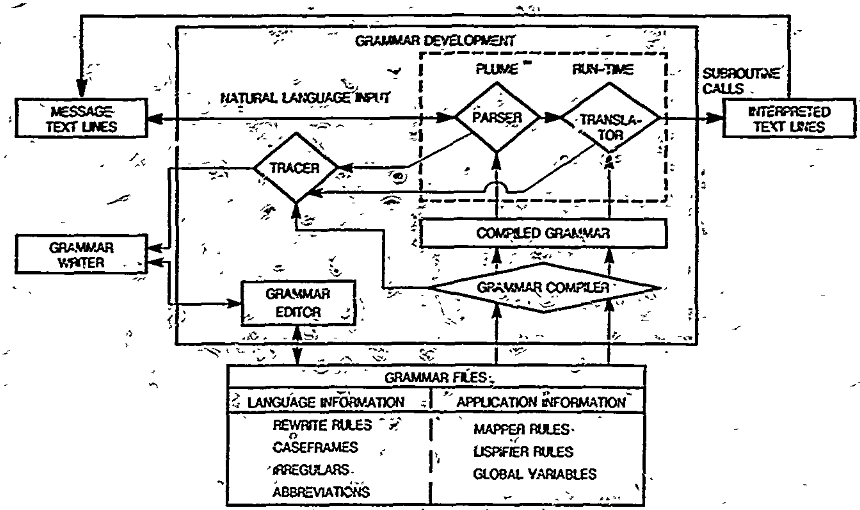


Figure 5. Natural Language Text Line Processor Functional Modules

Figure 5 provides an overview of the NLTLP functional flow.

AI Techniques Control Mechanisms

To achieve a balance between control (reduce the time spent responding to a particular input) and flexibility (spontaneous handling of expected and unexpected inputs), the ES design consists of the following: cooperating subexperts, use of an endorsement method for reasoning under uncertainty, and a mixed-initiative user interaction.

- Cooperating Subexperts: Primary control was established in the FRT subexpert because algorithmic correlators have already processed much of the data. However, the system uses trigger events to assist in transitions between subexperts so that certain events cause primary control to switch to the appropriate subexpert.

- Endorsement Method. This method captures endorsements or votes for or against a particular conclusion, which is then reviewed, summarized, and assigned a confidence level. As votes are collected, a record of the support for the typing of each object is kept, and the evidence is increased or decreased as appropriate. Endorsements also serve as a basis for constructing explanations of system decisions and for truth maintenance.

- Mixed-Initiative User I/A: This approach to ES control allows both the analyst and the ES to direct the flow of processing.

Relational Data Base Interface

Relational data bases provide an especially flexible organization system for storing large volumes of data and data relationships in tabular form. Interfaces between relational data base management systems and AI systems, like the NLTLP and ES, permit the latter to retrieve data and information automatically and selectively and store the results of the processing for later use by both systems and operators.

The relational data base interface allowed the NLTLP and ES to share a common relational data base for the OB with the algorithmic correlators and the analyst interface. It allows the ES to submit data base queries autonomously and update requests to the management system.

Analyst-Machine Interface

The project demonstrated the use and integration of both AI and conventional technologies for the analyst-machine interface. The interface includes a menu-based query-and-command, interactive map graphics, and an ES monitoring-and-control function. For the ES, the analyst can ask for descriptions of objects in the OB and explanations for their existence. The correlation analyst can query, move, change, and ask for unit hierarchy or parametric data, uncertainty ellipses, unit or emitter histories as well as perform other analytic functions.

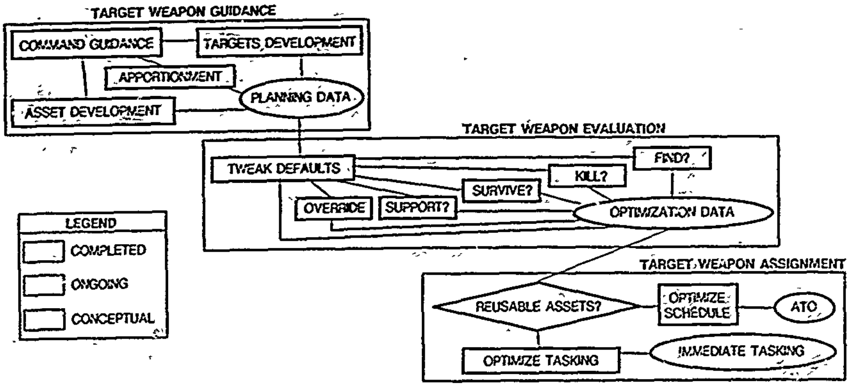


Figure 6. Target-Weapon Decision-Aiding

WEAPONEERING/TARGETING FUNCTIONS

The Testbed System intelligence analysis functions were integrated with a battle management function, targeting and weaponeering. This interface permits the battle management functions to access a dynamically created and updated intelligence OB, as it is happening. During system tests, we were able to observe the input of message-containing emitter data from the simulator into the Testbed Parser and Filter, Correlator, Data Fusion, through the NLTLP and ES, into the OB Data Base, and finally, watch the data appear as a target at the weaponeering/targeting analyst display.

Target-Weapon Pairing

The Target-Weapon Pairing (T-WP) algorithms determine the optimum allocation of weapons systems on alert (or with short response times) to an array of targets that have been selected. The allocation is based on the OB, weapons information from an Assets Data Base, the Single-Sortie Probability of Damage from the Joint Munitions Effectiveness Manual (JMEM), and the analyst's input of weights to be given for such factors as attrition, weapon system cost, and scarcity. The T-WP algorithms pair available weapon resources to a list of prioritized targets based on a criteria function, the number of weapons available, and the number of weapons needed to attack at the desired level of effectiveness. The weapons/targeteer analyst can chose targets on the OB map display using the cursor or by inputting target identifiers. Solutions are automatically generated and displayed. They may be saved while new criteria are entered for a different solution, compared, and either rejected or accepted. The algorithm handles up to 25 targets simultaneously to pair up with 12 weapons systems standard conventional loads.

Target-Weapon Scheduling

The scheduling algorithm determines the best way to load and assign weapon systems to all targets in a given Air Task Order (ATO) generation cycle. It compares available weapon assets with a target list based on multiple sorties per weapon and the cost effectiveness of all target-weapon pairs.

The algorithm maximizes the sum of the cost-effectiveness values for the target-weapon pairs subject to the following constraints number of available weapons, the minimum time from take-off for an aircraft from a given weapon set, number of sorties an aircraft can accomplish in one day and its cycle time, windows for each target, and weather for both the aircraft home base and the target. The approach avoids the trap of assigning the best weapons to the highest priority targets by detecting anomalies or special cases in the total target-weapon-time matrix for the scheduling period.

Joint Munitions Effectiveness Manual

JMEM methodology, along with user functions that permit problem specification and standalone execution of the algorithm, were implemented in the system The JMEM algorithm calculates the weapon-effectiveness estimates of the single-sortie probability of damage and number of weapons needed to achieve the probability of damage for a given target type and weapon combination.

Target-Weapon Pairing System, Aided Decisions

Under development within the Battle Management laboratory is a rapid prototyping and evaluation of decision-aids that would further automate the battle management functions of tactical C3I centers of the future Figure 6 illustrates a part of the decision-aiding necessary to advance the automation of this cycle.

Many existing tactical command, control, communications, and intelligence centers are severely limited in their ability to process the large volume of multiple-format, multisource message data automatically. Currently fielded systems lack the processing sophistication that is needed to reduce this great quantity of data into useful, manageable information. This shortfall in processing capability of current systems has stimulated the development of new, improved, state-of-the-art processing techniques that are more accurate, save time, and provide the battle manager with useful, reliable, automated tools.

The most significant operational conclusion regarding the application of artificial intelligence technologies to tactical intelligence processing and analysis is that these technologies, including the ES and NLTLP, can significantly enhance the support provided to the analyst over conventional methods if they are applied to the more abstract and less quantitative portions of the problem domain. These capabilities complement proven conventional and algorithmic approaches and provide an intelligent assistant for the analyst in the areas of correlation enhancement, text processing, and intelligent data base interface. The NLTLP and ES provide a faster method of handling high volumes of messages because they can reason about incomplete and uncertain data, provide inferences at multiple levels, and maintain multiple contexts.

The Testbed System demonstrated that analyst expertise can be embedded in a system. The ES knowledge base includes the doctrinal, TO&E, parameter data, and analytical expertise used by the analyst to process tactical intelligence reports. The ES inference engine uses this knowledge to perform analyses, reason about uncertain data, and present (display) the current tactical situation. The NLTLP knowledge base includes the grammar, vocabulary, and contextual associations that must be understood by an analyst to interpret these reports.

The use of a relational data base interface significantly reduced the analyst workload by automatically retrieving and updating data and information from data bases for use by the ES and NLTLP.

Natural language processing can extract useful information from free-form text, associate this information with the formatted data (report context), and format it for use by the ES. The NLTLP processes messages much more rapidly and accurately than human analysts. The use of caseframes to support natural language processing permits smooth integration of an NLTLP and ES.

Advanced and conventional techniques can be interfaced, can use a common analyst-machine interface, and can be implemented in a distributed environment that shares a common data base.

The development of customized knowledge bases, expert systems, natural language processors, and interfaces for each user or for each tactical unit is not necessary. A generic model of knowledge and metaknowledge can be developed that will serve as a basis for supporting the majority of tactical analysis needs.

Most specific knowledge in a knowledge base can be stored in frame structures and general rules rather than using specific rules. The frame, a flexible record-like structure with slots (fields), relationships, and attached procedures, can be used to represent most of the specific knowledge in a knowledge base, and general rules can be coded that reference entire classes of patterns rather than very specific patterns and data values.

A single deterministic method (endorsement) can be used as the basis for reasoning under uncertainty, truth maintenance, and explanation.

The Testbed System demonstrates a prototype of interoperability and integration of intelligence analysis (preparation of the OB) and weaponeering (targeting) functions of a tactical C3I center. This prototype indicates a significant time savings in generating targeting solutions through the use of an automatic optimization scheme and the use of direct interface with the intelligence OB Database.

FUTURE DIRECTIONS

Future research issues concerning the use of advanced processing techniques for tactical C3I centers can be outlined as follows:

- Automated knowledge acquisition and knowledge base extensibility, particularly for field use
- Cooperating distributed ES
- Extended natural language processing with human interfaces and contextual explanation generation
- Sensor-cueing based on situation assessment stemming from ES such as the Activity Monitoring and Force Recognition and Tracking
- Terrain-based geofeasibility extension to enhance correlation accuracy
- Extended automation and netting of Battle Management functions with intelligence functions.

Section V

Testing and Evaluation of C^3

Working Group Chairman:

Dr. Stuart H. Starr
The Mitre Corporation

A FRAMEWORK FOR ASSESSING THE DoD C3 PROGRAM

Dr. Paul A. Torelli

Center For C^3 Systems, DCA

1. INTRODUCTION

This paper describes some key methodological aspects of an assessment effort that looked at the entire DoD C3 program. The intent of the effort is to develop a summarizing document that articulates key issues and fiscally constrained choices regarding investment strategies in C3 improvement. The audience for this document is acquisition decision makers at the highest levels in DoD. Because of the lead times needed to develop, implement, and field technologically complex systems, a time frame of 15 years is used in this work.

The issue is framed as "What can the overall DoD program afford?" Any concept of affordability has three elements: a fiscal constraint, a shopping list, and price tags on each element in the shopping list. Each of these three elements entails a separate type of analysis: funding analysis, capability analysis, and cost analysis, respectively.

The purpose of funding analysis is to develop a set of estimates of the funds that will be available over the next 15 years to invest in C3 improvements. More precisely, the analysis develops a 15 year projection of current trends and then examines the effect of varying (positive and negative) growth rates on this trend. If the boundaries of C3 are defined fairly broadly to include a large portion of intelligence, than current annual expenditure is about $45 billion. The trend as of FY-87 is for about a 3% annual growth after inflation. This figure needs to be further broken down into four categories each of which should be managed differently:

(1) Over half of this money is spent to operate, maintain, and replenish the existing C3 infrastructure. This includes procurement to replace obsolete, non-supportable, or destroyed systems with new equipment having comparable functionality. The issues addressed in managing these dollars focus on cost-effectiveness.

(2) A second category is basic research and long lead developmental research. This involves over a hundred program elements in C3 alone, each of which is a few million dollars. These should not be closely managed at high levels of DoD beyond setting an overall funding level.

(3) A large category of programs were regarded as fenced for the purpose of this effort. These primarily consisted of C3 for SDI and of black programs.

(4) The fourth category, which is labeled "enhancements," consists of dollars to acquire, operate, and maintain C3 systems intended primarily to improve warfighting capability either through significantly new functionality or significantly improved survivability.[1] Although cost-effectiveness is considered in system design, the key high level issues focus on the value of the warfighting improvement in the context of overall defense priorities.

The fourth category, which accounts for about $8 billion annually, is the portion of the C3 program managed by mission oriented analysis (MOA) and is essentially the decision space for this affordability effort. It should also be noted that the 3% growth trend is concentrated in the third and fourth categories.

Capability analysis develops a mapping between C3 enhancement programs and warfighting capability. Having developed this mapping one can develop an "unconstrained" prioritization of C3

1. Deciding what is "significant" derives from capability analysis described below. There are close calls on this issue, but there is a surprising degree of consensus in the C3 community regarding which systems belong in this category.

enhancements, based on defense policy priorities, Services' doctrine, and contingency and operations plans. The priorities are unconstrained in that they do not account for cost and fiscal constraints. This effort did not attempt an independent capability analysis. Rather, it developed, over a five year period, a summary of several hundred plans, architectures, and mission analyses developed by CINCs, Services, and defense agencies encompassing in the aggregate strategic, theater level, and tactical C3. This summarization required developing a framework to structure the problem. Sections 2 and 3 of this paper describe framework development.

Cost analysis develops estimates of the 15 year price tag on C3 improvements. A key observation is that although there are hundreds of C3 improvement programs, there are approximately two dozen programs that represent over 70% of total cost of C3 improvements. Thus, one way of scoping this effort down to a manageable size is to focus the detailed cost analysis on these systems and assume that their portion of the overall program remains roughly stable. Aside from scoping down the number of systems for which estimates are needed there is an additional advantage that the larger systems usually have more complete documentation needed for cost analysis. The cost analysis includes not only research and procurement, but also operations, support, and personnel. A case by case analysis is needed to determine how much of the latter costs are offset by savings on existing systems. The cost analysis needs to be done at a sufficient level of detail to estimate the effects of increasing or decreasing the size of a buy and the effects of inefficiencies introduced by stretching out programs.

The remainder of this paper addresses developing a framework for capability analysis. Two key aspects of such a framework are the notions of hierarchical structure and of levels of capability. Both of these are central to any mission oriented analysis and are discussed in the next two sections of the paper.

2. HIERARCHICAL STRUCTURES

One of the most fundamental aspects of C3 analysis is its hierarchical structure.[2] These hierarchies are needed to relate technical improvements in the form of new radios, computers, or sensors to improvements in the ability to fight and win wars. The largest war-gaming and computer simulations either available today or envisioned can quantitatively analyze only very small parts of the total problem and cannot analyze how these parts fit together into an aggregate assessment. To look at the aggregate problem requires qualitative analysis which in turn requires a well thought out hierarchical framework.

Figure 1 shows the MOA process itself as a hierarchical methodology. The concept of levels of capability indicated in the figure and examples of levels are described in the next section. The hierarchical structure starts with broad goals for overall military capability as articulated by OSD and OJCS. These goals are translated into warfighting strategies by theater CINC's. These strategies, in turn, imply goals in each mission area (e.g., air defense, ASW, fire support). Down to this level all statements have described what the forces need to be able to do. It is at the lowest level of the chart that information needs are addressed and the resulting performance and survivability of the systems, facilities, and procedures for collecting, processing, and transmitting information. Figures 2, 3 and 4 present more detail on the theater, mission, and C3 capability tiers.

Figure 2 shows a theater and mission breakdown which has been used in presenting assessment results. This structure breaks C3 into three broad areas. Strategic C3 manages global nuclear war. Theater level C3 addresses C3 for echelons above Corps in conventional and theater nuclear conflict. Tactical C3 addresses C3 for lower echelons. For each of these three portions, the structure first shows a theater of conflict. Then the level of conflict or phase of conflict establishes the threat to both forces and C3. The mission levels describe own force action. For theater level echelons the mission is broad force management with specific nuclear responsibilities. For strategic and tactical portions, the missions are more recognizable. Figure 2 is not intended to be complete in representing all missions in all theaters. It does represent over 80% of the dollars that are being or might be invested in C3.[3]

There is an important underlying theme in the choice of the structure in Figure 2

2. Hierarchy here refers to the structure of the analysis and <u>not</u> to military echelons of command.

3. It does not make a statement about defense priorities since there are small dollar, high priority areas (e.g., special forces).

and in all of the remaining figures in this paper (including the next section). These structures are intended to articulate the effects of investments in C3 systems. To do this, one should be able, through shading or coloring or some other graphic arts mechanism, to state in which areas C3 is in good shape and in which areas C3 is poor. In order to do this the structure needs to be set up so that each box can be characterized as embodying unique C3 requirements and that the differences between boxes represent significant investments in C3. In Figure 2 these differences represent systems with vastly different performance or survivability characteristics. In Section 3 (on levels of capability) some of these differences include buying larger numbers of systems.

Figure 2 represents the top three tiers of a hierarchical structure that may extend up to a dozen levels. Figures 3 and 4 indicate the lower levels for the Fire Support/Interdiction missions in the European theater. This mission area is highly illustrative of the concepts in this section. The criterion in developing a structure at this level is that it describes all of the distinctions needed to articulate what C3 systems can and cannot do.

Figure 3 describes mission levels in that all of the statements pertain to what forces can do while Figure 4 focuses on C3 functions and systems. In Figure 3, conflict level establishes whether systems are under conventional or nuclear stress. The size and intensity of the conflict is implied by specifying the theater and corresponds to threats and war plans developed for that theater. Operations describes own force response with nuclear response involving C3 for custody, control, request, release, and execution. Target distance is a key determinant of requirement for sensor coverage. The three ranges shown are nominal values representing approximate range of ground and small aircraft based sensors, sensors for a high altitude, stand off platform like a TR-1, and possible futuristic sensor concepts, respectively. Target classification, as the term is used here, establishes how rapidly information needs to move from sensor to firing unit. Fixed time sensitive targets refer to ground forces or support units temporarily located at staging areas or choke points. Weapons class determines the tradeoff between the capability of the weapon to locate a target and the need for support from C3 external to the weapon. Thus, all of the factors in the hierarchical tiers of Figure 2 help to provide links between C3 capabilities and mission effectiveness. Figure 4 contains the bottom two tiers of the hierarchy which are C3 functions and elements.

3. LEVELS OF CAPABILITY

Levels of capability are a mechanism for articulating investment choices. If C3 improvements are contemplated it is because there is a gap between current capability and the capability called for by defense policy and war plans. The idea of levels of capability is to establish reasonable incremental steps between these two boundaries. An investment choice is reasonable if it consists of a package of C3 improvements designed to provide one of these incremental steps.

Figures 5, 6, and 7 present levels of capability corresponding to mission; theater, and overall defense capability in Figure 1. The mission is Fire Support/Interdiction that was described in Section 2 above and the theater is Central Europe. The levels shown are notional and not the ones used in the effort but they are reasonable and with some additional detail could have been used. The mission levels in Figure 5 correspond to conventional war, conventional operations, with all weapon classes.

There are two broad observations about these levels that should apply to levels of capability in any MOA. First, each level represents a sensible strategy for employing military forces. That is, given that fiscal constraints do not enable us to buy everything, each of these levels results in achieving a significant military capability compared to lower levels. The second point is that having achieved any given level, there is a significant dollar investment in C3 systems needed to get to the next higher incremental level. Thus, the levels articulate hard choices.

Note that the levels do not explicitly distinguish all combinations of factors in the hierarchy. Further, some levels of capability are combined in moving up the hierarchy from mission to theater and overall defense levels. Partly this is to keep the presentation of results manageable. Nothing is lost by this simplification since an initial analysis can determine a "space of interest" and then a closer look can focus on that space. The "space of interest" is the level of capability where the cost of needed systems and fiscal constraints intersect. That is where high level decision makers need to choose among missions and theaters and where the results of this effort can translate those decisions into specific system level choices.

4. RESULTS

The methods described in this paper were used in a pilot study. The aggregate 15 year cost of levels of overall defense capability similar to those in Figure 7 were over laid against two fiscal constraints. The two constraints represented modest real growth and zero real growth in spending for C3 over the next 15 years. The detailed results of this study are classified. However, in summary, the results illustrated very graphically a number of key choices between degrees of nuclear modernization and degrees of conventional modernization. The initial results have been briefed at the highest levels of OSD and OJCS and are currently being expanded to develop alternative fiscally constrained long term plans. In addition, these results are being used to explain to non-technical communities the impact of possibly severe fiscal constraints on the C3 program.

Figure 1
Mission Oriented Analysis

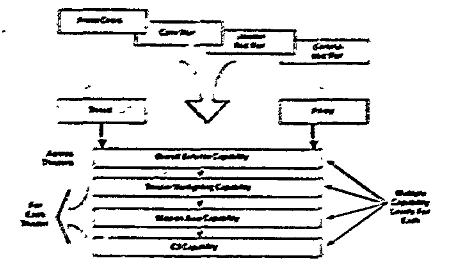


Figure 2
C2 Assessment Structure

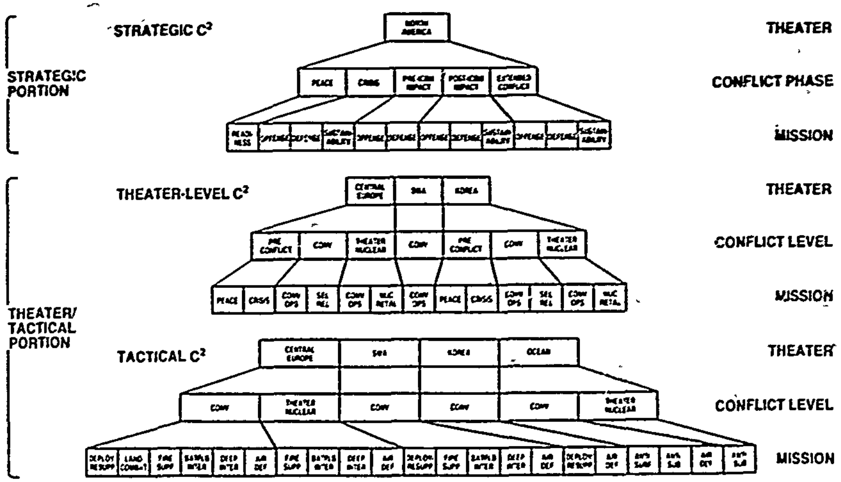


Figure 3
Mission Dimensions: Fire Support Interdiction

- Conflict level
  - Conventional war
  - Theater nuclear war
- Operations
  - Conventional
  - Nuclear
- Target distance
  - 0-30 km
  - 30-150 km
  - Beyond 150 km
- Target classification
  - Fixed
    - Non time sensitive
    - Time sensitive
  - Moving
- Weapons class
  - Artillery
  - Missiles (SSM)
  - Aircraft

Figure 4
C3 Capabilities Dimensions Fire Support Interdiction

- C3 functions
  - Target acquisition
  - Target analysis and planning
  - Target attack
  - Damage assessment
- C3 elements
  - Command centers
  - ADP
  - Communications
  - Sensors
  - C3CM
  - Navigation
  - Procedures

Figure 5
Mission Levels: Fire Support/Interdiction

V. Interdiction beyond 150 km

IV. Moving targets to 150 km

III. Time sensitive targets to 150 km

II. Fixed targets to 150 km

I. Fixed and moving targets to 30 km

Figure 6
Theater Levels: Central Europe

IV. Moving target interdiction

III. Fixed target interdiction

II. Responsive maneuver and reinforcement

I. Static defense from fixed positions

Figure 7
Levels of Overall Defense Capability

Level	Capability
VI	Conduct enduring flexible strategic nuclear operations.
V	Fight global conventional war.
	Conduct aggressive maritime strategy versus Soviet Navy.
IV	Conduct limited enduring strategic nuclear operations.
	Employ NSNF selectively in response.
III	Execute flexible SIOP.
	Execute modernized air land battle concept in Europe.
II	Execute preplanned TPFDL and hold defensive positions in NATO Warsaw Pact conventional conflict.
	Maintain choke point and regional open ocean control versus Soviet Navy.
I	Execute preplanned SIOP.
	Conduct power projection for crisis or low intensity conflict.

EVALUATING THE IMPACT OF STRATEGIC C3 SYSTEM PERFORMANCE

R. W. Grayson

The MITRE Corporation
7525 Colshire Dr.
McLean, Virginia 22102-3481

ABSTRACT

The primary purpose for evaluating the impact of any element or aspect of U.S. forces is to help establish priorities for programs and budgets. For strategic command control and communications (C3) analysts, this can be a particularly frustrating process. The performance of the strategic war machine is dominated by weapon hardware. Consequently, the contribution of C3 systems to strategic force effectiveness has historically been overlooked or taken for granted in the broad defense community. Those who have struggled in the programming and budgeting processes understand the axiom that if your program can't "make things go boom", you don't have much leverage in those arenas.

This paper presents a methodology for expressing the impact of strategic C3 performance in terms of war fighting capabilities. It begins with traditional strategic force evaluation techniques and develops specific measures of C3 performance that can be integrated directly into those techniques. It allows the analyst to portray C3 impacts in terms of mission accomplishment and does so in a manner that can be widely understood. It allows direct comparison among C3 systems and among C3 and force related systems.

ANALYTICAL OBJECTIVES

To devise an approach to evaluating any components of the national defense structure, it is necessary to first establish the ultimate objective of that effort. What are we trying to accomplish? It is all too easy, otherwise, to get caught up in the details of simulation and synthesis and find that our product is not really useful. I believe that the ultimate objective of any funded activity in the defense community should be to establish or maintain the capability to wage war whenever and however it is in the national interest to do so. I know this puts me in conflict with some deterrence advocates; but without the capability to wage war successfully, I doubt the likelihood of deterrence. It follows, therefore, that I believe the ultimate objective of evaluations, including C3, is to help prioritize the use of our limited defense budget toward the ultimate capability for war fighting in the national interest. We must prioritize development and procurement programs through which we can establish the necessary capabilities. And, we must prioritize training and exercise programs through which we can assure maximum effectiveness of existing resources.

Given these overall analytical objectives, it is necessary for C3 evaluations to support two categories of comparisons:

- o C3 systems, subsystems, and components in the context of overall C3 capabilities, and
- o C3 systems in the context of overall strategic force capabilities.

The first of these two types of comparisons is to allow prioritization of potentially diverse C3 elements contributing to the accomplishment of the same or diverse C3 missions. Sensor processing versus command center display systems versus communications systems is a fair example. The second of these two comparisons is to allow prioritization of diverse elements contributing to the same or diverse strategic missions. Sensors versus C3 systems versus weapon delivery systems is a common example.

Accomplishment of these comparisons and prioritizations in a traceable, auditable way implies the need for common, mission-related measures of effectiveness. It follows, also, that the measures used must be understandable and useful to a range of people including war planners, C3 planners, engineers, budgeteers, programmers, and targeteers.

STRATEGIC EVALUATION CONTEXT

It is prudent to recognize that the evaluation of strategic mission success (i.e., projected outcomes for the various levels of potential nuclear conflict) offers a unique context for the analyst. This is evidenced primarily in two ways. First, in contrast to tactical assessments, the evaluation of strategic effectiveness is dominated by weapon considerations. Within the bounds of the scenarios normally considered, the contributions of the engineering of the delivery systems and the physics of the weapons' effects tend to dwarf the contributions of C3 systems. Similarly, evaluations of the total costs involved tend to be dominated by the costs of the weapons and weapon delivery systems. Second, in contrast to evaluations within the Strategic Defense Initiative (SDI) program for example, there is a long-standing precedence in the way potential strategic mission effectiveness is calculated and priorities derived.

Because the impact and cost of C3 system performance were recognized, or more likely believed, to be so relatively small, strategic analyses were conducted for many years with virtually no consideration given to them. There was a common acceptance of a nebulous "force multiplier" effect attributable to C3. However, the tendency was to assume perfect C3 as a baseline thus converting any less-than-perfect C3 performance, which is likely, into a "force demultiplier", as it is sometimes called.

C3 analysts attempting seriously to prioritize their systems in the context of overall strategic force capabilities have found themselves in the role of relative newcomers to the strategic force arena. They have been forced to accept and integrate with an analytical process of long standing. Although new and different methodologies for common mission-related measures of strategic effectiveness have been advanced in the C3 community, there has been little success in convincing the strategic force community to change. The pragmatic problem, therefore, is to devise a way to integrate or assimilate measures of C3 effectiveness into the established measures of strategic effectiveness.

INTEGRATION OF MEASURES

Strategic force evaluations are based almost exclusively on expected levels of damage potentially achievable against enemy war fighting, war supporting, and C3 capabilities and occasionally against selected urban and industrial centers. Although sometimes computed with great specificity, the outcome is more frequently expressed in aggregated terms like those implied in Equations #1 and #2 which are shown below.

$$D.E._j = 1-[1-(SSPK_{ij})]^{N_{ij}}$$

Equation #1

Where $D.E._j$ = Cumulative probability that a designated level of damage is achieved on designated targets of type j.

$SSPK_{ij}$ = Single shot probability that a weapon of type i will achieve the designated level of damage against a target of type j.

N_{ij} = Number of weapons of type i allocated against each target of type j.

$$SSPK_{ij} = (PPLS_i)(PLG_i)(PP_i)(PD_{ij})$$

Equation #2

Where $PPLS_i$ = Probability of pre-launch survival for weapons of type i.

PLG_i = Probability that weapon delivery systems of type i will launch and fly successfully (does not account for weapon accuracy, only delivery system reliability).

PP_i = Probability that weapons of type i will penetrate the appropriate defenses.

PD_{ij} = Probability that a single detonation of type i delivered with its associated accuracy can achieve the designated level of damage against a target of type j.

I prefer to call these the "general strategic equations". In a simplistic fashion, they serially link the uncertainties associated with the delivery of nuclear weapons on target and the uncertainties of the weapon effects themselves. The users of these equations group the uncertainties into larger or smaller aggregations consistent with the needs of each particular analysis. Penetration through several layers of defense might each be entered separately, for example, instead of the single probability (PP) shown in Equation #2 The grouping shown, however, has proved most useful to me, particularly because it correlates with the sequence in which the uncertainties would impact chronologically along the weapon flight path, and is thus intuitively clear. In this sense, it may be convenient to view the equation as representing a series of "barriers" through which the weapons must pass. Extending this analogy in 1980, The MITRE Corporation and the Defense Communications Agency added a parameter representing another barrier; namely the uncertain likelihood that the forces themselves would receive the order to execute. This is called the "Probability of Correct Message Receipt (PCMR)" and its inclusion is shown below in Equation #3.

$$SSPK_{ij} = (PCMR_i)(PPLS_i)(PLG_i)(PP_i)(PD_{ij})$$

Equation #3

Where $PCMR_i$ = Probability that the order to execute is received by weapons of type i exactly as sent.

This parameter, PCMR, now serves as the principal measure of strategic communications connectivity throughout the defense community.

The linkage of the PCMR measure to strategic force effectiveness is easily recognized and understood and is increasingly being accepted. It would be convenient if the other contributions of C3 systems could be linked in a like manner; by adding the probabilities of correct or timely decisions, for example. This can be done to some people's satisfaction. However, the circumstances must be rigidly controlled to make those probabilities meaningful and the coupling of C3 performance to them becomes quite tenuous. I believe the solution lies in the reexamination of the general strategic equations (including, of course, PCHR).

Each of the parameters of uncertainty in Equation #3 (PCHR, PPLS, PLG, PP, PD) must be calculated separately for each set of analytical circumstances of interest (weapon type, threat, scenario, etc.). This outcome is most credible when those parameters are calculated within the constraints of reality. For example:

- Only whole numbers of weapons assigned to each target; i.e., N constrained to be an integer number.
- Range constraints honored in assigning weapons to targets.
- Footprint constraints honored in assigning missile reentry vehicles to targets.
- Only alert weapons, or weapons that can be brought on line in a timely manner, considered for assignment.

The obvious simplifying assumptions implicit in these equations are 1) that the cumulative probabilities associated with individual weapons are independent of each other, and 2) that the parameters of uncertainty also are independent of each other. The impact of the former is generally lost in the aggregation. The impact of the latter can be minimized by selectively computing a range of values for various conditions.

In calculating the parameters of uncertainty, the analyst has the straight forward mechanism for relating small changes in system or subsystem performance directly to the aggregated force effectiveness. This is true, of course, provided that he can express that coupling. C3 performance cannot be coupled to the parameter PLG. It can be related to PD only if the location of the target (arguably a C3 function or a C3I function) is incorporated. However, it can be related to PPLS and PP. The value of those two parameters vary according to the time in the conflict at which the strategic response is executed; generally, both decreasing as the response is delayed. C3 systems support the processes of decision and execution and dictate the time at which a decision can be made intelligently and executed efficiently. The timeliness of C3 functions (with associated quality levels) is thus an indirect coupling, but nonetheless one that should be acceptable and understandable to those who must establish program and budget priorities.

The character of C3 support to strategic force operations, of course, is not simply a matter of the timeliness of decisions. It is also a matter of the quality of those decisions. The C3 goal is an informed, intelligent decision, made in a timely manner, consistent with the achievement of national objectives in whatever conflict scenarios emerge.

Based on these considerations, I believe the analytical construct shown in Figure 1 allows complete coupling of C3 to strategic force effectiveness.

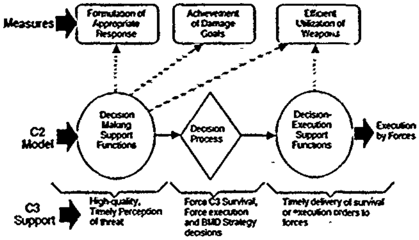


Figure 1. Analytical Construct

This construct shows a command and control model with C3 decision-making support (before the fact) and decision-execution support (after the fact). The measures of effectiveness shown are discussed separately below. They are all mission oriented and are expressed in or related to the terms of the general strategic equations.

<u>Appropriateness of response selection</u> is a comparison of the target set selected based on the actual perception of the threat (i.e., based on C3 performance) and the target set that same decision maker would have selected given perfect knowledge of the threat. Note that this is not a measure of the quality of the decision, nor is it a measure of whether or not we will win the war. People do not even know how to define "winning" in a nuclear war. It is rather a measure of the C3 system's ability to give the decision maker the best possible basis for a decision. The "appropriate" decision, thus, is the one most closely matched to the planned achievement of national objectives under the circumstances at hand. Figure 2 characterizes the implications of inappropriate decisions.

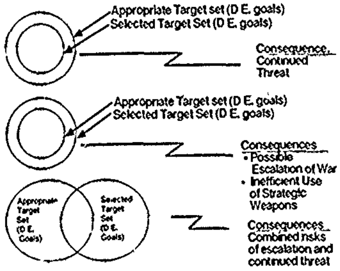


Figure 2. Inappropriate Decisions

Achievement of damage goals is a comparison of the damage expectancy associated with the actual execution time and that associated with the execution time that would have accrued with perfect knowledge. As implied earlier, every strategic response has an execution window after which the response becomes increasingly difficult to achieve. The primary causes are the loss of communications connectivity to the forces (i.e., lower PCMR from direct attack and counter measures), the loss of weapons (i.e., lower PPLS from direct attack), and the loss of defense suppression (i.e., lower PP from loss of defense suppression weapons or delays in receipt of the order by those forces).

Efficient utilization of weapons is a comparison of the number of weapons required when execution time is based on the actual perception of the threat and the number required if execution were based on perfect knowledge. Use of this measure implies one of two US capabilities to assign weapons to targets. In the first, war plans might be prepared in advance and stored with the weapons based on two or three values for SSPK reflecting increasingly higher levels of damage incurred by the US C3 structure and forces. To accommodate the lower values of SSPK, greater numbers of weapons would necessarily be assigned to each target. The decision maker would then choose the appropriate preplanned damage level. In the second, adaptive planning mechanisms could be developed to allow real time targeting/retargeting at the time the decision is made.

Achievement of damage goals and efficient use of weapons are simply alternative applications of the equations; one application letting D.E. vary, the other letting N vary. The following is a very simple example of the alternative uses.

Objective: 0.9 D.E. on each of 20 defended targets

Weapons Assigned: ICBMs

Coupling: A. SSPK = .6 based on .9 survival and 1.0 penetration, given EAM release $\leq T_1$

B. SSPK = .27 based on .9 survival and .45 penetration, given EAM release $> T_1$

Impact measured by achievement of D.E. goals:

Case #1: EAM released by T_1

$$D.E. = 1-[1-SSPK]^N$$
$$.9=1-[1-.6]^N \text{ and}$$
$$N=3 \text{ per target}$$

Case #2: EAM released after T_1

$$D.E. = 1-[1-.27]^3$$
$$= 1-[.39]$$
$$= .61$$

Impact measured by efficient utilization of weapons:

Case #1: EAM released by T_1

$$.9=1-[1-.6]^N$$
N=3 per target (as above)
60 RVs required

Case #2: EAM released after T_1

$$.9=1-[1-.27]^N$$
$$N=8$$
160 RVs required

The range of coupling of C3 performance to strategic force decision and execution processes is pictured in Figure 3. Couplings are shown by the dashed lines, real-time functional interfaces, all of which can be expressed in terms of the quality and timeliness of information.

APPLICATION OF THE METHODOLOGY

The application of this methodology is pictured in Figure 4.

In the manner already described, the analyst would compare whatever alternatives are available (the right hand side of the figure) against the reference standard, namely that which would accrue with perfect knowledge. To do this, the analyst needs the following tools and capabilities.

1. Subsystem performance models: There must be detailed models or algorithm with which to calculate the parameters of uncertainty for Equation #3, SSPK. While these are not commonly in use in the offices performing C3 analyses, they are not hard to find. The Air Force Center for Strategic Analyses (AF/SA), for example, should be able to provide a relatively complete set.

2. Wargame model: A two-sided, two-phase or three-phase model is needed. That is, one in which one side attacks, the second side retaliates and, possibly, the first side strikes again. The model supporting the SIOP-RISOP games (i.e., the US Single Integrated Operations Plan versus the hypothetical Red Integrated Strategic Operations Plan) is the most credible example. Models from the National Test Bed (NTB) being developed for the SDI program would seem to be another credible example, although the specific capabilities of the NTB have not yet been established. Use of either of these models or facilities would likely be difficult and costly for evaluating most C3 enhancement concepts. Fortunately, however, representative SIOP-RISOP type gaming can be emulated fairly easily.

3. Decision rules and/or decision makers: The methodology requires that a decision be made. The decision need not be "optimum", but must be made in as professional a manner a possible. The best way to do this is to build a set of "if-then" decision "rules" to serve as guides for response

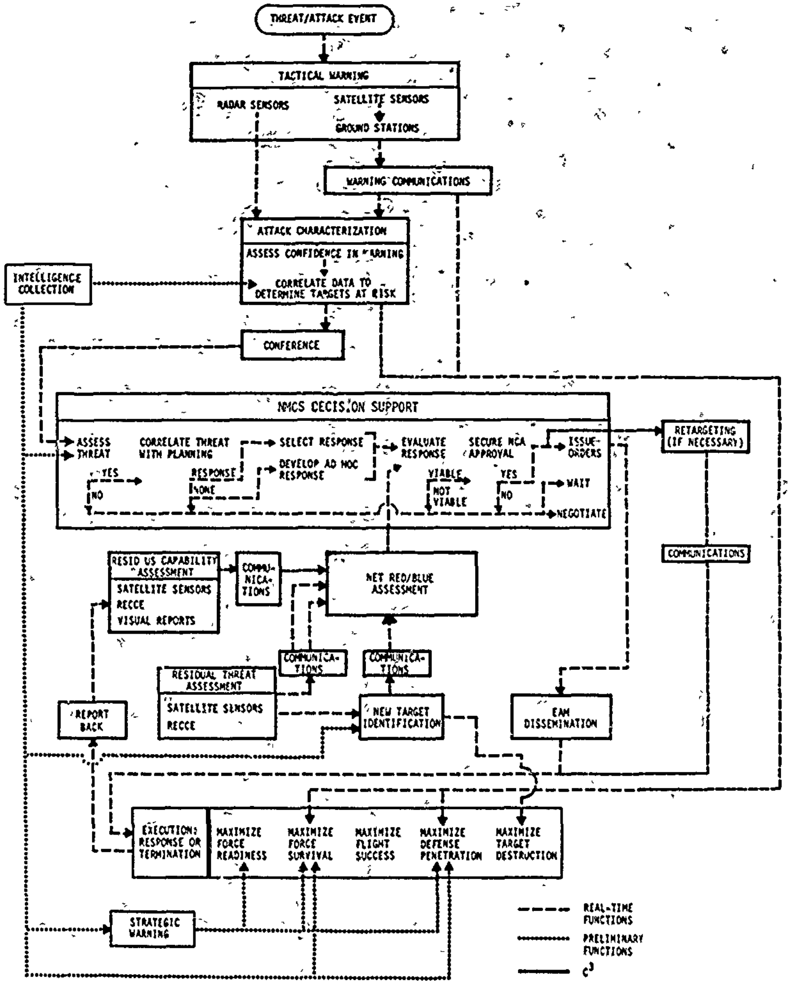


Figure 3. Schematic Coupling of Measures

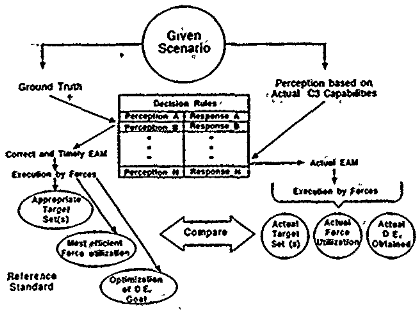


Figure 4. Application of the Methodology

selection. These rules advise the decision maker that for each perception of the situation and threat, a given response is most appropriate, based on preplanning and current guidance. As more decision making role players exercise the model and critique the rules, the more realistic they would become. As this happens, the results from using the methodology would obviously become more credible.

4. Test driver: Finally, a test driver model is needed to synthesize the flow of information into the decision process. This can be as simple as a script or as complex as a computer generating outputs from individual sensors or data fusion facilities.

The modeling implied in Figure 4 may be very aggregated, using representative but stylized threats, weapon sets, or target bases. It can be event-driven, clock-driven, on-line or using look-up tables. However, for credibility, the analyst must make a conscious effort to be realistic in the kinds of decisions that would be made.

SUMMARY

The methodology described in this paper is much less sophisticated than most operations research professionals would prefer. It is purely pragmatic, based on hard experience (and some success) in making the needs of the C3 community heard and considered in the DOD programming and budgeting arenas.

The analytical processes associated with this methodology are cumbersome. They require literally that the analysts synthisize the outcome of an entire war. However, that is where the ultimate measure of effectiveness lies; namely in the projected potential capability to wage war in the national interest. And, I believe that adequate war game models exist and can be emulated.

The computational processes associated with this methodology are potentially time consuming. However, one impact of aggregation is to mitigate this fact.

The "bottom line", in summary, is that the measures generated using this methodology directly relate to the business of national defense and compare directly to the measures commonly used throughout the defense community. The conclusions as to C3 program or budget priorities may not always be accepted. However, the measures used should be understood and acceptable to programmers, war planners, engineers, operators, budgeteers, and targeteers as well as C3 planners.

When the old school general says, "How does C3 make things to boom?", you can say, "I'll show you."

THE AUTHOR

Mr. Grayson is one of only two or three people with hands on experience as director of major DoD studies in the structure, deployment, and employment of 1) strategic offensive forces, 2) strategic defense systems, and 3) strategic C3 capabilities. In the late sixties and early seventies, at Air Force Studies and Analysis, Bob led the SABER LANCE series of annual strategic force posture studies. In that period, he also led three assessments of BDM concepts. One of these, the "Missile Offense-Defense" system, was subsequently incorporated for a time in the DoD Five Year Defense Program. In the late seventies, at MITRE, Bob led the analysis for the final three minimum Essential Emergency Communications Network (MEECN) Master Plans and the first two MEECN Communications Plans. In both cases, he was principal author of the resulting plans.

Mr. Grayson has worked on Ballistic Missile Defense throughout the eighties and currently leads the Terrestrial C2 Working Group of the SDI Phase One Engineering Team.

Planning for Integrated System Evaluation:
An Application to S.D.I.

Dr. Michael F. O'Connor

Decision Science Consortium, Inc.

The conceptual development and subsequent demonstration and validation of a major system is admittedly a very complex, dynamic process and a theoretically sound procedural plan will usually meet with numerous implementation difficulties. This presentation discusses a theoretically-based approach to the organized integration of high-level value trade-offs to system design decisions. The work presents a top-down approach to the assessment of total system worth which provides for the integration of measures of different aspects of system worth. Measures can be performance projections from highly detailed simulations or expert assessments of potential performance.

The application of the approach to the actual evaluation of proposed SDI battle management/C^3 architectures is with emphasis on the planning for the total evaluation and the practical problems involved in the implementation of the plan. One of the issues addressed is the relation of measures of subsystem performance to yield a valid assessment of overall performance.

This presentation discusses planning for integrated system evaluation and is based on both theoretical work and empirical work done on the evaluation of Battle Management/Command, Control, and Communications (BM/C^3) architectures that are being designed for the United States Army Strategic Defense Command (USASDC) for the Strategic Defense Initiative (S.D.I.). While the application pertains to S.D.I., the concepts involved are relevant to any program evaluation that involves projection of the future performance of a proposed major system. The work is prescriptive and methodological in nature, discussing what should be done rather than presenting, for example, a piece of software designed to solve part of the problem. Nonetheless, implementation of the ideas discussed here do not involve the development of major new algorithms or associated software. In fact, much of what is discussed has been termed "good engineering design" by one researcher who was presented with the work. He indicated that these ideas are routinely accomplished in good engineering design implementations. The claim here is that the approach expands the meaning of "good design" and also that the more restricted view of good design that the researcher implied was routinely implemented is, in fact, not routinely implemented for any of a number of reasons depending on the program and the designers. These reasons include increased cost of rigor, schedule slippage, programmatic constraints, the usual organizational approach to system acquisition, or in some cases simple laxity.

The approach here termed a top-down approach to integrated system evaluation necessarily involves a linkage of user, procurer, and designer in a trade-off analysis that is broader in scope than the analysis usually accomplished, and the proposed analysis is to be done early in the program with whatever tools are available. Difficulties in accomplishing the approach are not technical, but rather organizational, and even though parts of this presentation deal with the use of complex performance simulations and related technical problems, the view adopted here for system acquisition decisions is one of an organizational decision maker. Put theoretically, the utility function relevant to the design decisions is a social one involving numerous organizations. Put practically, the information required to accomplish design trade-off analysis resides within several organizations and, thus, requires a systematic in- [illegible] tion of tr [illegible] red [illegible] erall eval[illegible] lyses necessarily involve more than subsystem performance trade-offs often addressed in complex simulations.

This presentation will involve three phases: the theory of the approach, the methodology involved, and an example of procedures to be followed in implementation.

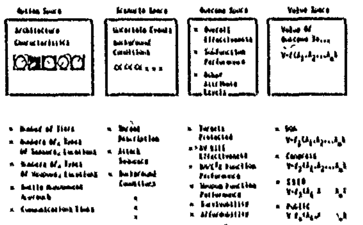
1.0 A Theoretical Framework for Trade-off Analysis

Any decision where the options are clearly defined involves uncertain future events and value trade-offs with respect to alternative consequences. The system design problem is further complicated by the fact that the decisions do not involve fixed options, but rather involve the design, development, testing, and deployment of systems that must meet defined performance goals in uncertain future scenarios and must also satisfy numerous societal criteria such as low life-cycle cost and political acceptability name two that are relev ... active

The traditional decision analytic representation of a decision tree in which the relevant outcomes of a decision are specified as branches, each of which is a sequence of acts and events terminating in a set of consequences uniquely associated with the particular branch of the tree. An alternative representation of the decision problem appears in Figure 1 in which four abstract spaces are represented. This particular representation has some characteristics that are particularly suitable for discussing the system acquisition problem. (This approach is based on work presented in O'Connor and Edwards, 1977 and O'Connor, 1985). Briefly, Option Space contains all possible designs of the kind in question. For this S.D.I. problem, it is the space of all architectures and associated characteristics. It can be thought of as a vector space; but however it is viewed, it contains all the characteristics relevant to *specification* of an architecture. This is the space of the engineer or designer who specifies an architecture by selecting a set of levels of characteristics. In Figure 1, the specification of an architecture is simplistically displayed as settings on several dials. It is acknowledged that in Option Space, design dials cannot, in general, be manipulated independently. A change in one often changes many others. Needless to say, designs can be highly specific or very general, and a set of specific characteristics can be mapped into some more general system characteristics. Design options can, thus, be represented in hierarchies. The main point is that all design decisions and changes occur as changes in Option Space.

FIGURE 1

Four Abstract Spaces



Once a system architecture such as a BM/C^3 architecture for S.D.I. is specified, it can be evaluated by considering all futures in which the system will operate. These futures are characterized in Figure 1 as scenarios in Scenario Space. This space, as with Option Space, can also be viewed in several ways: as a vector space, as an event tree whose branches comprise the set of all future scenarios, as a sample space, or as some combination of these. A scenario is viewed as consisting of two parts: the stage setting into which the system is introduced and the subsequent unfolding of acts and events that details the future with the system deployed. More than one scenario is possible and, in fact, all potential futures for a system comprise an infinite set. Some care must be taken to develop an efficient, manageable set of scenarios that will allow for valid system evaluation and that will also be cost efficient. Such a set of scenarios should validly represent the future in which the system will be deployed while at the same time discriminate among alternative system designs in terms of value differences. These two requirements will most likely be in conflict, the first yielding broad, general scenarios while the second requires fairly specific scenarios.

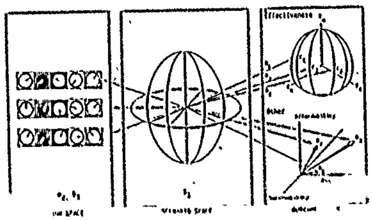
The value differences referenced are represented as levels of attributes in Outcome Space, which is the third space depicted in Figure 1. As with the previous two spaces, Outcome Space can be visualized as a vector space, and attributes in Outcome Space can be types of performance, different "ilities" such as survivability, or different attributes such as "world political stability". Outcome Space contains all attributes relevant to specification of the outcome of an Option Space, Scenario Space combination, and one can think of these attributes as structured in a hierarchy in which descending levels contain subattributes of increasing specificity.

The fourth space in Figure 1, Value Space illustrates an important fact, which is that the same outcomes have different values to different populations. For the S.D.I. example in Figure 1, different agencies such as DoD, Congress, SDIO, the general public, and the Soviets have different values for the same set of outcomes, and value trade-offs among the attributes of Outcome Space may well be different for these different populations. Thus, the same point in Outcome Space can map into a large number of points in Value Space.

Another important point with respect to system design that is often neglected by those who emphasize performance simulation as the major aspect of system evaluation, illustrated in Figure 2, is that the same levels of overall system performance can be achieved by different systems that have very different values on other attributes of Outcome Space such as affordability, feasibility, and survivability. While this is no revelation to system evaluators, what to do about this has usually been overlooked. These attribute trade-offs are not after the fact evaluation issues. Rather, they are decision issues that are relevant to the design problem and should be examined and resolved before highly detailed system design and testing take place. To do so is difficult work and part of this, of course, involves a good mission analysis (so often neglected), but another part is the policy analysis that addresses those complex value trade-offs characterized by Outcome Space/ Value Space combinations. Which is the best way to achieve a given desired level of performance? One way would be to deploy a large number of lower performance platforms that represent a feasible, near-term technical solution. A second way would be to go for a more risky, less feasible but higher performance technical solution that will lead to fewer required platforms and will achieve the same performance levels. Costs will differ, including development costs, deployment costs, and replacement costs to ... e ...

FIGURE 2

Mapping: Option Space to Outcome Space



What about political aspects? How much of this should be examined up front? For S.D.I., much of the answer to this question lies with the policy analysts whose domain involves the political intricacies of developing a system where different options also represent different political positions. One thing seems clear, however, and that is the desirability of answering so many of these value trade-off questions early in the design process. A decision must be made with respect to those attributes relevant to evaluation of this design problem. The problem of developing measurement scales for these attributes must also be addressed, and that problem will not be discussed here save to state that such scales, if they are to have much value, will probably be specific to the evaluation ... Measures of Effectiveness (MOEs)", while having a nice ring to it, sounds like having generic measures of beauty.)

The message of this theoretical introduction is simple, that value trade-offs relevant to system development should be addressed up front in the design process and in an organized, methodical fashion that yields design direction. In order to do so, some evaluation tools are required. With S.D.I. as with most other major system developments, performance simulations will be necessary, and numerous simulation models exist, each of which answers some part of the performance simulation problem, and none of which does the complete job. The more complicated the simulation, the more expensive it is to run, change, and understand. But the system evaluator need not shrink in fear of these seemingly ominous considerations. Rather, he should plan to do what he will eventually be forced to do anyway, and that is to use what he has when he has to make his decision. How to do this will be discussed later when the concept of an attribute bank is introduced.

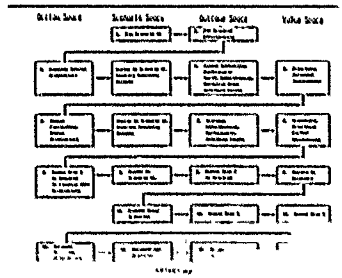
2.0 The Methodology

2.1 Evolving Architectures

Figure 3 illustrates the use of the approach to evolve architectures. As indicated in the figure, the process involves first specifying a relevant scenario (Step 1), specifying the required effectiveness in the scenario (Step 2), and identifying an initial architecture (Step 3) that will yield the desired effectiveness. That architecture is then "deployed" in the scenario and the remaining scenario details (part of Step 4) of the scenario are developed. This can be done using a computerized simulation to estimate performance, or by using a judgmental approach such as step-through simulation (see Ulvila, J. and Brown, R.V., 1978). The assessment of system performance and other attribute levels is accomplished in Outcome Space. Part of this involves verification that the proposed system actually yields the desired performance. The detail to which performance can be modeled at this point is an issue of paramount importance. The goal is not to obtain two-digit or better performance estimate accuracies, but rather to examine value trade-offs in detail. Such trade-offs will involve intersystem performance trade-offs as well as trade-offs between performance and other attributes such as cost, political stability, etc.

FIGURE 3

Using the Approach to Evolve Architectures



By examining attribute levels associated with the proposed system including cost, "ilities", political, social, and other outcomes, potential improvements are identified and prioritized (Steps 4 and 5 in Figure 3). The architecture is then reworked with the designer and improvements (with respect to attributes) are sought (Steps 6 through 8). Note that this can be an exploratory policy analysis at this point. Analyses can be judgmental. Performance estimates can be expert assessments because architecture differences yielding different attribute levels with respect to major political and social issues will be major differences, and detailed performance assessments may not be required even if sufficient detail exists to make such estimates.

Alternatively, such analyses may require detailed performance simulation. For an SDI example, consider an architecture that yields acceptable leakage protection in a scenario, but is designed such that its surveillance system and weapons are deployed at fairly low levels thus having increased vulnerability to anti-satellite weapons over systems in higher orbit. Suppose further that the performance simulation does not address the dynamics of satellite attack. That is, vulnerability is not addressed in the simulation. Then several different architectures can be designed, each yielding different leakage performance, and each having different vulnerabilities. The two attributes are assessed separately and then addressed in value trade-offs. Which is better, increased performance with increased vulnerability or vice versa? Depending on simulation complexity, such off-line analyses will be required.

It is easy to see that this early trade-off analysis, involving an approach such as Multiattribute Utility Assessment (MAUA--see O'Connor, 1985) can be of different levels of generality and can also be an exploratory, iterative analysis that involves "excursions in Outcome Space". For example, the implications for architecture design of stressing early technical feasibility, survivability, political acceptability, low life-cycle cost, graceful degradation, etc. can be explored. The crucial point here is the link between Outcome Space-Value Space trade-off analysis and design implications.

The level of precision here need not be that of a simulation that estimates, for example, leakage in terms of the number of ree[illegible] v[illegible]icles (RVs) striking the Continental U[illegible]ed [illegible]es (CONUS) (for an S.D.I. example) [illegible]

meaningful measurement scales for attributes should be used, and they can be defined as relative to the attribute levels of the architectures under consideration. As more precision and detail are required, more specific measurement devices may be required. This will be discussed later.

Once the architecture trade-offs have been examined for this first scenario, the same process must be repeated for the other relevant scenarios (Step 10). This is most relevant to performance estimates, although changes in scenario can affect levels of other attributes also. An example is survivability if it is not part of the performance assessment be discussed in the example above. Another example would be political acceptability.

Note that any architecture changes made which would yield attribute levels in subsequent scenarios analyzed will potentially affect attribute levels in earlier scenarios, and checks must be made to ensure that this has not occurred. If it has, it must be noted, and either a compromise design must be achieved, or a decision must be made to go with one or the other of the designs in question. Thus, interscenario performance trade-offs are yet another aspect of the organized trade-off analysis.

Most of the iteration between scenarios and the impacts of design changes across scenarios as discussed will involve somewhat specific details relevant to performance assessments. Other broader attribute trade-offs relevant to such issues as policy analysis will often require fairly broad scenario differences. For example, a different scenario might be relevant to discriminations among designs with respect to each of several attributes. This issue is too extensive to discuss here. For a detailed discussion see O'Connor and Edwards, 1977.

2.2 The Use of an Attribute Bank

The issue of quantification of attributes in trade-off analysis and the related issue of measures of effectiveness were introduced in Section 2.1. This section will expand on the discussion. The issue of validity and reliability of subjective value assessments is one with which practitioners of decision analyses are familiar. Decision analyses involving one time evaluation of decision options usually [illegible]ify attributes using relative intra-[illegible]ibute value scales with ranges determined [illegible]

who seek to develop more global measures of "effectiveness", value, utility, or the like to be applied more than once to different sets of options must find intra-attribute value scales whose ranges include all potential options that may be considered. This is a problem faced by those who develop measures of effectiveness (MOEs). This can be a difficult problem and, thus, arises the issue of relative versus absolute measurement scales, or in measurement theory, the related issue of the uniqueness problem. It is fairly simple to determine an appropriate measurement scale for the height of potential basketball team candidates. It is quite another to define a measurement scale for the attribute known as Battle Management/Command, Control, and Communications (BM/C^3) system effectiveness. In fact, even defining the attribute is difficult let alone defining an appropriate MOE.

The focus of this paper is not the establishment of measures of effectiveness for BM/C^3 systems, although that was a required part of the actual evaluation. The focus is, however on the use of trade-off analysis in the de[illegible] process, and such analysis necessarily involves quantification of attributes. An earlier recommendation made in this paper was that the evaluator faced with the problem of implementing a system design evaluation should use the tools at his disposal and get on with the task. Often too much concern is placed on the so called "apples and oranges" problem, and large amounts of time and/or money are spent in the development of detailed performance simulations that will have all the properties deemed necessary to address the relevant trade-off issues involved. In the application used as an example for this discussion, numerous simulations of parts of S.D.I. system performance were available, but none directly represented the specific BM/C^3 to system (weapons and sensors) trade-off issues in the BM/C^2 architecture evaluation. Because time and resources were not available, there was no choice but to proceed using the tools (and the contractors) available at the time. The attribute bank was developed as a means of organizing that evaluation, and, at the same time, attacking the so called "apples and oranges" problem.

Figure 4 lists the characteristics of the attribute bank approach used in the development of methods for assessing the worth of BM/C^3 architectures being designed in studies for the ...DC. The bank consists of a hierarchy o... osed evaluation attributes which procee... sively more specific measures at each hierarchy level. Similarly, the hierarchy contains simulated measures to the left and expert assessments to the right. Thus, the uppermost left-hand corner of the hierarchy will have aggregated, simulated measures and the lower right-hand part of the hierarchy has expert judgments with respect to highly specific, disaggregated attributes. Although this discussion is most amenable to performance assessments, it is also relevant to the broader trade-off analyses required for the design process.

FIGURE 4

Attribute Bank Approach

- Hierarchical Bank Of Proposed Evaluation Measures
- Detailed Decomposition Of Overall Architecture Performance To Highly Specific Subsystem And Component Performances
- Left to Right . . . Simulated "M.O.E.'s" To "Expert" Assessments

Figures 5 and 6 illustrate some of the hierarchy for evaluating the worth of BM/C^3 architectures. Figure 5 represents the highest level whereas Figure 6 decomposes the node numbered 112132, BM/C^2 system effectiveness For each node in the hierarchy, a definition was developed with an accompanying discussion of measurement details. This detail is contained in an evaluation handbook (see O'Co...or, 198)

FIGURE 5

Overall BMD... Decomposition

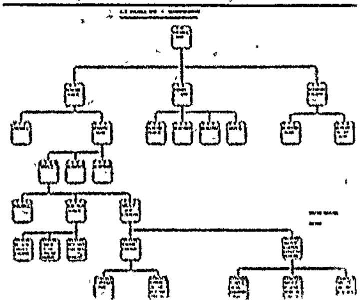
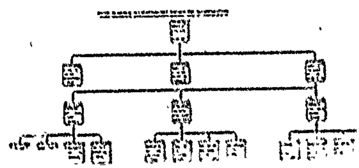


FIGURE 6

BM/C... Resource Allocation/Management Capability Decom...



For many of the attributes, performance simulations were available, and the outputs were characterized alternatively as measures of system effectiveness, measures of effectiveness, measures of performance, or simply expert assessments. Examples of simulation outputs appear in Figure 7. For other attributes, analysis or expert assessments may be required. The point of the hierarchy is to integrate all aspects of the effectiveness evaluation into an organized framework and to focus trade-off analyses.

In evaluating performance, an overall measure is desirable such as that for node 1121, overall BM/C^3 performance. However, any simulation capable of assessing such a complicated, highly aggregated performance will necessarily be highly detailed and potentially difficult understand. The evaluator must understand the simulation used or have a translator, a go between who can translate the model outputs into language the evaluator can follow. Theoretically, if the evaluation is driving the simulation and not vice versa (which is often sadly the case) the outputs will answer evaluation questions. The point of the trade-off analysis based on the attribute bank structure is to assure this.

FIGURE 7

Performance Modelling

Analytical/Simulation Model Types	Modelling Examples	Output Measure Type	Output Measure Examples
Strategic Defense System (SDS)	• End-to-end System Representation	MOSE	• Value of Defended Targets Surviving • System Leakage
Specific Grouping of System Elements*	• SDS Midcourse Tier • Specific BMS & Subordinate LMs, Weapons, and Sensors	MOE	• Tier Leakage • Threat Capacity (for Specified Grouping)
System Element	• BM Component of SDS	MOE	• BM Computer Processing Delays (System Element Level) • Discrimination Capacity
Function/Small Functional Grouping	• Discrimination • Comm Link Propagation	MOP	• Discrimination Data Processing Delays • Critical Message Delays
Phenomenology	• BMA • Conventional Fragmentation Warhead Effects	Inputs To Higher Level Models	• Atmospheric Perturbation

*e.g., elements affecting a specific tier, fusion of other defense values.

Any simulation, however good, will omit certain aspects of system worth, and the attribute bank will contain nodes indicating that aspect of worth and a potential measure. Familiarity with the particular tools used for evaluation will allow mapping to the nodes of the bank that are covered and will pinpoint those attributes not covered. They can be separately assessed through analysis or judgment and the results can be combined with simulation outputs to yield a complete evaluation. Often decision-analytic trade-off analysis will be required to accomplish the integration of such measures, but this is an implementation procedural issue as opposed to a theoretical problem.

An example of this problem, that was discussed ealier, is when a performance simulation is used to estimate system performance using leakage. However, survivability aspects of certain components are not included in the simulation. The simulation can be methodically run for several different architectures, the survivability of which can be separately assessed. Two attributes can then be defined; leakage performance without attack by the enemy and survivability. The trade-offs between the two attributes can be assessed and architectures located on each. Note that the issues of independence in Option Space and value independence in Value Space must be carefully addressed in such analyses.

Such an organized, thorough evaluation is not easy, even if planned a priori using trade-off analysis as is recommended here. Information such as that appearing in Figure 8 on evaluation issues covered must be organized in the attribute bank so that the evaluators know what issues are covered, to what degree, and by what instrument. The evaluators then have control over the evaluation issues and can use the expertise available to them in a continuation of the organized trade-off analysis as discussed earlier.

The evaluator picks the issues that are relevant to the evaluation at hand using the attribute bank. Examples appear in Figure 8. The mapping of available performance models to the attribute bank allow for a choice of attributes to be assessed using either performance models, analyses, or expert assessments. Where a model lacks coverage of a specific aspect of system design and related performance, assessments are obtained with respect to that attribute and are combined either judgmentally using a form of trade-off analysis or by an integrated application of available models. The latter process is further described in Section 3.0.

FIGURE 8

Evaluation Issue Areas

Issue	Level of Coverage	Models
BM/C^3 processor/comm tradeoffs • processor speed and location • communications throughput	• Time delays explicitly accounted for • Representation of SDS architecture at system element level • Performance related comm. characteristics for comm. link types	TDS, SIMSTAR
Man in the loop issues	• Impacts of time delays associated with manned functions	TDS
Survivability of SDS elements	• Temporary or permanent damage due to hostile activities	TDS
Subsystem performance capability tradeoffs • sensor discrimination • BFM requirements	• Impacts of specific system element capabilities on engagement timelines, battlespace	TDS, DEOS, COVED, SAS library

3.0 Procedures Used in Applying the Methodology to the S.D.I. BM/C^2

Evaluation Problem: An Example

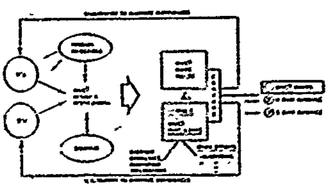
3.1 The Procedures

In this section, the difficult problem of assessing the worth of different BM/C^3 structures will be addressed indicating how the trade-off analysis should proceed using the tools that are available. In Figure 9, the BM/C^3 to systems iterative integration process is illustrated. A problem with S.D.I. BM/C^3 evaluation, and with C^3 evaluation in general relates to the "force multiplier" problem. "C^3 does not kill people, bullets do!" This argument is old and need not be discussed here save to indicate that most simulations emphasize weapon and sensor characteristics and configurations while minimizing representation of BM/C³ characteristics, partly because of the difficulty of including all of these details in one simulation. The system dynamics representation for a problem like S.D.I. is already huge for simply the weapon and sensor performance modeling without considering the many complicated details of BM/C³.

Thus, it is usually the case in BM/C³ design, that weapon and sensor parameters are constraints on the BM/C³ system and not vice versa. Figure 9 represents an alternative approach in which reasonable bounds on the BM/C³ system serve as constraints which limit the range of allowable weapon and sensor parameters, and, thus, the BM/C³ becomes the "driver" of the design optimization. Given the great uncertainty with respect to BM/C³ feasibility, this is, indeed, appropriate.

FIGURE 9

BM/C3 To Systems Iterative Integration Process



In Figure 10, the guidelines developed for BM/C³ by the well-known Eastport Study Group are presented. This group was concerned with BM/C³ feasibility, in particular software and algorithm feasibility. As indicated, that group argued that BM/C³ should be an integral system component of the design and not simply an applique added after thorough consideration of weapons and sensors. The evaluation problem then becomes one of ascertaining the impact of these recommendations, if adopted, on performance and associated system. While it was possible to develop some off-line assessment of BM/C³ performance impact and integrate this with other measures using trade-off analysis as discussed in Section 2.0, it was believed that a more integrated assessment of weapon-sensor/BM/C³ performance trade-offs was required. The question to be answered is displayed in Figure 11. Given definition of the axis labeled as BM/C³ simplicity, which was the correct curve? The dotted line represents required performance in the scenario, and it is necessary to decide where on the BM/C³ continuum to locate to assure that requirements are met.

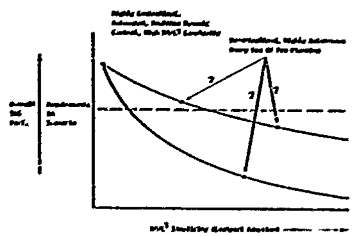
FIGURE 10

Eastport Study Group BM/C³ Guidelines For SDI

- BM/C³ Must Be Integral To Overall Architecture, Not An "Applique"
- Principal Influence On BM/C³ Design Must Be Reducing Software Complexity
- Most Flexible BM/C³ Architecture Is Hierarchic
  - Decentralized C³
  - "Battle Groups" With Local Autonomy
- BM/C³ System Should Be Open And Distributed
  - Loose Coordination
  - [illegible]
- BM/C³ Components Should Be Simple And Redundant, Even If Suboptimal
- BM/C³ Testing Should Be Realistic
  - Infer Full-Scale Performance Through Test Of Small Segment
  - Use Simulation Extensively
  - Develop An "In-Line" Testing Capability
- BM/C³ Architects Should Consider Establishing A Separate, Dedicated Communications Network

FIGURE 11

Performance/BM/C3 Tradeoffs



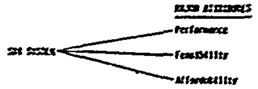
While numerous simulations of total system performance were available for the evaluation, none of those available provided detailed representation of overall S.D.I. system performance (e.g., R.V. leakage into CONUS) as a function of weapon details, sensor details, and similar levels of detail with respect to BM/C³ characteristics such as organizational structure, communications throughputs, BM/C³ processor throughputs, etc. Those that were large, detailed, multitiered models generally provided detailed dynamic representation of threats, weapons, and sensors, but little or no representation of BM/C³ characteristics. (For example, the version of D.I.D.S.I.M., Defense in Depth Simulation, available assumed instantaneous communications between nodes and a centralized BM/C³ management structure. Later versions were under development to change these simulation characteristics.) How could the evaluation be accomplished with the tools at hand?

The next few charts display the recommended procedure for the evaluation problem described

Assume that a simulation of overall performance is available that accurately portrays weapon and sensor characteristics and resultant performance impacts. (Numerous detailed simulations, e.g., DIDSIM, were available at the time.) Figure 12 illustrates the first step in the procedure. The system is optimized in terms of numbers, locations, and capabilities of weapon and sensor parameters in detailed performance trade-off studies. Impacts of major weapon and sensor characteristics are identified and quantified (e.g., midcourse tier leakage). At this point, BM/C^3 characteristics have not been introduced. BM/C^3 essentially is assumed to be a "perfect" system, i.e., no BM/C^3-induced performance decrements are assumed.

FIGURE 12

Evaluation Procedure



- Optimize System For Leakage/National Objectives
  - Simulation/Modelling - E.G., DIDSIM
  - Assume BM/C^3 "Perfect"
- Major System Trades Identified
  - Drivers E.G.
    - Late Midcourse Discrimination (Near Term)
    - Space Based Asset Survivability (Far Term)

Next use the attribute bank to develop a set of BM/C^3 related attributes to be varied in the evaluation. Identify major structure and capability impacts and a method to characterize performance impacts. Examples of such BM/C^3 attributes appear in Figure 13. Using these details, develop the BM/C^3 structure that yields appropriate attribute levels and overlay it on the simulation by indicating which functions will be performed where, which information will be available to which platforms, and what threats can be engaged as a result. This step is detailed in Figure 14. An example with respect to such attributes as performance, survivability, and feasibility is a highly centralized BM/C^3 architecture versus a completely autonomous one and the BM/C^3 management algorithms required for each. For the centralized structure, the system can utilize weapons and sensors to engage threats in an optimal overall fashion. For the autonomous architecture, each weapon must engage the threats in the order it encounters them until it runs out of "bullets".

FIGURE 13

Architecture Trades

[illegible]
- Processing Efficiency
- Flexibility, Graceful Degradation
- [illegible]
- [illegible]
- [illegible]
- [illegible]

[illegible]
- Performance Efficiency
- System Feasibility (Software, Algorithms)
- Performance Speed
- [illegible]
- Survivability
- Flexibility
- [illegible]

[illegible]
- [illegible]
- [illegible]
- Performance Quality (Plan Accuracy)
- Situation Assessment Requirements
- Graceful Degradation, Survivability

FIGURE 14

Evaluation Procedure (Continued)

OVERLAY BM/C^3 STRUCTURE

- Functions To Platforms
- Information Flows
- Concept Of Operations

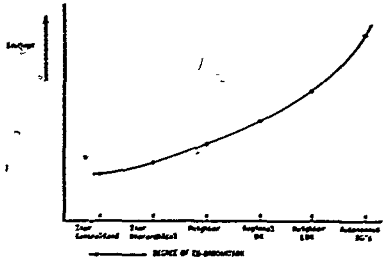
Plus Threat Scenarios

- Run Simulation To Obtain Architecture Structure Induced Leakage
- Still Zero Time Delays For Comm. & BM/C^3 Processors
- Leakage Differences Due To BM/C^3 Concept Of Operations
- Required System Timelines For Specific BM/C^3 Construct

At this point only the BM/C^3 management structure is modeled in terms of information available and decisions made. No delays due to communications or processors are included. Given the analysis thus far, the impact of increased simplicity should be decreased S.D.I. system performance as illustrated in Figure 15 in which simplicity is represented as degree of Battle Manager Coordination (BM – Battle Manager; LBM – Local Battle Manager; BG – Battle Group).

FIGURE 15

Leakage Increase Due To BM Structure



However, there are costs to centralization that are not represented at this point. Thus far, the BM/C^3 network has been assumed to cause no delays, and weapons and sensors receive information as required to operate within their respective cycles. However, the more centralized architecture will involve detailed communications and processing delays that will not be incurred in the autonomous battle groups. Thus, an estimate of such delays must be included. Another issue here is key node vulnerability and the potential for catastrophic failure of the system through system fault or enemy destruction. One of the Eastport recommendations is for a separate, dedicated communications network. A separate communications simulation can be run to include battle managers and communications nodes and it can be used to estimate communications and processing delays due to the BM/C^3 network. (Such simulations were available at the time.) Although this is not the same as a complete representation of the entire network, it must suffice given the lack of the "complete" simulation. Given such a modeling effort, communications and BM/C^2 processor delays can be estimated and inserted into the simulations as delays in sensor and weapon performance. This concept is illustrated in Figure 16. The exact method for doing this depends, of course, on the simulation. The resultant system performance is then estimated and the outcome might be similar to the illustration in Figure 17.

FIGURE 16

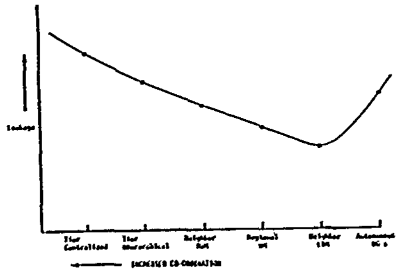
Evaluation Procedure (Concluded)

- Introduce Comm And BM/C^3 Processor Delays
- Feasible Comm, BM/C^3 Processor Capabilities → BM/C^3 Delays → Leakage
- Required BM/C^3 Timelines Met → BM/C^3 Requirements
  - BM/C^3 Processors
  - Comm Throughput

OUTPUTS:
- Preferred BM/C^3 Concepts Given Feasible Capabilities
- Allowable Comm And ADP Delays

FIGURE 17

Leakage Assessed For Different BM Structures and Associated Command Processing Delays



3.2 Summary

The approach described here allows the estimation of required BM/C^3 characteristics given weapon and sensor parameters and it also allows estimates of system performance given "reasonable" BM/C^3 parameter values derived from models of the BM/C^3 network. In the implementation discussed here, it is recommended that the attribute bank serve as a guide to the procedures adopted to ensure complete and systematic coverage of all BM/C^3 characteristics.

The approach described here provided for evaluation of the impact of BM/C^3 architectures on the same measure of performance as that used in modeling weapon and sensor system performance. This provides a capability for modeling BM/C^3 versus sensor or weapon trade-offs which is often perceived to be a more "valid" analysis. It also allowed for completion of the evaluation within the time, funding, and organizational constraints existing at the time.

In a particular evaluation, it may be quite feasible to develop completely new simulations of performance that include all system detail relevant to architecture performance trade-off considerations. That would not change the desirability of using the attribute bank approach to integrate the evaluation and to focus an a priori trade-off analysis to identify the architecture characteristics for which key performance or effectiveness simulations are to be developed. This structured approach will give the program manager, who in the end must make the crucial decisions, control of the evaluation from beginning to end, and he will not have to rely on tools or reports that do not directly focus on the evaluation issues.

References

O'Connor, M. F. *Establishment of standards and procedures for evaluating and comparing SDI BM/C^3I architectures* (Working Paper WP-85W00070). McLean, VA: The MITRE Corporation, April 1985.

O'Connor, M. F. *Evaluation handbook for the USASDC Composite Architecture Synthesis Team (CAST)*. McLean, VA: The MITRE Corporation, July 1986.

O'Connor, M. F. and Edwards, W. *On using scenarios in the evaluation of complex alternatives* (Technical Report DDI/DT/TR-76-17). McLean, VA: Decisions and Designs, Inc., 1976.

Ulvila, J. W., and Brown, R. V. Step-through simulation. *Omega: The International Journal of Management Science*, *6*(1), 25-31, 1978.

ANALYSIS OF BM/C³I SYSTEM PERFORMANCE

by
Dr. Thomas E. Moriarty
United Technologies Systems Analysis
McLean, Virginia

1. INTRODUCTION

This paper addresses the analysis of the performance of BM/C³I systems within the overall context of large scale military operations. The second section develops the natural hierarchy of systems within military operations and the points of view of the various decision makers that work with elements of the hierarchy. The corresponding hierarchy of models is then presented along with a discussion of the limiting factors in the models at various levels. Section 3 discusses the criticality of BM/C³I issues in determining the effectiveness of systems in a large-scale military operation. In Section 4, various measures of effectiveness are compared, with emphasis on finding the proper context for judging the comparative military utility of alternative systems. The fifth section of this paper compares the various analysis methods including simulations, simulators, wargames, and exercises; and it shows how they all fit together to develop a comprehensive analysis of BM/C³I system utility. Section 6 presents a top-down modeling method, called the Combat Information Flow Model, which has proven useful in analyzing BM/C³I systems in various disciplines. The final section contains a summary of the salient features of the paper and draws several important conclusions.

2. SYSTEM AND MODELING HIERARCHIES

Military systems can be conceptualized within a natural hierarchy that is arranged according to increasing scope or entities included. As shown in Figure 1, the lowest level of the system hierarchy consists of Components & Assemblies - devices which are not functional when they are taken apart. Components & Assemblies combine to form Subsystems, which in turn combine to form Systems. In general, Systems are characterized by the capability to do damage to the enemy or aid in achieving military objectives. Systems become Weapon Systems when properly combined, usually with human beings included to provide direction and/or decision making. Weapon Systems are brought together to form Fighting Units, groups of weapon systems and people, which require increased human coordination in order to function properly. As Fighting Units are brought together, they must coordinate their efforts to prosecute a conflict over a significant geographic area, and these combined units are usually termed Sub-Theater Forces. Finally, when dealing with the entire scope of a military conflict, the system is termed a theater, and the entities involved in the activity are usually called Theater Forces. At the Theater Force level, considerable emphasis must be placed on such issues as doctrine, strategy and the management of the battle, all of which are implemented via the Command, Control and Communication systems. It should also be noted that the time window of interest increases by orders of magnitude as one goes upward in the hierarchy. Components, Assemblies and Subsystems often complete their functions in seconds, whereas Sub-Theater and Theater Force performance is often assessed over periods of days, weeks or even months.

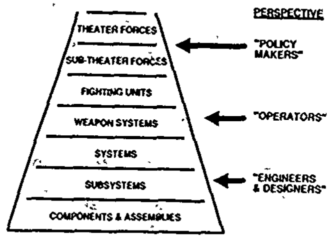


FIGURE 1 SYSTEM HIERARCHY

An important consideration in the hierarchy of military systems is the perspective taken by decision makers at various levels. Referring again to Figure 1, the engineering and design decisions tend to focus on subsystem and system hardware issues. The operators, usually the military users of the weapon systems, tend to emphasize the performance of the weapon systems in the field, as they interact with the other elements of the fighting units. The upper levels of the hierarchy are of particular interest to the senior military and civilians that set the policy and continuously make difficult force structure decisions in the face of budget limitations. The increased involvement of BM/C³I phenomena in systems that are high in the hierarchy requires that high-level policy makers be concerned about BM/C³I capabilities in their decision making processes.

The modeling hierarchy in military systems analysis (See Figure 2) is similar to, and actually driven by, the hierarchy of military systems. The models in the hierarchy are developed to facilitate the understanding of the elements and their interactions within the various levels of military systems.

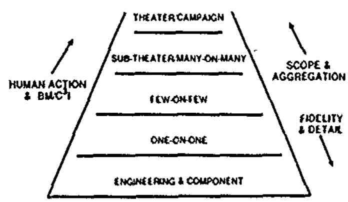


FIGURE 2 MODELING HIERARCHY

At the lowest levels of the model hierarchy, a modern computer will be fully engaged in calculating the flow field over a section of an airfoil, or the radiation pattern of a particular antenna. At the higher levels, the same computer will simulate all the entities in thirty days of conflict in Central NATO or an all out nuclear exchange. Clearly, as the scope of the model increases, going up in the modeling hierarchy, data and descriptions within the models must become more aggregated. Analysts working with the hierarchy are constantly challenged by the task of facilitating the flow of data between the models at different levels.

There is an unfortunate dichotomy in the modeling hierarchy that arises from the tendency to reduce fidelity and detail at the upper levels. Human action and BM/C^{3}I phenomena are increasingly important at the higher level systems and the higher levels of the hierarchy as well. Unfortunately, there is often too little fidelity in the high level models to be able to capture the functional BM/C^{3}I activities, but at the lower levels these activities don't occur in the same way if they occur at all. The consequence of this dichotomy is that

BM/C^{3}I systems and phenomena are frequently assumed to be perfect, or they are ignored which amounts to the same thing. Weapon systems evaluated in this unreasonably idealistic environment are then often characterized as being more effective than they really can be when actually put into the field.

3. BM/C^{3}I AND THE TOP-DOWN PERSPECTIVE

Analysts of weapon systems and warfare, generally agree that in large scale operations, modern military combat is dominated by the flow of information and messages among the participants. Battle Management - Command, Control, Communication and Intelligence are critical functions with strong leverage on battle outcome, and a fundamental part of battle management is decision making. Contributing functions are intelligence gathering, fusion, dissemination, surveillance and tracking, target correlation, prioritization and assignment. Also included are such concepts as strategies, deployments, allocation and apportionment, situational awareness and mutual support. The common denominator in all these functions and concepts is the flow of information. The policy makers must be able to capture the capabilities of the BM/C^{3}I systems in order to assess the capabilities of the force structure. Thus the methods of evaluating BM/C^{3}I system performance must be able to relate to the outcome of the conflict, not just the ability to pass information.

4. MEASURING SYSTEM UTILITY

The yardsticks used to measure system performance have a number of names such as Measure of Performance (MOP), Measure of Merit (MOM), or Measure of Effectiveness (MOE).

When BM/C^{3}I systems are evaluated, the same terminology is used and yet there is a trap that is frequently fallen into. The trap consists of the fact that the ability to pass a lot of information in a timely manner is not necessarily good. Effective systems must pass the right information at the right time so that correct decisions are made and the military forces are successful in their missions. Costs must be considered, because nothing is less useful than a system that isn't affordable. Thus, the term Cost Effectiveness Analysis (CEA) or, better yet, Cost and Operational Effectiveness Analysis (COEA) capture more of the needed philosophy.

The key to meaningful analyses of BM/C^{3}I and BM/C^{3}I-oriented systems is to address the military utility of these systems in their proper context. The process begins by looking at the broad military objectives associated with the use of the systems. The policy makers are usually looking for the minimum cost system configuration that meets these military objectives. In order to judge the degree of accomplishment of these objectives, the system must be judged in the context of the entire scope of the conflict at hand. Ultimately, the measure of military utility must come down to the issue of "does it help win the war?"

There are major differences in the realizations of military utility for different defense functions. Strategic forces, for example, require dissemination of specific quanta of information in a precise and rapid manner during very early stages of the conflict. The effectiveness of strategic forces hinge largely on BM/C^{3}I functions that occur within minutes of the start of the conflict. Air, sea and space forces have vastly different media thru which communications must flow, and the timing requirements differ as well. Many of the command and control functions are preset, and forces are often trained to respond in very well defined ways.

Tactical forces tend to function for a much longer period of time, and thus the effectiveness of tactical BM/C^{3}I systems is often judged over periods of days or weeks. Air, sea and land forces must be coordinated and the changing requirements for interoperability pose a special challenge. In the tactical environment, the management of the forces must react to changes in strategy on both sides of the conflict over a long period of time, and the command and control functions will tend to change considerably over the period of the conflict.

Intelligence dissemination and fusion are integral elements of the BM/C^{3}I systems, and are powerful contributors to the force effectiveness in both strategic and tactical environments. Here, major differences in effectiveness measures occur between strategic and tactical applications, but the more notable difference occurs in comparing intelligence functions in peacetime and wartime. Peacetime functions strive for completeness and accuracy so as to drive preparation and readiness for contingencies. Wartime intelligence functions have the added requirement for currency and timeliness which changes the manner in which their effectiveness is measured.

5. TOOLS OF ANALYSIS

The analysis of BM/C^{3}I systems effectiveness requires a number of tools to be used in combination to obtain the true measure of a system's utility. Three major types of tools should be used in conjunction with analysis of BM/C^{3}I systems: test data - the results of testing the hardware involved; exercises - where the systems are exercised in a semi-realistic environment, but the performance is often clouded by the conditions of the exercise and the fact that there is no repeatability and therefore little statistical reliability, and models - to be discussed here in more detail. Models, in general, are representations of reality and can be divided into a number of classes simulations, simulators, wargames, contingency analyses, and cost-estimating relationships.

Simulations are computer programs that can capture a large portion of the application context, and can be used to generate statistical replicates of the performance of the systems. Frequently unmanned, occasionally simulations will contain humans in their proper roles in a large scale evaluation effort. In trying to control the number of variables that simulations have to contend with, decisions on both sides of the conflict are often preprogrammed, which can result in erroneous conclusions on the part of the analytical community.

Simulators are usually manned facilities that carefully replicate the environment that the human will experience when using the system in question. Useful for training and determining human behavior on a local scale, they often have limited scope and generally the human experiences a preprogrammed environment.

Wargames are generally large-scale representations of the higher-level decision making process associated with a particular manifestation of conflict. They usually include two-sided variations and are very useful as training devices. Wargames rarely result in multiple samples of the same situation owing to the length of each play period.

Although not usually thought of as models, contingency analyses are representations of realities that could occur and as such they are models. These contingency analyses develop the credible scenarios and plans that in turn provide the context for evaluation of BM/C^3I systems.

Finally, cost-estimating relationships (CERs) are also models, and are particularly important because they provide the life cycle and/or budget costs that are necessary in developing cost-effectiveness measures. The most effective system has no military utility at all if it is not affordable. The reality of military system development is that budget limitations are always in effect and the analyst must respect these limitations in quantifying military utility.

It is only by putting all of the above tools together in an integrated and coherent analysis that one can perform a comprehensive evaluation of the cost effectiveness of a BM/C^3I system. This is the way to meaningful evaluations of the military utility of systems, and analysts must strive to achieve the complete scope necessary for these investigations.

6. THE COMBAT INFORMATION FLOW MODEL, A TOP-DOWN MODELING METHOD

The Combat Information Flow Model (CIFM) is a structured simulation methodology that is particularly well suited to studying BM/C^3I and BM/C^3I-oriented systems. It simulates systems in their environment by focusing on the way in which information is received, processed & transmitted by each element. (The interested reader will find a detailed description of CIFM in the appendix to this paper, thus the modeling method will be only briefly summarized here.) In CIFM, the BM/C^3I elements are modeled explicitly. Other elements are transformed into an information flow format in order to provide a common structure.

CIFM models the system elements by nodes that are connected by links. Messages (in the most general sense) pass over the links to provide the flow of information in the model. The node structure in the model mimics the real world layout of the system. An important feature of the CIFM methodology is the capability of the model to include the combatants, weapons and the interaction of combat units. The structure of CIFM and the dynamic memory of SIMSCRIPT II.5 allow system elements to be created by messages during execution, providing considerable modeling flexibility. The structure of CIFM requires the receipt of a message by a node before an action can take place. Similarly, an action is required for a node to send a message. The result is that the BM/C^3I concepts cannot be short changed or by-passed. Especially important is the fact that the CIFM methodology is designed to develop models that encompass the full scope of the combat objectives associated with the systems being analyzed.

CIFM experience includes the Theater Missile Defense Architecture Study - the design and analysis of the BM/C^3I portion of a tactical missile defense system in NATO; a trade study among the parameters of the Space-Based Kinetic Kill Vehicle concept in SDI, development of a ship's combat system model; and several avionics/weapon system integration tasks. At the time of this writing, CIFM is being configured to analyze: the BM/C^3I architecture of Anti-Submarine Warfare (ASW), the high-level BM/C^3I functions in a joint command (US Forces Caribbean); and the BM/C^3I requirements of Anti-Satellite (ASAT) systems.

7. SUMMARY

The analysis of BM/C^3I systems must always consider the systems hierarchy, the modeling hierarchy that goes along with it, and most important, the policy maker's need to consider the top-level concepts of force effectiveness and total system cost. In order to properly support these decision makers, BM/C^3I systems analysts must take the top-down approach in structuring and executing their evaluations.

Developing and evaluating a valid measure of military utility is a difficult and at the same time critical issue. In the case of BM/C^3I systems, these tasks are particularly difficult because the systems never operate without the presence of a number of weapon systems and human beings, so that the scope of interest is always large, and the activities are very complex. These difficulties notwithstanding, however, the analysts must consider system costs and the military objective which precipitates the use of the system. In most cases, a single model or test is not sufficient to gain the necessary insight, and a number of tools such as simulations, simulators, wargames, contingency analyses, cost-estimating relationships, tests and exercises must be coherently integrated in order to obtain meaningful system utility analyses.

In the specific area of simulations, there are methods available (for example CIFM) which are focused on BM/C^3I issues. They can provide the necessary top-down, structured approach and capture the full scope of the conflict required for credible evaluations of the military utility of BM/C^3I systems.

Finally, the analysts must always keep in mind that the bottom lines to the analyses of the military utility of BM/C^3I and BM/C^3I-oriented systems have to be:

"Is it affordable?"

and

"How does it help win the war?"

APPENDIX

COMBAT INFORMATION FLOW MODEL (CIFM) METHODOLOGY

1. INTRODUCTION

This paper provides a perspective on modeling and simulation appropriate to the needs of individuals or organizations concerned about the performance of systems that involve Battle Management or Command, Control, Communication and Intelligence (BM/C^3I) functions. It discusses the usage of models and simulations in modern military system design and evaluation and presents some of the considerations in choosing tools to support these activities. Finally, the Combat Information Flow Model (CIFM) is presented along with some of the features that will be of particular value to a BM/C^3I system analysis or design effort.

2. THE ROLES OF MODELS AND SIMULATIONS IN ANALYSIS AND ENGINEERING

A major obstacle in developing combat systems, particularly C^3I and C^3I-oriented systems, is the inability to accurately test or even predict their effectiveness before actual hardware is built and tested. This deficiency compounds the difficulty of advocating new systems in austere budget environments and hinders the process of evaluating system trade-offs, developing operational concepts, and assessing performance against postulated threats. Although models and simulations are now in widespread use as a means of solving this problem, many have severe limitations. Some, for example, tend to be complex and inflexible in their ability to handle varying aspects of system performance. Frequently, a combination of "few-on-few" and a "many-on-many" models must be used to evaluate a system's performance against multiple diverse threats. Finally, most models are unable to handle the critical C^3 elements of system performance. In response to these modeling deficiencies, a simulation methodology has been developed that is specifically designed to overcome these major problems associated with many current models. This methodology, called the Combat Information Flow Model, can be applied to the problem of developing C^3I systems and system concepts, assessing their effectiveness in threat scenarios, and performing trade-off analyses for the elements of the system.

Modern military systems are becoming increasingly complex and expensive as they have to contend with threats that are increasing in capability and numbers. Mechanical and electrical designs have to be validated before a significant expenditure in fabrication is made and software has become so expensive to develop that the algorithms must be thoroughly verified before they are put to code. Engineering design relies heavily on models to provide insight into the function of complex systems. As the systems become larger and more complex, models provide the only way to capture the many phenomena that the systems will deal with in the real world. Thus the models must be chosen, designed and/or developed as carefully as one develops the system itself, for they will have a direct bearing on the eventual capabilities of the operational system.

The Combat Information Flow Model (CIFM) can be the primary method for assessing operational utility and system effectiveness against various threat scenarios. CIFM is an innovative approach to simulating large scale systems where overall system performance is strongly driven by interaction among the system elements. At the higher levels of conflict modeling, communications surveillance, situational awareness and other forms of information transfer have great influence on the Task Group Commander's decisions as well as weapon system effectiveness and thus affect battle time lines and levels of intensity. These considerations often have considerably more leverage than the technical details of the particular weapon systems since they provide the opportunity to use, or misuse, the ship's weapon systems. CIFM provides a means of accounting for these important aspects of combat and effectively evaluating their influence on battle outcomes.

3. SIMULATION TYPES AND CONSIDERATIONS

3.1. Models vs Simulations

A model is a representation of reality in the most general sense. There is a wide spectrum of model types, ranging from mental models of a relationship, through closed form algebraic equations that relate one set of phenomena to another, on up to massive computer programs that encompass thousands of cause-and-effect relationships and their interactions. A particular and important subset of the general class of models is the simulation, a model in which the phenomena of interest interact and are monitored as time progresses. The Combat Information Flow Model produces simulations that allow the user to observe the time behavior of the system of interest.

3.2. Deterministic and Stochastic Simulations

Deterministic simulations have no random occurrence in their execution processes. Each run will be identical to the previous one, assuming that none of the input parameters or data have been changed. Since real world phenomena usually include various random processes, deterministic simulations must characterize the random processes by single valued functions that are based upon expectations. These simulations are frequently called "Expected Value" models. It should be noted, however, that this kind of simulation is not used all that often for modeling combat, and one of the reasons is that it does not portray uncertainty very well.

A more common type of simulation is the stochastic simulation which actually contains random variables and random processes in the execution stream. In this type of simulation, the random draws will result in two subsequent runs being different because of the probabilistic nature of decisions or occurrence within the run. Stochastic simulations, typically called Monte Carlo simulations, require a number of replications of the same situation in order to achieve a level of statistical confidence.

Stochastic simulations can be further broken down into two classes (1) those which characterize phenomena by means of an expected value or probability of occurrence, and then model the phenomena by comparing the results of a random draw against a threshold for occurrence, and (2) those which model phenomena by their distributions and then draw from those distributions. The former (occasionally they are also called "Expected Value" models) are fairly good at capturing overall average system performance. The latter models are more complex, but they capture excursions from the norm and the effects of nonlinearities and non-gaussian distributions that are not available any other way.

3.3. Modeling Trade-offs

Every modeling or simulation task is unique in that the underlying issue or analysis will pose certain requirements and constraints on the approach taken. The higher the level of the analysis or issue, the larger the scope of the simulation required. However, as the number of phenomena is increased, the level of detail or fidelity in the simulation must go down in order to be compatible with modern computing capacity - both size and speed. This, in turn, requires aggregated data and descriptions of the systems and phenomena being simulated. The best modeling approach for problems with large scope is one that takes a top-down perspective and includes all phenomena of interest at the minimal level of detail that will still yield the required insight into the system performance. Additional detail should be added only when the analysis shows that (1) an unmodeled parameter is causing a particular behavior and (2) the optimization or modification of that parameter is part of the analysis. As will be discussed in the next section, the Combat Information Flow Model lends itself very well to highly aggregated descriptions of system elements and natural phenomena, and it also results in a highly modular model which accommodates increases in detail when necessary. This helps keep the development times, run times and memory requirements at the lowest levels that still satisfy the program's analytical goals.

Another important consideration is the time required to configure and use a model. Computer speed and memory size are constantly increasing as the technology evolves in the direction of more efficient and cheaper computation, and the result of this evolution is that the cost of the people involved in these tasks becomes the overriding consideration. In today's environment it makes sense to capitalize on bigger, cheaper computers, if this results in less time spent in the construction of the models, and more time spent in the analysis of the modeling results. This is the approach taken when using the Combat Information Flow Model to develop a new model, where the emphasis is on quick configuration of the model and ease of understanding and manipulations of the simulation output.

4. THE COMBAT INFORMATION FLOW MODEL NODE/LINK/MESSAGE STRUCTURE

The Combat Information Flow Model (CIFM) is particularly well suited to the task of modeling BM/C^3I oriented systems. Originally developed by and in use at United Technologies Advanced Systems Division in San Diego, CA, CIFM is being used by United Technologies Systems Analysis, in McLean, VA, and others in UTC on a number of projects. Thus a large and growing community of modelers and analysts will have a common understanding of the model. CIFM has the following inherent advantages over alternative simulation approaches:

1. A major portion of the code (called the CIFM Framework) for each application is written, tested and debugged, and does not have to be retested for each new modeling task. Its features (input and output formats, user friendliness, models of physical phenomena, etc.) are common to each model application. Because of the large amount of code on hand, already written and tested, models can be constructed quickly.

2. Modules written for one model can be transferred directly to other models. Because there is a CIFM testbed model, and because CIFM is being used on a number of other projects, there are many modules already developed that can be used either as they are or with minor modifications.

3. A person's knowledge of one simulation will apply to other simulations, resulting in a significant reduction in the time required (learning curve) to begin writing code for a given model.

In combination, these advantages provide a modeling environment which is oriented toward reducing the model development time associated with an analysis requirement, and providing a tool that the analyst will find is both robust and easy to use. The emphasis is on solving problems well and quickly, rather than the simulation as an end product.

4.1. The History of CIFM

CIFM is based upon modeling concepts that have been proven and tested in two large-scale and long-term modeling efforts. First, the Ship's Combat System Simulation (SCSS) was initiated in the late 1960's to study the ability of a ship and task force to survive a massive anti-ship attack. It was the first model that attempted to faithfully model the ship's combat system and its ability to react to a mass attack. Unlike previous anti-ship missile defense simulations, the combat system was shown to be a major player in the successful employment of the hard-kill weapons. The surge of information generated as the attack began, led to numerous information bottlenecks, as well as information being either lost or distorted. Also, because of the incorrect time sequence processing of the information, the human operators were often supplied information that would lead to incorrect decisions if blindly followed. SCSS is still being used throughout the US Navy for ship combat system studies. SCSS, unfortunately, was not general enough to handle ECM and ESM correctly. Deceptive jamming was particularly difficult to simulate. SCSS was also built to simulate one or many independently operating combat systems (operating in the anti-air warfare area), with no provision in the structure to allow different warfare areas of a combat systems to communicate or allow communications between combat systems, let alone have a command and control structure superimposed.

The second simulation employing concepts that are fully developed in CIFM was the System Level Air-to-Air Tactical Simulation (SLAATS), built in 1977 to evaluate the Advanced Medium Range Air-to-Air Missile (AMRAAM). It modeled air-to-air combat with enough fidelity that the utility of AMRAAM versus SPARROW could be evaluated. SLAATS also modeled the air to air avionics of the F-14, F-15, F-16 and F-18, and provided dynamic simulations of AMRAAM, SPARROW and PHOENIX. Additionally, SLAATS provided a realistic threat model. The model structure was enhanced to allow a much more flexible information flow to be accommodated. In this simulation, a system could easily communicate with other systems and a command and control structure could be imposed on a group of systems. This simulation, like SCSS, could not handle deceptive ECM easily and had a very rigid method of imposing C^3 structures.

In its present configuration, in use since 1983 by United Technologies Advanced Systems Division in San Diego, CIFM has solved the problems of simulating deceptive jamming in a realistic, yet simplified manner, and incorporating systems into hierarchical C^3 structures of any complexity. These structures can be modified dynamically within the model if desired. Therefore, CIFM can faithfully model the effects of destruction or reliability failure in parts of the C^3 structure and a subsequent reconfiguration of the network to allow the system to operate at some degraded level of effectiveness.

United Technologies Systems Analysis, in McLean Virginia, has used the present CIFM configuration to analyze the BM/C^{3}I requirements for the European Theater Tactical Ballistic Missile Defense Architecture. CIFM has also been employed to study the fire control and guidance algorithms used in the Space Based Kinetic Kill Vehicles portion of the Strategic Defense Initiative. Efforts are underway to also apply CIFM to such areas as helicopter operations and Anti-Submarine Warfare.

CIFM is an established modeling method that has undergone two major revisions, and each time the best features of the previous version have been retained.

4.2. The CIFM Framework

The CIFM Framework is the large body of code that is already written and is common to all applications. The CIFM Framework acts as an interface between the programming language compiler (SIMSCRIPT II 5), and the application code, much like an operating system provides an interface between the computer hardware and a user at the terminal. The Framework is highly structured, using object oriented coding practices, and supplies the standardization necessary for ease of model configuration and common element description formats. The Framework contains all the connections, pointers, etc., necessary to properly connect the elements of the simulation as they are developed. Elements of the system are defined as nodes, and the Framework contains the support structure necessary so that each new node can be designed by defining the parameters in a structured template. The CIFM Framework also contains a number of special nodes called "Monitors," that will be discussed in the next section.

4.2.1. Nodes.

The basic construct or building block of CIFM is the node. Each node can be thought of as having an input processor, a processing section and an output processor (See figure below). In creating a model, the modeler is usually simulating the actions of various systems such as radars, computers, vehicles, humans, etc. Each node is a modular construct and is conceived and implemented as an entity separate from other nodes in the model. The inclusion or deletion of a node does not affect the rest of the model, and the design of a node does not require consideration of the other nodes, other than their functional relationships in the real world. The nodes may be constructed and tested in any order, and they need not all be constructed in order for the model to begin working.

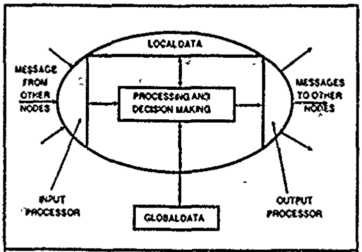


FIGURE 1 CIFM NODE

The input and output processors of a node are collectively referred to as the node's message processor. The message processors for different nodes have an identical structure but perform differently because of the implementation of node-specific messages for that node. The structured approach to coding is illustrated by the fact that a node template has been designed as part of CIFM that can be used as a starting point for writing any node. This template contains all the code that is common to all nodes, and highlights the sections that must be written by the programmer.

4.2.2. Links.

When large-scale systems are being modeled, the way in which their elements interact is usually of critical importance. In CIFM, these interactions are accomplished via links, which define the paths over which the interactions can occur. After the user creates a link, the nodes that are linked can communicate, while nodes that are not linked cannot communicate directly but can go through a special node such as the Radiation Monitor (see next section). It should be noted that the links are not created when the nodes are constructed, but rather they are created when a particular run is made. This means that systems can be easily reconfigured on a run-to-run basis by merely changing the input data files.

4.2.3. Messages.

In CIFM, information is transferred among the elements by means of messages. The message construct is very flexible and is designed to carry any amount of any type of information. Messages are transferred according to the link structure in place at the time the message is being generated. A CIFM model is completely driven by the message traffic that occurs within it. All activity is the result of acting upon a message in the most general sense, and thus it is extremely well suited to investigating C^{3}I oriented systems where the issue is evaluating the effects of changes in the connectivity and message patterns within combat systems.

5. MONITORS

This section describes five special entities in the CIFM framework called Monitors. Monitors are unique nodes that appear in almost every application and perform a number of important and far-reaching functions. The use of these monitors helps make CIFM the unique and capable model that it is, because they are already written and debugged, and they alleviate much of the work normally done in developing a new simulation. The following subsections provide a description of each of the CIFM monitors.

5.1. Radiation Monitor

In modeling information flow in large scale combat situations, one of the most important phenomena to simulate is electromagnetic radiation. CIFM is designed with this in mind and has a special node called the Radiation Monitor to keep track of all electromagnetic radiation present in the simulated world. The word "node," in reference to the Radiation Monitor, is somewhat misleading because the Radiation Monitor does not represent a real object. The Radiation Monitor stands alone (not linked to any other nodes) and is not part of the "Real World." It can be thought of as an invisible entity that knows the levels of all kinds of electromagnetic radiation at all points in space.

The Radiation Monitor is concerned with only certain kinds of nodes: those that emit electromagnetic radiation; those nodes that can reflect it; and those that sense it. The Radiation Monitor keeps track of the activity of all these nodes, and keeps track of all the radiation and propagation associated with them. It performs all required calculations to determine reflections and properly degrades the signal strength of the emission, accounting for the propagation losses from the natural propagation of radiation, atmospheric losses, and reflection of the energy from reflective targets. Jamming, both noise and deceptive, is also handled by the Radiation Monitor.

The Radiation Monitor handles radio communications as well. The signal is subject to the same losses as regular emissions, but a message is also passed. Messages transmitted in this manner can be jammed.

5.2. Motion Monitor

For any node to move correctly through the external world, the forces and moments acting on that node must be correctly calculated and properly integrated over time. The Motion Monitor is designed to carry out these calculations over all moving nodes within the system, and can handle both three and six degree-of-freedom aerodynamics when the problem calls for them. Depending on the degree of fidelity required for the underlying analysis, and the level of sophistication of the equations of motion, different integration schemes are available.

5.3. Intercept Monitor

In most applications of CIFM, there are a number of situations where two objects will come close to each other with the intent of one to do damage to the other in some way. The Intercept Monitor keeps track of the motion of all objects in the simulation and performs a number of functions when they are close enough to interact. Terminal target detection and/or fuzing is handled in the Intercept Monitor, along with any munitions effects that might be present. Kill probabilities are calculated and properly weighted random draws are used to determine losses in the appropriate situations. When one or both of the objects involved in the intercept are destroyed, the Intercept Monitor removes the killed objects, cleans up the associated memory locations and pointers, and does the tabulation updates required for the given output specifications.

5.4. User Interface Monitor

One of the key features included in CIFM is a facility to allow the user to communicate with and affect a run in progress. This is called the User Interface Monitor. It is a node in the same sense that the Radiation Monitor is a node, that is, it stands alone and is not part of the "Real World." In a batch run, the User Interface Monitor remains dormant, while in an interactive run it can be accessed periodically to interact with the run. By using this node, the user can interrogate any node to see its characteristics, send a message to any node, and see other characteristics of the run in progress. In addition, the value of any single variable in the simulation can be output to the screen. This capability is particularly valuable when debugging additions to the model.

The User Interface Monitor has been designed to be particularly user friendly. It is totally menu driven, contains error-checking routines so that unreasonable values will not be accepted, and is flexible in the way it will accept commands.

5.5. Simulation Status Monitor

CIFM also includes a Simulation Status Monitor which can accumulate simulation data and generate statistics.

6. INPUT AND OUTPUT STRUCTURES

Input and output features, designed for ease of use and understanding are integral and important parts of CIFM. Because these features are in the CIFM Framework, they are part of the structure that is common to all CIFM applications, and thus a great deal of consistency exists among the models. This is important because a person can carry his or her knowledge of running one model over to a new model, and start out well into the learning curve for the new application.

6.1. Type and Run Data

There are two major types of data used in a CIFM model, type data and run data. Type data is data which is associated with a particular type of system (node) and does not change from run to run. It can be thought of as a built-in, or design characteristic of a particular devise. For example, the maximum number of missiles that can fit on a certain kind of launcher would be type data. On the other hand, run data is data that is not built into a system, and it can change for any run. An example in this case would be the number of missiles that are actually loaded on a certain launcher at the beginning of a scenario. Clearly, this could change in different situations of interest, while the launcher itself remains the same.

The input structure for a CIFM model is broken into several major files. The Type Data Base is a file containing a collection of type data that is to be used for the run. The User Input File contains the run data for the given run and may contain parameters that will supersede the elements of the Type Data Base. The third major input file is the Keyboard, corresponding to the keyboard of the user's terminal. The input structure for any application is essentially the same, and this saves time in developing a new model.

Two interesting and useful features have been built into the input file structure. One is that the Type Data Base may contain as much data as desired, but only those data which have a possibility of being used will be read into the model at execution. The other is that type data parameters that are in the User Input File will be read into the model before the Type Data Base is accessed. This feature can be used to supply data to the model that do not exist in the Type Data Base. If, however, versions of a certain section of data exist in both the User Input File and the Type Data Base, then the version in the User Input File will supersede the version in the Type Data Base

6.2. Output Format and Processing

The CIFM Framework uses a default condition of three output files. The Screen is the screen of the user's terminal. The Error File is a file containing all of the errors discovered during the execution of the program. The Standard Output File is a file that contains the normal informative output and the results of the run. The Final Report File is a file that contains selected system characteristics written in an individual event or summed form. In a given execution stream, it is possible to associate the Error File or the Standard Output File (or both) with the Screen, so that all the output goes to the user's terminal screen. Considerable latitude exists in CIFM to easily generate custom output files to meet the needs of the particular analysis being undertaken.

7. TESTING AND INTERACTION OPTIONS

CIFM has built-in test and debugging facilities which allow the user to both interact with the model as it runs and to test any part of the model in isolation from the rest of the nodes in the model being developed.

7.1. Test Node

The Test Node is a special node in CIFM. It is specifically designed to be inserted in place of any node in a system, in such a way as to emulate the chosen node. In this way, the linking structure and message flow to and from the emulated node can be tested. This provides a fast and efficient method for testing large and complex systems.

7.2. Built-In Debugging Facility

Included in CIFM is the capability to generate a detailed trace of messages and routines which aids in the debugging process. The information desired is specified in the User Input File by indicating the name of the node to be analyzed, the type of output desired, and the start and end times during the simulation when the information is needed. There are two possible outputs which can be generated using this facility. One output shows when the routines associated with the specified node are called and the other output shows detailed information that allows the user to trace through selected algorithms implemented in the code.

7.3. User Interface Monitor

The User Interface Monitor was developed to allow the user to interact with the simulation, and this has proved to be of great value in the debugging process. The user has the capability to run the simulation for a specified period of time, stop it, and then examine any variables of interest. Values can be changed interactively and the model can be restarted where it left off. This capability saves a tremendous amount of time when debugging a new node or when trying to ascertain the effects of new nodes on other nodes.

8. IMPLEMENTATION DETAILS

As a model is being constructed using CIFM, care must be taken to insure that the resulting model will be usable to the analysts that might need it over the duration of the project and, even more important, over the desired lifetime of the model. In addition to capitalizing on the robust structure already in the CIFM Framework, attention must be paid to coding practices in the application specific code, and to the documentation that will accompany the model throughout its lifetime. To this end, United Technologies follows established and effective coding and documentation standards in the development and documentation of CIFM models.

A FUNCTION-BASED DEFINITION OF (C2) MEASURES OF EFFECTIVENESS

Commander Paul E. Girard, USN

ASW Systems Department
Naval Ocean Systems Center
San Diego, California 92152-5000

Abstract

The perennial problem of defining Measures of Effectiveness and Measures of Performance (MOE/MOP) has consistently failed to differentiate between the two in any succinct way. This may be due to the lack of an integrating concept of System Analysis in which to embed the MOE/MOP set.

As a result of an application of an analysis of hierarchical objectives (Rahmatian) for Antisubmarine Warfare, a Systems Analysis methodology reflecting the spirit of the Mission Oriented Approach (Signori) has been devised. In the process, a simple, unambiguous definition of MOE has evolved, to wit, the Probability of achieving the Mission Objective Function(s), given a set of constraining or prior conditions. All non-probabilistic metrics (such as throughput, power, gain, speed) are defined as Measures of Performance derived from System Parameters (such as size, color, fuel capacity).

This distinction between MOE and MOP is simple and definitive. It also provides a direct relationship between MOE and Mission Objective Functions. Realizing that decision making is the Command Objective Function provides a structure for relating C2 MOE's and MOP's to Force and System MOE's.

INTRODUCTION

Within the Space and Naval Warfare Systems Command, the Warfare Systems Architecture and Engineering (WSA&E) Directorate (SPAWAR-30) directs the development of architectural descriptions and assessments of current and future Naval Warfare Systems under the sponsorship of the Deputy Chief of Naval Operations for Naval Warfare (OP-07). In collaboration with the Antisubmarine Warfare Directorate (SPAWAR PD-80), the ASW Architecture Division (SPAWAR-315) has solicited the Naval Ocean Systems Center to lead a team of Navy Laboratories to address ASW Architecture. The process is initiated by the issuance of a Top Level Warfare Requirement (TLWR) by OP-07. In response, the Architecture team is attempting to devise a means of providing a traceable accounting of the relationship between system performance and the TLWR. This has given rise to the development of a methodology for Architectural description, modelling and assessment which is on-going. A by-product of this methodology is a function-based definition of measures of effectiveness. This paper focusses on how to utilize that definition to establish C2 measures of effectiveness.

After a review of some motivating concepts, a summary of the methodology will be presented. The process results in a general definition of MOE's and MOP's. This will be followed by a discussion of the role of Command decision making in the execution of functions. The definition of a canonic C2 MOE is supported by an example and this provides motivation for application of the definition to future analyses.

BACKGROUND

There appears to be a concensus among the C3 community that the justification for C3 systems must be based on the combat or mission outcome. In other words, the effectiveness of decision making (and decision support systems) has no meaning outside the context of a mission or purpose. Conversely, the historical approaches to modelling and assessment of operational systems have implicitly assumed "perfect C3". This results in optimistic forecasts of performance, not an accurate model. In other words, the expected outcome of missions can not be properly modelled unless the effect of decision making is included. These two complementary ideas suggest a synergistic relationship between mission analysis and decision analysis.

In fact, that relationship is one of cause and effect; the mission is not executed unless it is initiated by a decision to carry it out. In other words, it is the role of Command (decision making) to initiate required mission functions. This involves recognizing which functions are required or will be effective (or appropriate or authorized) and allocating available resources under his control to carry them out. Of course, the initiation must be accomplished in a timely manner, that is, early enough for the mission to be carried out before the enemy accomplishes his objectives, but not so early that more effective alternatives might be pre-empted. Figure 1 highlights three motivating concepts, just described, which will be incorporated in the approach.

MOTIVATING CONCEPTS

EFFECTIVENESS OF DECISION MAKING HAS NO MEANING OUTSIDE THE CONTEXT OF A MISSION OR PURPOSE

EXPECTED OUTCOME OF MISSIONS CAN NOT BE MODELLED UNTIL THE EFFECT OF DECISION MAKING IS MODELLED

THE ROLE OF COMMAND (DECISION MAKING) IS TO RECOGNIZE AND INITIATE REQUIRED MISSION FUNCTIONS AND ALLOCATE RESOURCES TO THEM IN A TIMELY MANNER

Figure 1. Motivating Concepts

Another concept which is depended upon heavily is one of a hierarchy of objectives (Rahmatian). Although it is not an original idea, Rahmatian provides a simple picture (Figure 2) of the interlocking role of the objectives (See note 1). From an arbitrary level, "what" is done is done for a higher level purpose ("why") and "how" it is done becomes a lower level "what" whose purpose ("why") is at the original level. The right side of figure 2. recasts the hierarchy of objectives in terms of Missions, Functions and Tasks. For a particular Force or System, its functions are what it does in order to accomplish its mission. Its tasks are its subfunctions, which are performed by its parts or subsystems.

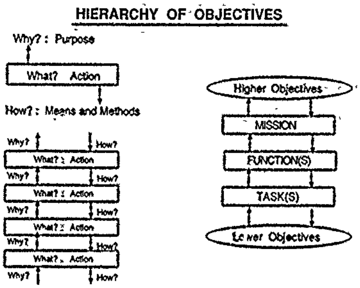


Figure 2. Hierarchy of Objectives

The Navy states its Mission, Functions and Tasks in Naval Warfare Publication (NWP)-1, Strategic Concepts of the U.S. Navy. By law, the Mission of the Navy is to "be prepared to conduct prompt and sustained operations at sea in support of U.S. national interests". The Functions of the Navy are to perform Power Projection and Sea Control. Recently the Navy has been assigned the job of Sealift and an unstated objective has always been to defend the United States. NWP-1 goes on to state that the Tasks of the Navy consist of Warfare Tasks and Support Tasks. Among the Warfare Tasks is Antisubmarine Warfare (ASW). Based on the hierarchy of objectives, therefor, the Mission of ASW Forces is to conduct ASW operations in support of Power Projection, Sea Control, Sealift and Defense of the United States. What, then, are the Functions of ASW Forces? This is the purview of the ASW TLWR and is the basis of the functional analysis embodied in the methodology.

METHODOLOGY

The approach (Figure 3) consists of performing a functional decomposition in each of three context-setting domains (Mission, Organization, and Resource). This decomposition is driven by the Mission Success Criteria (MSC) and Required Capabilities (RC) established by the TLWR. As we shall see later, the MSC's and RC's actually establish the initial tiers of functions. The functions within these domains (Warfare or Support Mission Functions, Command/Decision Functions, and Equipment or Personnel Functions, respectively) are mutually supportive in achieving the goals of the Mission and establish a hierarchy of objectives within each domain as well as across them. Realizing that the achievement of the objective is the criterion of success establishes a one-to-one correspondence between functions and metrics. In fact, this correspondence was recognized while reviewing the MSC's and RC's. In turn, the relationship among functions also corresponds to the mathematical relationship (equations) among metrics at the same level. This concept, too, has its roots in the relationship between the MSC's and RC's of the ASW TLWR, as we shall see. This suggests the potential for finding a relationship for aggregating metrics for resources, organizations and missions to the achievement of the TLWR objectives.

Note 1. This simple picture masks the complexity of the true interrelationships of functions. A true functional decomposition is not a pure tree and may not be strictly hierarchical Rather than a tree, a functional structure is more properly represented by a graph. If it is hierarchical, a multiplicity of functions at one level may support or be supported by a multiplicity of functions at another level. Then again, it may not be hierarchical, in that there may be cyclical or mutually supportive purposes, such as sinking submarines in order to protect supply ships which provide logistic support in order to sustain operations to sink submarines. The interdependence or lack of strict ordering of functional relationships does not negate the usefulness of the concept They will result in simultaneous and higher order equations.

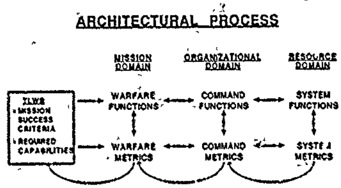


Figure 3. Architectural Process

Figure 4 summarizes the elements of the ASW TLWR. For twenty-two stressing cases, the mission was stated in terms of the type of ASW Mission, Area (interdictive) or Local (protecting other forces), and the nature (peace, crisis or war), region and timeframe of the conflict. Thus a typical Mission context might be to "conduct Area ASW in the Norwegian Sea during the first phase of global conventional war. Then the Mission Success Criteria for that Mission are stated, such as, attrite a percentage of the expected Order of Battle. These are the firm "Top" Level Warfare Requirements. The Required Capabilities were devised to exhibit a degree of credibility to the potential of achieving the MSC's. The RC's are selected factors in an equation relating several performance factors to the MSCs. Examples of RC's are Kill probabilities given Detection, Classification and Localization or the Area that can be searched to achieve a specified Detection probability. The equation that aggregates the RC metrics to the MSC metrics is called the audit trail

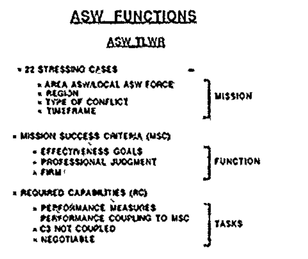


Figure 4 ASW Missions, Functions and Tasks

Examining the MSCs and RCs and other elements of the audit trails reveals most can be stated in terms of the probability of successfully accomplishing a function. This suggests that a function defines its own metric, which is the probability of a successful outcome of performing the function. Conversely, a probability metric defines the outcome of its related function. For example, the probability of detection identifies the detection event as the outcome of the search function. In this way, these functions and their related metrics are inseparable. By induction, subfunctions have their related (sub)metrics. The RCs, therefor, define a set of subfunctions for the functions defined by the MSCs. These subfunctions can be associated with the Task designation in the hierarchy of objectives.

When a set of subfunctions is carried out in some prescribed manner, we call that a procedure. This represents the implementation of the higher order function in terms of its subfunctions. If this procedure is modelled, the resulting equation or simulation prescribes the relationship of the lower level metrics to the higher level ones. Figure 5 shows some simple examples of this convention.

FUNCTIONS AND METRICS

•INSEPARABLE

FUNCTION ←——→ METRIC

KILL SUB — PROB (KILL SUB)

DETECT, GIVEN CUE — PROB (DET/CUE)

•PROCEDURES PRESCRIBE AGGREGATION EQUATIONS

–SUBFUNCTIONS PRESCRIBE SUBMETRICS

–EQUATION REFLECTS PROCEDURAL COMBINATION OF SUBFUNCTIONS

–EXAMPLE:

FUNCTIONS & PROCEDURES — METRICS & AGGREGATION

DETECT → CLASSIFY → ATTACK

Figure 5. Relationship of Functions and Metrics

Although the audit trail only relates two levels of functions and metrics, further examination of the nature of the requirements reveals more relationships. The twenty-two stressing cases clustered into three groups. One dealing with Area ASW in peacetime, low conflict or crisis situations, one for Area ASW in global conventional war (nuclear war has not been included here) and one involving Local ASW in war. There is an increasing scope of operations implied in these three groups as well as a correlation with the size or aggregation of forces involved. When the representative functions associated with the metrics for these groups are displayed as in Figure 6, a pattern emerges of a hierarchy of objectives, the top of which depends on the relevant group of situations or missions. (The top of each group, of course, supports higher level objectives but these are beyond the scope of ASW.) This hierarchy represents a decomposition of Warfare Functions within the Mission Domain of our architectural approach. A later section will address the Command Function decomposition within the Organizational Domain. The System Functions that are defined in the Resource Domain are implementations of the Warfare and Command Functions. These System Functions are supported by Equipment Functions (such as Process, Store and Display) or Personnel Functions.

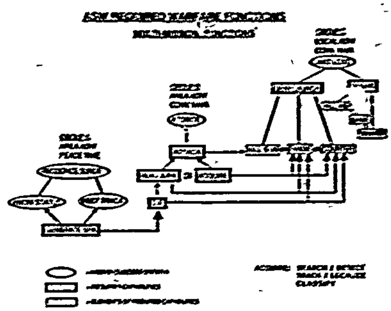


Figure 6. Hierarchy of ASW Missions and Functions

Some of the factors in the audit trail equations are not stated in terms of probabilities. Some of these can be recouched in probabilistic form but some can not. The latter often represent characteristics, such as speed, that are normally not event oriented so probabilities are not appropriate metrics. Others, such as throughput, may involve events, but do not represent success or failure, at least not at that level of aggregation. This suggests a dichotomy of metrics into two classes, one of which is defined by the probability of successful accomplishment of a function.

MEASURES OF EFFECTIVENESS

Following this suggestion leads to the definitions of Measures of Effectiveness shown in figure 7. This way of differentiating among MOEs and MOPs is meant to emphasize the mission orientation of the MOE. In fact, in this approach, the Mission (or Function) defines the MOE. The MOP class then provides for the collection of metrics, such as gains, rates, capacities and delays, that are not probabilities of successful outcomes of functions. A third class of metrics represent physically measurable parameters. MOPs are the consequence of a configuration of physical elements. MOEs are the result of a combination of physical parameters, MOPs or other MOEs.

MOP/MOE DEFINITIONS

MEASURE OF EFFECTIVENESS*

- PROBABILITY OF SUCCESSFUL ACCOMPLISHMENT OF A FUNCTION
  - FUNCTION: PROCESS RESULTING IN AN OUTCOME OR EVENT
  - ALL PROBABILITIES ARE CONDITIONAL
  - DERIVED FROM MOPs AND OTHER MOE'S

MEASURE OF PERFORMANCE

- NON-PROBABILISTIC PARAMETER OF PERFORMANCE
  - CAUSED BY SYSTEM CONFIGURATION
  - DERIVED FROM PHYSICAL PARAMETERS OR CHARACTERISTICS
  - e.g. GAIN/LOSS, THROUGHPUT, POWER, CAPACITY (#TRACKS)

PHYSICAL PARAMETERS/CHARACTERISTICS

- TANGIBLE ASPECTS OF SYSTEM
  - LENGTH, WEIGHT, CHEMICAL DENSITY
  - NUMBER OF CHANNELS
  - COLOR

* MOFE, MOPE, MOSE, DEPENDING ON LEVEL

Figure 7. General Definition of MOEs and MOPs

The way that various MOEs are related was implied by the concept of the audit trail, above. In mathematical terms, the MSCs are the dependent variables and they are functions of the independent variables we call RCs. But MSCs and RCs can both be defined in the form of MOEs at the two levels of analysis implied by the audit trail. The analogous relationship can be extended to all levels of a hierarchy of objectives. Then, an MOE defined by a function at an upper level is a dependent variable and is a mathematical function of the MOEs defined by objective functions at a lower level. Furthermore, the MOE at the upper level is conditioned on the accomplishment of the lower level functions.

This leads to the realization that an important consideration in understanding the significance of this approach is the notion that all probabilities are conditional. They are all conditioned on the set (of events or conditions) which is the support for the probability density function. For example, the probability of detection is only defined for the condition that there is a target in the search area. (Otherwise, the event is a false alarm. This is an important complementary measure of effectiveness which supports the objective (or constraint) of not wasting resources.) This notion of conditioning also relates to the hierarchy of objectives. The Mission context is what establishes the conditioning on the highest level MOE and Function. For example, what is the probability of detection, given the peacetime mission in the Norwegian Sea in summer conditions? When the accomplishment of a function depends on the accomplishment of its subfunctions, the event must occur in the intersection of the subevents. For example, killing an enemy submarine involves detecting, classifying, localizing, attacking AND inflicting lethal damage, given there is an enemy submarine within prosecution range. Then the definition of conditional probability can be used to write the MOEs as chains of conditional probabilities, although analysts are usually guilty of improper notation. A proper audit of the conditional events may be critical to the analysis. In complex analyses involving concurrent or complementary objectives, the calculus of conditional sets and conditional probabilities that has been devised by (Goodman) and (Calabrese) will be needed to trace these relationships.

The aggregation of probabilities is not a new idea. It is very common in the assessment of operational effectiveness. But the hierarchy of objectives provides a way of formalizing the approach. The role of functions in the hierarchy will provide the setting for examining the effectiveness of C2 functions.

ROLE OF DECISION MAKING

The key to the definition of C2 MOEs is the role that decision making has as a function in the hierarchy of objectives. It is a function that occurs at every level of the hierarchy. It is the function that must occur if any other function is to be initiated. In other words, it enables all other functions. Figure 8 shows this concept in an adaptation of Rahmatian's symbology from Figure 2. "What" function to perform is determined by the "Decide" function at that level. "Why" it needs to be performed is established by the higher level objective(s) and is the motivation for the decision. "How" has the same meaning as before. At each level, the "Decide" function is further decomposed into the "Decision Functions" that are needed to make those decisions. We have called them Command Functions for obvious reasons and they too will play a role in the definition of C2 MOEs as will be shown.

The box in the lower right corner of Figure 8 acknowledges that this approach bears a strong similarity to and derivation from the Alphatech TR-293 referenced. The FROG acronym, however, is the property of the author. Alphatech used "processes" instead of "functions", but CORP was not aesthetically pleasing.

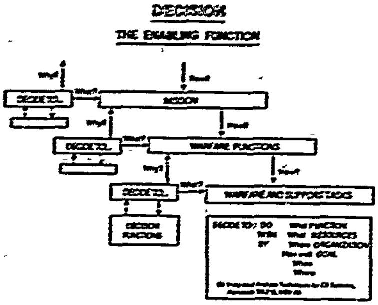


Figure 8. Decision as the Enabling Function

An example of a functional decomposition that incorporates the "Decide" function is shown in Figure 9. Starting with the objective to limit losses, it follows that we can accomplish that by countering the weapon or the enemy platform. Supporting functions are surveillance and cueing. The large "Decide" diamond represents the choice among these and /or when to execute them. The next level of decomposition is shown as ovals within the four boxes representing the first tier functions. The smaller diamonds represent a choice among the lower level functions, such as whether to attack or evade the platform or both. This process can be extended ad infinitum. The branching paths among the diamonds represents a hierarchy of decisions that are involved in controlling the activity of the organization and resources whose objective is to limit losses.

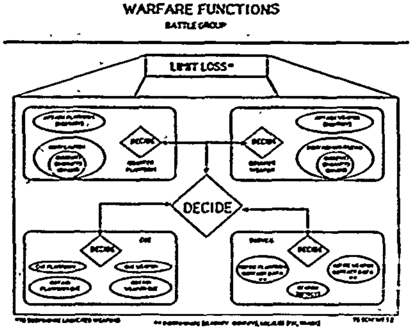


Figure 9. Warfare Function Decomposition with Decide

There is also a decomposition of the "Decide" function that supports the decision making at each level. Figure 10 depicts a two-tier decomposition of "Command Functions". These functions are related to the functions normally found in any concept model of decision making, such as the Lawson model or SHOR paradigm, et al. We often refer to it as POA&E, for Plan, Observe, Assess and Execute. The basic structure is intended to be a refinement of the HEAT whirlygig model. A more detailed description of the functions to the third tier is another product of the methodology development mentioned above. The POA&E is shown here in order to highlight two subfunctions, in particular, which are pivotal elements in the decision outcome. The culmination of the decision cycle is the recognition of the appropriate "current COA (course of action)" and the assignment of available resources to carry it out. These are the top two subfunctions of Execute.

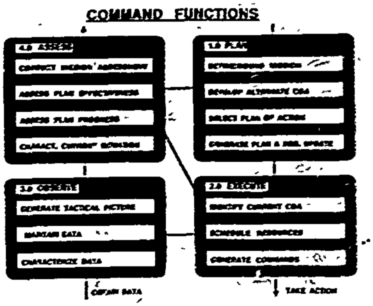


Figure 10. Command Function Decomposition

C2 MEASURES OF EFFECTIVENESS

Following the definition of MOE established earlier, the MOE for decision making must involve the probability of making a decision. Figure 11 presents a canonic form for an MOE that reflects the essence of the outcome of the decision process, which is to initiate a function with resources. The necessary conditions for making the decision are also used to "condition" the MOE. Authority is required for each of the elements of the decision: the function initiation and the resource allocation. In addition, resources must be available and information is needed to recognize that a situation exists which calls for the initiation of the function and other information is needed to know which resources are available to assign to the task. Finally, the MOE is conditioned on whether the Mission context (current situation) actually calls for the function as being appropriate. This last condition accommodates the concept of the "accuracy" of the decision, which is often mentioned as a C2 metric.

A CANONIC FUNCTION - BASED C2 MOE

PROBABILITY OF DECIDING TO

INITIATE FUNCTION
WITH RESOURCES

(AT A TIME TO EXCEED THE GOAL)

GIVEN

AUTHORITY TO INITIATE FUNCTIONS
AUTHORITY TO CONTROL RESOURCES
INFORMATION NEEDED TO MAKE DECISION
RESOURCE AVAILABLE TO PERFORM FUNCTION

(SITUATION CALLS FOR THE DECISION)

Figure 11. Canonic Definition of C2 MOE

Another classical C2 metric is timeliness. This is represented by the requirement to initiate the function in time to meet or exceed the goal. The MOE for the mission oriented function will be dependent on and conditioned on the time that it is initiated. If the time of this initiation is within the "window of opportunity", we say that the decision was timely. In fact, the probability of accomplishing a function is a variable function of time. This is also true of the probability of making a decision. Since the Mission MOE is dependent on the decision MOE, their time factors will also be related. Future work is intended to address this relationship.

A third oft-mentioned C2 metric is completeness. No aspect of the proposed C2 MOE addresses this. I am not sure that completeness is a necessary metric. It usually applies to the information available to the decision maker. The issue here is sufficiency of information, not absolute completeness. Completeness may be an MOP by the definition above and may be useful in assessing equipment performance.

IMPACT OF C2 MOE ON ANALYSIS

Figures 12 through 14 provide an example of how the C2 MOE fits into an analysis. In Figure 12, the normal Probability of Kill is calculated as the product of the Conditional Probabilities of Detection, Classification, Localization AND Kill. When this is aggregated over all the targets, the Mission Success for Attrition is realized. The Probability of Detection, in addition to being conditioned on the presence of a target, as was mentioned earlier, is also conditioned on the initiation of the Search function. The Probability of Initiating and Allocating Resources to Search is the C2 MOE in this case. It is shown to be conditioned on whether or not a cue has been received. This reflects the difference in likelihood of a decision maker initiating a search, depending on whether there is information available to limit the search area. The Probability of Detection will be the product of the detection probability conditioned on the search and the search decision probability. Note that the conditional detection probability is also dependent on the availability of cueing information. This suggests a magnification of the effect of a cue since, not only is the detection probability increased, the decision probability is also higher.

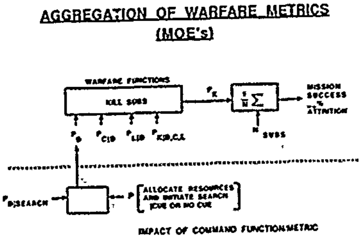


Figure 12. Impact of C2 MOE on Warfare Analysis

In Figure 13, the classical role of the sonar equation in determining the probability of detection is shown schematically. This also provides a representation of the role of the definitions for MOP and physical parameters, as noted by the small numerals in the figure.

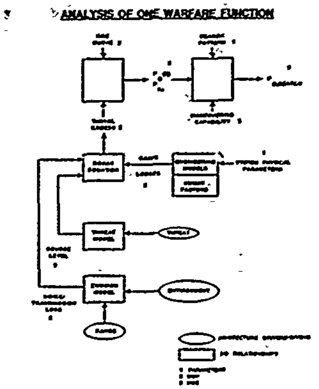


Figure 13. Standard Analysis of Detection Probability

Figure 14 is a pictorial representation of the decision making process (POA&E) and its requisite conditions that provide the basis for the decision outcome. A variable not mentioned previously is the human factor element. As we know, this will have a major influence on the Probability of Deciding.

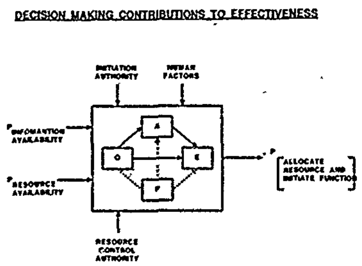


Figure 14. Decision Making Product Relationship

SUMMARY

The final example only shows the role a decision probability has on the outcome of a single mission function. Consider that the concept is intended to apply to the initiation of each and every function involved in the operation. While it may be highly likely that the right decisions will be made when the conditions are right and the information and resources are available, the large number of decisions that are involved implies that even small differences from unity will accumulate to deflate the overall effectiveness of the Force. The proposed MOE provides the analytical means to account for the effect of "imperfect C2". It also provides the means to couple the availability of information and resources to the mission outcome. From an organizational perspective, the effect of delegation of authority to initiate action or control resources is explicit and provides the potential for examining dynamic organizational issues.

The Canonic Function-Based C2 Measure of Effectiveness satisfies the motivational concerns shown in Figure 1. The MOE is embedded in the Mission-driven hierarchy of objectives so that the function to be performed provides the context for assessing the effectiveness of the decision making. The MOE is also embedded in the aggregation equation for the operation so that the overall effectiveness is dependent on the outcome of the decision process. Finally, the MOE reflects the purpose of decision making directly, which is to allocate resources to perform functions in a timely manner. The canonic form provides a common definition for all cases and a means for orienting the analyst to the essential conditions to be accounted for, which are information and resource availability and authority to carry out the assigned Missions, Functions and Tasks.

ACKNOWLEDGMENTS

My thanks to Bryson Pennoyer, George Ruptier and Jay Martin of the Naval Ocean Systems Center and Janis Bilmanis of the Naval Surface Weapons Center, White Oak, for their intense efforts in the development of the architectural methodology. And to Vic Monteleon of NOSC for the clue that the purpose of C2 is to allocate resources.

REFERENCES

Rahmatian, Sasan. "The Hierarchy of Objectives: Toward an Integrating Construct in Systems Science" *Systems Research* Vol. 2, No.3 pp. 237-245, 1985

Signori, Dr. David T., and Starr, Dr. Stuart H., "The Mission Oriented Approach to NATO C2 Planning" *Signal*, September 1987, pp. 119-127

Wohl, Joseph G., and Tenney, Robert R., *Integrated Analysis Techniques for Command, Control, and Communications Systems*, Burlington, Mass., ALPHATECH, Inc. Technical Report TR-293, November 1986.

Goodman, I.R. "A Measure-Free Approach to Conditioning", *Proceedings of the Third AAAI Workshop on Uncertainty in Artificial Intelligence*, July 10-12, 1987, Seattle, Washington, pp. 270-277

Calabrese, Philip, "An Algebraic Synthesis of the Foundations of Logic and Probability" Information Sciences 42, 1987, pp 187-237

Naval Warfare Publication 1, *Strategic Concepts of the U.S. Navy*, Unclassified

ASW Top Level Warfare Requirement, Chief of Naval Operations, 1987, Secret

A SYSTEMS APPROACH FOR RELATING BM/C^3 FUNCTION PERFORMANCE TO MILITARY SYSTEM EFFECTIVENESS

J.J. Shaw, P.E. Coughlin, and W.S. Powell

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

ABSTRACT

We report here progress on research conducted to relate the performance of Battle Management functions to overall system effectiveness*. We describe a method that captures this relationship by expressing the operation of a BMC^3 system in terms of a time-varying finite-state Markov chain.

We also present several analysis methods based on two important sensitivity measures that are derived from this relationship: pseudo-costates and System Effectiveness Derivatives. These two measures offer interpretations that permit the identification of time epoches during a conflict when the BMC^3 system under study is most sensitive to both favorable and unfavorable function outcomes, and to processing bottlenecks. These measures support diagnostic analysis of BMC^3 systems, and should prove useful in the development of more efficient Battle Management systems.

SECTION 1. INTRODUCTION

1.1 OBJECTIVE OF THIS PAPER

We present in this paper a method to relate the performance achieved by Battle Management functions to the consequent effectiveness achieved by the larger military system served by those functions. The quest for such a relationship is by no means new, having been the subject of intense research over the last ten years [1, 2, 3, 8], and the motivation for working groups sponsored by the Military Operations Research Society (MORS) [4-7], and represented at MIT/ONR and the JDL C^2 Symposia [8-11]. While several of these earlier efforts sought a theory for C^2 systems at large, we seek here a more modest goal: a method to relate function performance to system effectiveness.

There are two critical distinctions between earlier efforts and the method we develop and present here. First, our method aims to describe behavior without prescribing how BMC^3 systems should conduct operations to maximize some measure of effectiveness. Second, our emphasis is on performance measures for automated functions, meaning that our method need merely describe the behavior of those portions of a C^2 system that have a pre-programmed decision logic. This second distinction is important because automated functions exercise only a finite number of decisions, thereby permitting us to model both the determinants and consequences of decisions in a succinct manner.

The systems of interest to us here bear strong similarities to assembly lines (admittedly, stochastic assembly lines) where Battle Management functions operate in concert to advance the larger military system from one state to the next during a conflict. We illustrate this analogy in Figure 1 from the point of view of threats being engaged by a defense system. The circles in this figure denote functions performed against threats, the arcs indicate potential outcomes from any function, and flows along these arcs represent the threats undergoing processing by this system. The processing on any assembly line is always prone to some error or incomplete work, and so it is in this example. Nevertheless, there is a general pattern to the processing and, more to the point, the BMC^3 system must successfully advance each threat through a requisite set of processing states to achieve a successful engagement. Thus, we might ponder how an improvement in performance at some stage in this assembly line leads ultimately to an improvement in effectiveness for the larger system. Indeed, a corollary to this inquiry is the question "where are improvements in performance most needed?" The research described here seeks answers to these questions.

1.2 THE ELEMENTS OF THE PERFORMANCE MEASURE METHOD

The method we describe encompasses a quantitative relationship between function performance and system effectiveness and analysis methods for computing sensitivity measures from these relationships. The relationship captures the operation of an arbitrary BMC^3 system as a time-varying finite-state Markov chain, and is accompanied by a set of principles for assembling a state-equivalent model for a Battle Management system. The analysis methods quantify key sensitivity relationships between function performance and system effectiveness using state-transition statistics collected for the BMC^3 system of interest. We direct the reader to [13] for a complete description of the methodology.

1.3 ORGANIZATION OF THIS PAPER

The remainder of this paper spans five sections. We develop the relationship between function performance and system effectiveness in Section 2, and present associated analysis methods in Section 3. We present in Section 4 the objectives and the test conditions for experiments we conducted to demonstrate the method, and present computational results in Section 5. We conclude this paper with remarks on the methodology, and comment on future modeling directions in Section 6.

*The research described here was conducted for Dr. Doyle Thomas at the U.S. Army Strategic Defense Command (USASDC) under contract DASG60-87-C-0004 to Unisys Corp.: "Performance Measures for System Battle Management Controller

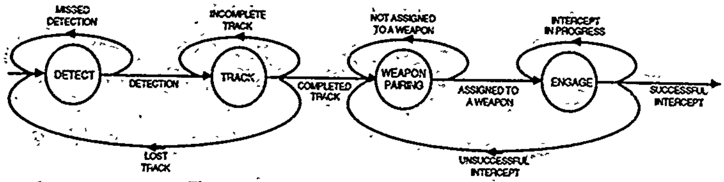


Figure 1. Battle Management Viewed as an Assembly Line

SECTION 2. A QUANTITATIVE RELATIONSHIP BETWEEN PERFORMANCE AND EFFECTIVENESS

2.1 BMC³ PERFORMANCE WITHIN DYNAMIC MILITARY SYSTEMS

Consider a military system that can be in any one of a number of states S at any time during a conflict. We defer the question just what constitutes the 'state' for this system, except to note that it can denote the current disposition of each object of concern to the system (threat, sensor, weapon, Battle Management node, asset at risk, etc.), or it can denote a statistical measure of current dispositions taken over objects in the aggregate. We will consider this issue in more detail later in this section. Nevertheless, we introduce the distinction here between terminal and intermediate states; the former denoting the set of states that are reached by the system at the end of the conflict, and the latter denoting the set of states that the system can reach any time prior to the end of the conflict. We also introduce the notion of desirable and undesirable terminal states to denote states that the C^2 system would like to reach or avoid, respectively, at the end of the conflict.

We also assume this system exercises a set of functions, F_t, at every time period to maintain the system within a desired set of states. Were we to carefully monitor the activity of a function at a particular instant in time we would discover that the function achieved a certain processing throughput, produced a certain set of decision outcomes, and perhaps even committed a certain number of 'errors' in its decisions. We use the notional term 'operating condition' to refer to these activities as a whole, and let f_t denote the operating condition of function f at time t.

We assume that this system seeks a sequence of states throughout the duration of a conflict that minimizes an expected cost incurred at the end of the conflict:

$$\text{MIN } E[J] = \sum_{i \in S} \rho_T(i)\, g_T(i) \qquad (2\text{-}1)$$

where:

S: the states that the system can reach (with |S| denoting the number of possible states)

$\rho_T(i)$: the probability that the system is in state i at the end of the conflict; and

$g_T(i)$: the cost to the system if it is in state i at the end of the conflict.

In short, this system faces a terminal cost optimal control problem. (We state without proof that the general case where the cost is also accumulated at every time step can be reduced to the terminal case problem; we direct the reader to [12] for details.)

The reader should note that we have not imposed any restriction on the functional form of the cost function, $g_T(\cdot)$; this term can be a scalar or a vector, linear or nonlinear.

We express the system state dynamics in terms of probabilities of state transitions:

$$\rho_{t+1} = P_t(\rho_t, F_t)\, \rho_t \qquad (2\text{-}2)$$

where $P_t(\rho_t, F_t)$ represents the state transition probability matrix for time t, and the initial state distribution, ρ_0, is given. $P_t(\rho_t, F_t)$ is square and of dimension equal to |S|, and is functionally dependent on the current state probability distribution, ρ_t, and on the current operating conditions of the functions acting on those states, F_t. (In the interests of brevity, we will express this matrix simply as P_t, and refer to the arguments only when the discussion requires.)

We introduce the vector J_T to denote the terminal cost to the system on a state-by-state basis:

$$J_T = \begin{bmatrix} g_T(1) \\ g_T(2) \\ \cdot \\ \cdot \\ g_T(|S|) \end{bmatrix} \qquad (2\text{-}3)$$

The reader will note that the righthand side of Equation 2-2 can be re-written in vector form as $\rho_T' \cdot J_T$ where ' denotes the transpose. We also introduce the vector J_t to denote the cost-to-go from time t on a state-by-state basis, and compute the values within this vector through the following backward-moving recursion beginning at time T-1:

$$J_t = P_t' \cdot J_{t+1} \qquad (2\text{-}4)$$

We can partition the states that can be reached from an arbitrary state i at time t into a desirable set, $D_t(i)$, and an undesirable set, $U_t(i)$, where:

$$J_t(i) \geq J_{t+1}(j \mid j \in D_t(i)) \qquad (2\text{-}5)$$

$$J_t(i) \leq J_{t+1}(j \mid j \in U_t(i)) \qquad (2\text{-}6)$$

The reader will note that we use the categories 'desirable' and 'undesirable' in terms of intermediate states that are reached from a particular (reference) state at a particular time in the conflict. The implication here being that a 'desirable' transition from a particular state is more likely to put the system on a trajectory towards a desirable terminal state than would an 'undesirable' transition. This is not to imply, however, that the intermediate states reached from a desirable transition are necessarily desirable terminal states in their own right.

We seek a measure to account for the change in the cost-to-go from a state were the BMC^3 system slightly more successful arranging favorable state transitions from that state. We introduce the term the pseudo-costate to represent this measure; the pseudo-costate represents the derivative of the effectiveness measure with respect to the likelihood that the system will make a transition to a desirable state. We first illustrate the derivation of pseudo-costate for a state that has only two exit transitions, and then derive the expression for pseudo-costate for the more general case.

Let us suppose that the system is in state i at time t, and can reach one of only two states j and k at the next time period with probabilities α and $(1-\alpha)$, respectively. The value of the cost-to-go from state i at time t is given by:

$$J_t(i) = \alpha J_{t+1}(j) + (1-\alpha) J_{t+1}(k) \qquad (2\text{-}7)$$

Let us suppose that $J_t(k) > J_t(i) > J_{t+1}(j)$: state j is the desirable state to reach from state i at time t, while state k is the undesirable state to reach from i. Suppose that the BMC^3 system can increase by ϵ the probability that the system will go from state i to state j. From inspection, the net change in the cost-to-go from state i is:

$$\Delta J_t(i) = \epsilon \left(J_{t+1}(j) - J_{t+1}(k) \right) \qquad (2\text{-}8)$$

leading to the following expression:

$$\Gamma_t(i) = \frac{\partial J_t(i)}{\partial \alpha}$$

$$= \lim_{\epsilon \to 0} \frac{\Delta J_t(i)}{\epsilon} = \left(J_{t+1}(j) - J_{t+1}(k) \right) \qquad (2\text{-}9)$$

where $\Gamma_t(i)$ denotes the pseudo-costate from state i at time t.

In general, the system may be able to reach an arbitrary number of states from a particular state. The value of the pseudo-costate for any state is the value of the average cost-to-go from desirable states minus the value of the average cost-to-go from undesirable states, or:

$$\Gamma_t(i) = \frac{\sum_{j \in D_t(i)} P_t(i,j)\, J_{t+1}(j)}{\sum_{j \in D_t(i)} P_t(i,j)} - \frac{\sum_{j \in U_t(i)} P_t(i,j)\, J_{t+1}(j)}{\sum_{j \in U_t(i)} P_t(i,j)} \qquad (2\text{-}10)$$

where $P_t(i,j)$ denotes the probability that the system will go from state i at time t to state j by time $t+1$.

Let us suppose that we have observed the behavior of a system for nominal operating conditions of the functions with the state transition probability matrices $P_0, P_1, \ldots P_{T-1}$; associated with this system is a nominal value of the effectiveness measure $J = E[g_T(X_T)]$. We would like to establish the derivative of J with respect to the operating condition for function f at time t. The value of the derivative we seek is expressed by the following equation:

$$\frac{\partial J}{\partial f_t} = \sum_{i \in S_f} \rho_t(i) \frac{\partial \alpha_t(i)}{\partial f_t} \Gamma_t(i) + \sum_{\tau > t} \sum_{i \in S_f} \rho_\tau(i) \frac{\partial \alpha_\tau(i)}{\partial \rho_\tau} \Gamma_\tau(i) \qquad (2\text{-}11)$$

where S_f denotes the system states that permit the operation of function f, and $\alpha_t(i)$ denotes the likelihood that transitions out of state i at time t will be desirable. The change in the performance of function f at time t leads to an immediate change in state transition probabilities; this immediate effect is captured in the first term in the righthand side of the equation above. The change in state transition probabilities at time t yields changes in state probability distributions in future time periods that lead to changes in the state transition probabilities in those time periods (due to the nonlinear nature of the state dynamics model in Equation 2-2). This persistent effect is captured in the second term in the righthand side of the equation above.

If we adopt a Markov approximation for our system, replacing $P_t(\rho_t, F_t)$ with simply $P_t(F_t)$, we can ignore the persistent effect to arrive at the following approximate relationship between function performance and system effectiveness:

$$\frac{\partial J}{\partial f_t} \approx \sum_{i \in S} \rho_t(i) \frac{\partial \alpha_t(i)}{\partial f_t} \Gamma_t(i) \qquad (2\text{-}12)$$

2.2 THE CHOICE OF STATE FOR A SYSTEM

What constitutes the state of a BMC^3 system? In principle, 'state' at any time should denote the disposition of each object of concern to the system (a threat, a sensor, a weapon, a Battle Management node, an asset at risk, etc.). This is impractical, however, for a system involving thousands of objects even if every object has a binary state.

An alternate definition of state emerges when objects are collected into classes, with a finite number of states defined for each class: the system state is the Cartesian product taken over these separate classes. For example, we might create three object classes: THREAT, SENSOR, and WEAPON, and distinguish objects in the class 'THREAT' according to whether they have been 'detected', 'tracked in two dimensions', 'tracked in three dimensions', 'assigned to a weapon', etc. This approach does not presume that all objects within a single class are alike since we can always define the states within that class in such a way that we preserve important differences between objects appearing therein.

We can reduce further still the possible values of system state by considering only the states that can be reached by objects belonging to a single class. The idea here is to account for the states obtained by all other objects implicitly within the parameters of the Markov model used to describe the behavior of the system. While we continue to lose modeling precision with this extension, we gain both computational efficiency and the ability to assemble a Markov model of the system using fewer sample points (i.e., using fewer system-level simulations or exercises) than would otherwise be needed to assure statistical significance.

Under this definition, S denotes the set of states that can be reached by objects under consideration. Similarly, S_f denotes the set of object states that meet necessary conditions for processing by function f, these conditions having been determined by the rules of operation of the system.

We adopt this last definition of state in this paper when we examine a midcourse ballistic missile defense (BMD) system in Section 4. We model explicitly the states obtained by RVs - these being the primary focus the system - and account for the states of all other objects implicitly within the transition probabilities computed from repeated simulation of that system.

We have necessarily omitted several details that should be considered in the selection of 'state' for a BMC^3 system. In particular, we have not considered whether there is an obvious logic that one can use to define the states that are germane to a set of automated BMC^2 functions, nor have we considered how to associated functions and function characteristics (particularly function enablement and function outcomes) to states and state transitions. We direct the reader interested in pursuing the method further to [13] for our own reflections on these issues.

SECTION 3. ANALYSIS METHODS

In this section we present several analysis methods that emerge from the performance measures theory. The first method, based on the pseudo-costates introduced in Section 2, provides a way to identify processing bottlenecks and critical state transition timelines. The second method is based on a measure known as the System Effectiveness Derivative; this measure identifies time periods when overall system effectiveness is most sensitive to changes in the performance of any one function.

3.1 COSTATE ANALYSIS

In this subsection we provide an interpretation of the pseudo-costate terms we introduced in the previous section. The utility of Costate Analysis depends largely on being able to successfully interpret the meaning behind rising and falling pseudo-costate values for select system states over time.

It has been our experience that the pseudo-costates for many states typically have relatively small values at the beginning of the conflict. It is at the beginning of the conflict that a BMC^3 system typically has sufficient time to recover from unfavorable state transitions with little ill effect. A large number of the states that the system can enter are, therefore, largely indistinguishable from each other, thereby yielding relatively small pseudo-costate values. This phenomena is illustrated in Figure 2 where we present representative pseudo-costates for select states that can be reached by threats engaged by a ballistic missile defense system.

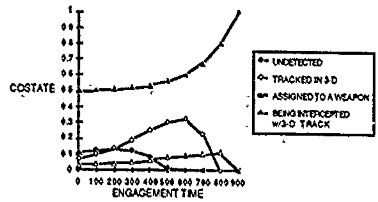


Figure 2. Representative Pseudo-Costate Profiles Over Time

The profile of the pseudo-costate for a particular state following the beginning of the conflict is often distinguished by one of two patterns. The first involves a monotonic rise in value followed by a monotonic decline in value. The rise in value occurs when the time required by the system to recover from an unfavorable state transition and reach a desirable terminal state is quickly approaching the time available for recovery (e.g., time remaining in the conflict). The costs-to-go from those intermediate states that are reached through unfavorable transitions grow faster than the costs-to-go from those intermediate states reached through favorable transitions, yielding an increase in the value of the pseudo-costate for that state. Conversely, the value of the pseudo-costate will decline when the time required to advance the system from an intermediate state reached by a favorable transition toward a desirable terminal state is quickly approaching the time remaining in the conflict. Alternatively, the decline in value can indicate that the system no longer has the resources needed to effect favorable changes in state. In short, monotonically rising and falling pseudo-costates reveal critical processing timelines for the system under study.

The second pattern is where the value of the pseudo-costate for a particular state rises and falls over several successive time intervals. This pattern typically indicates that processing bottlenecks are occurring in the system. A processing bottleneck prevents the system from capitalizing on a favorable transition from the state under consideration. This reduces the distinction between a favorable and unfavorable changes in state, and causes the value of the pseudo-costate value to decline. The removal of the bottleneck provides an environment where the system can capitalize on a favorable change in state, thereby causing the value of the pseudo-costate to rise. The oscillation in the value of the costate for the state in question corresponds to the periodic formation and removal of processing bottlenecks.

3.2 THE SYSTEM EFFECTIVENESS DERIVATIVE

We have established that Battle Management functions contribute to the effectiveness of the overall system by advancing the system along a trajectory of intermediate states to a terminal state. We developed a Markov model to express the performance of Battle Management functions in terms of state transitions that led to the relationship between function performance and system effectiveness presented in Equation 2-12. This relationship can be difficult to evaluate in practice, however, since it presumes we know how a change in the operating condition of the function in question leads to a change in the probability that the system will be advanced to a favorable state. Specifically, this expression incorporates the partial derivative $\frac{\partial \alpha_t(i)}{\partial f_t}$, where $\alpha_t(i)$ denotes the probability that the system will advance to a favorable state if it is in state i at time t.

We can derive a sensitivity measure that avoids this derivative altogether. This measure expresses the sensitivity of the system effectiveness measure with respect to the probability that the function in question will advance the system to favorable states. We refer to this as the System Effectiveness Derivative (SED), and express it through the following equation:

$$\vartheta_f(t) = \sum_{i \in S_f} \rho_t(i)\,\Gamma_t(i) \qquad (3\text{-}1)$$

where S_f denotes the system states that permit the operation of function f.

The SED is an __aggregate__ measure of sensitivity since it presumes that $\frac{\partial \alpha_t(i)}{\partial f_t}$ is constant for all states i that permit the operation of function f. The value of the SED for any particular function at any time is determined jointly by the probability that the system in in a state that permits the operation of the function, and by the magnitudes of the pseudo-costates involved. In this respect, a comparison of values of SEDs across functions reveals which functions are the major contributors to system effectiveness at any one time.

3.3 THE UNIT SYSTEM EFFECTIVENESS DERIVATIVE

A normalized version of the SED is the __Unit__ System Effectiveness Derivative (USED), and is given by:

$$\varphi_f(t) = \frac{\sum_{i \in S_f} \rho_t(i)\,\Gamma_t(i)}{\sum_{i \in S_f} \rho_t(i)} \qquad (3\text{-}2)$$

The USED has the same interpretation as a time-line sensitivity measure for functions that pseudo-costates have for system states. Indeed, oscillations in the value of the USED for a function typically signify the formation and removal of processing bottlenecks at one or more functions appearing after that function in the normal processing sequence.

SECTION 4: DEMONSTRATION OF PRINCIPLE

We describe in this section the analyses we conducted to demonstrate the utility of our methodology. Specifically, we describe the BMC^3 system used for our study, describe the hypotheses we formulated to establish the capabilities of our method, and describe the procedure used to test each hypothesis.

4.1 THE BMC^3 SYSTEM MODELED

The BMC^3 system used for our analysis controls the operation of a midcourse ballistic missile defense system. The system is partitioned along regional lines of authority, with overall operations within any one region under the control of a regional battle manager. Each region is divided into two sectors, and each sector has its own organic sensors. The weapons are located at (ground-based) Launch Control Complexes, and are under the direct control of the regional battle manager.

The BMC^3 system performs 13 major functions:

1. Multi-Sensor Object Count: determines which objects were seen by the sensors in each sector sensors during the last surveillance interval, and assigns every object detected to a sector;
2. Multi-Sensor Correlation: correlates the detection reports from the sector sensors, and assembles and maintains sector tracks on objects;
3. Multi-Sector Data Fusion: fuses sector tracks to assemble/maintain regional tracks on objects;
4. Threat Assessment: classifies the lethality of an object, determines whether the object is eligible for intercept, and places objects eligible for intercept into an assignment queue
5. Battle Planning: allocates weapons for select time intervals and selects the engagement

strategy (e.g., shoot-shoot, shoot-look-shoot, etc.);

6. Engagement Planning: assigns interceptors to those targets present in the queue awaiting assignment, and instructs the weapon Launch Control Complexes to conduct engagements;

7. Weapon Launch: schedules interceptor launch times per the allocation provided by the Battle Planning function, and executes the launch instructions received from the Engagement Planning function;

8. Sector Guidance Assignment: assigns the responsibility for interceptor guidance updates to sectors;

9. Sensor Guidance and Designation: provides guidance updates to interceptors in flight;

10. Sector Kill Observation Assignment: assigns the responsibility for kill observation to sectors

11. Observation Assignment: assigns the responsibility for kill observation to a sector sensor;

12. Sensor Kill Observation: observes a completed interceptor/target engagement; and

13. Battle Assessment: accumulates and assesses statistics on engagement outcomes.

The nominal processing sequence conducted by this Battle Management system is illustrated as a network of queues in Figure 3. Two important characteristics of this system are shown here. First, most of the functions have a finite processing throughput rate that is determined by the allocation of processing capacity provided by the battle manager responsible for that function. Second, RVs declared to be alive by the Kill Observation function are reported back to the Threat Assessment function for subsequent lethality assessment and weapon assignment.

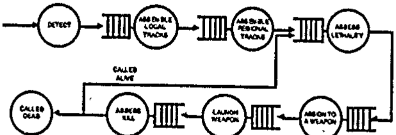


Figure 3. The Nominal Function Processing Sequence for the BMC[3] System Studied

The state space used for our analyses comprise the states that can be reached by any re-entry vehicle (RV) as it is engaged by the defense. The operations of this mid-course system are captured in a time-varying Markov model comprising 68 RV states. We direct the reader to [13] for further details on the state definitions.

4.2 HYPOTHESIS 1

The purpose of our first hypothesis is to determine whether the relation between function performance and system effectiveness expressed in Eq. 2-11 is accurate. Our hypothesis is that this relationship is accurate in the neighborhood of an arbitrary nominal operating condition.

The resolution approach for this hypothesis is to predict system effectiveness for a function at an operating condition that is slightly different from a nominal condition. We assume we know perfectly the threat state transition probabilities for the off-nominal condition, but do not know the cost of leakage at that off-nominal condition. We approximate the latter using the average cost statistics computed at the nominal condition.

Consider both a nominal and an off-nominal operating condition for a BMC^3 system. From Eq. 2-1, the system effectiveness measure under the nominal condition, J^n, is expressed in terms of the terminal RV state distribution as:

$$J^n = \sum_{i \in S} \rho_T^n(i)\, g_T^n(i) \qquad (4\text{-}1)$$

where $g_T^n(i)$ is the average cost incurred by the defense for every RV appearing in state i under the nominal condition; this statistic is computed from a Monte-Carlo simulation of the BMC^3 system operating at the nominal condition. The equation for the state dynamics under this nominal condition is given by:

$$\rho_{t+1}^n = P_t(\rho_t^n, F_t^n)\, \rho_t^n \qquad (4\text{-}2)$$

Similarly, the system effectiveness measure under the off-nominal condition, J_O, is expressed as:

$$J^o = \sum_{i \in S} \rho_T^o(i)\, g_T^o(i) \qquad (4\text{-}3)$$

while the state dynamic model is given by:

$$\rho_{t+1}^o = P_t(\rho_t^o, F_t^o)\, \rho_t^o \qquad (4\text{-}4)$$

The relationship expressed in Eq. 2-11 implies that the following estimate of the effectiveness measure under the off-nominal condition

$$\hat{J}_I = \sum_{i \in S} \overset{o}{\rho}_T(i)\, g_T^n(i) \tag{4-5}$$

should be accurate for small deviations around the nominal operating condition. In words, we compute an estimate of the effectiveness measure under the off-nominal condition using the threat state transition probabilities observed for the off-nominal condition, together with the cost of leakage observed for the nominal condition.

A confirmation of our hypothesis has two implications. First, we can conclude that the relationship between function performance and system effectiveness is sufficiently accurate to predict changes in effectiveness within the neighborhood of a nominal operating condition. Second, we can conclude that the pseudo-costate statistics derived from a nominal operating condition accurately measure the change in system effectiveness if the BMC^3 system can effect favorable changes in state with higher likelihood than it is achieving under the nominal condition

4.3 HYPOTHESIS II

The purpose of our second hypothesis is to determine whether the Markov approximation used in Equation 2-12 is valid. Our hypothesis is that the Markov approximation is valid in the neighborhood of an arbitrary nominal operating condition.

The resolution approach for this hypothesis is similar to that used for Hypothesis I, with one important difference: we no longer assume we know perfectly the entire threat state transition probability matrices at the off-nominal condition. We assume instead that we have perfect knowledge of the off-nominal transition probabilities for just those states that are directly affected by the function under study, and know only the nominal transition probabilities for all other states.

To make this clear, consider a function f that can act upon objects that appear in states i_1 and i_2. A momentary change in the operating condition for this function at some time interval will directly affect the state transition probabilities for those two states at that time, and will indirectly affect the transition probabilities for other states in the future. We ignore the latter effects by constructing hybrid transition probability matrices for every time period that incorporate the nominal transition probabilities in all columns with the exception of i_1 and i_2, and incorporate the off-nominal transition probabilities for columns i_1, and i_2. Figure 4 depicts the construction of a hybrid matrix.

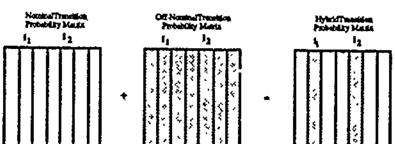


Figure 4. The Construction of a Hybrid Transition Probability Matrix

If we let $\overset{+}{\rho}_T(i)$ denote the state probability distribution generated at the end of the battle through the use of the hybrid transition probability matrices in Equation 4-3, then our estimate of the off-nominal system effectiveness measure under this hypothesis is given by:

$$\hat{J}_{II} = \sum_{i \in S} \overset{+}{\rho}_T(i)\, g_T^n(i) \tag{4-6}$$

A confirmation of our hypothesis will lead us to conclude that the SED and the USED measures, which are derived using the Markov approximation, provide an accurate estimate of the contribution of a function to the system's effectiveness during a conflict.

4.4 IMPLEMENTATION OF THE ANALYSES

The analyses were conducted using statistics collected from a Monte-Carlo simulation of the BMD system described earlier in this section. The simulated battle involves 650 re-entry vehicles (RVs) and 350 decoys. The system effectiveness measure is the total value of U.S. targets destroyed by the RVs. This effectiveness measure is nonlinear in the number of RVs leaking through the system when multiple RVs can be targeted against the same defended asset, since the probability that the asset survives the attack is the product of the probabilities that it will survive each RV. There is no handover of track from the boost-phase battle to the midcourse battle; all RVs and decoys in the simulation are initially undetected. The simulation provides data on state transitions, leakage, and asset value destroyed for a state equivalent model of the system.

We tested Hypotheses I and II using statistics collected from simulations of this BMD system for multiple values of a key parameter in the Threat Assessment function. The parameter selected was the lethality threshold that is used to distinguish lethal objects from non-lethal objects; this parameter can be assigned any value between 0 and 1.0. We varied this parameter from 0.2 to 0.8 in increments of 0.1, simulating the BMD system for 30 replications at each parameter setting. State transition data was collected every 5 seconds of engagement time for 400 time intervals at each replication, yielding 400 state transition probability matrices for each parameter setting.

We used both a variable nominal condition and a fixed nominal condition for these hypotheses. Under the former, we set the nominal value of the lethality threshold to 0.8, predict the value of the system effectiveness measure at the next lower threshold setting (0.7), re-define the nominal condition to be the value of the threshold at that next lower setting, and repeat the process until we have examined the entire range of parameter values under consideration. The purpose here is to determine the accuracy of the predictions within a local neighborhood of a nominal point.

Under the fixed nominal condition, we simply set the nominal value of the lethality threshold to a fixed value (0.4 in our experiments) and estimate the value of the system effectiveness measure at all other threshold values. The purpose of this approach is to determine whether the hypotheses are valid for relatively large deviations away from a nominal operating condition.

SECTION 5. COMPUTATION RESULTS

5.1 INTRODUCTION

In this section we present the results of the analyses we conducted to demonstrate the efficacy of our performance measures method. We begin with the computational results developed to accept or reject the two hypotheses we presented in the previous section. We then present the results of sensitivity analyses we conducted for the principle functions performed by this BMD system.

5.2 COMPUTATIONAL RESULTS FOR HYPOTHESIS I

The computational results from our first set of experiments confirm Hypothesis I. Figure 5 compares the value of the effectiveness measure predicted by our performance measure methodology to the actual value for the variable nominal threshold condition. The extremely close agreement between the predicted and actual values confirms our hypothesis that Eq. 2-11 accurately captures the relation between function performance and system effectiveness. The implication of this finding is that the pseudo-costate derived for a particular state at a particular time accurately indicates the additional effectiveness that can be gained by the system if it is better able to effect favorable state transitions from that state.

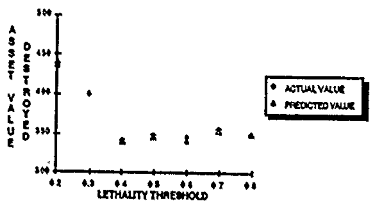


Figure 5. Effectiveness Predicted by the Performance Measures Methodology for the Hypothesis I Experiments (Variable Nominal Condition)

The computational results from this first set of experiments also suggest a finding that is stronger than the one implied in Hypothesis I, namely that the linear approximation is valid for relatively large excursions away from a nominal operating condition. The evidence of this is found in Figure 6 where we use a fixed nominal lethality threshold. All predictions fall within one standard deviation from the actual values.

5.3 COMPUTATIONAL RESULTS FOR HYPOTHESIS II

The computational results from our second set of experiments confirm Hypothesis II, namely that the Markov approximation is accurate in the neighborhood of an arbitrary nominal operating condition. Figure 7 compares the value of the effectiveness measure predicted by our performance measure methodology to the actual value for a variable nominal condition. While the predictions yielded by this experiment are relatively accurate on the whole, they are decidedly inferior to the quality obtained in the previous experiment, indicating that the Markov approximation does introduce noticeable modeling errors. Despite the prediction errors observed, all predictions fall within 1.5 standard deviations of the actual values, indicating that the hypothesis can be accepted at a significance level of $p = 0.10$. The confirmation of Hypothesis II implies that the System Effectiveness Derivative for a function accurately indicates the sensitivity of system effectiveness to a change in the performance of that function.

Figure 6. Effectiveness Predicted by the Performance Measures Methodology for the Hypothesis I Experiments (Fixed Nominal Condition = 0.4)

The computational results for this set of experiments do not support the stronger suggestion that the Markov approximation is valid for relatively large excursions away from a nominal operating condition. The evidence for this conclusion is found in Figure 8 where we use a fixed nominal condition (0.4) to predict the value of the effectiveness measure for all other values of the lethality threshold. The predictions in the immediate neighborhood of the nominal condition fall well within 1 standard deviation of the actuals, but the predictions for the higher values of the lethality threshold tend to fall more than 3 standard deviations away from the actual values.

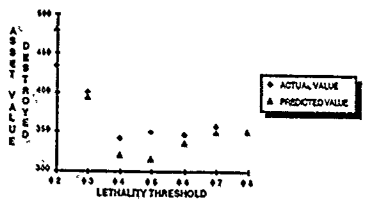


Figure 7 Effectiveness Predicted by the Performance Measures Methodology for the Hypothesis II Experiments (Variable Nominal Condition)

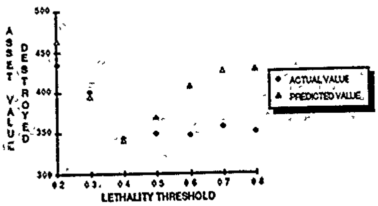


Figure 8. Effectiveness Predicted by the Performance Measures Methodology for the Hypothesis II Experiments (Fixed Nominal Condition = 0.4)

5.4 COMPUTATIONAL RESULTS FOR REPRESENTATIVE SENSITIVITY ANALYSES

We present in Figure 9 time-profiles of the SEDs for the principle functions performed by this midcourse BMD system; these derivatives were computed with the value of the lethality threshold equal to 0.4. The reader will recall from our discussion in Section 3 that the SED for a function at a particular time measures the sensitivity of the system effectiveness measure to the probability that the function can advance the system to desirable intermediate states. Under the definition of state used for this study, the SEDs in Figure 9 represent sensitivity to the probability that the function in question can advance the threats it is processing along desirable state transitions. Thus, the SED for the Detection function measures the sensitivity of the system effectiveness measure to a change in the probability of detection within a single 5 second time interval. Similarly, the SED for the Kill Observation function measures sensitivity with respect to a change in the probability that Kill Observation will complete the processing of an outstanding kill observation request within a single 5 second interval. The profile of the SED for any one function over time indicates the periods in the battle when that function exerts its largest and smallest contributions to system effectiveness, while a comparison of SED profiles for several functions indicates the major contributors to system effectiveness during any one time interval.

Several important observations emerge from Figure 9. Detection is the major contributor to system effectiveness at the beginning of the battle when the majority of objects have not yet been detected. Similarly, the Engagement Planning and Kill Observation functions are the major contributors to system effectiveness during the middle of the battle when weapons must be committed to targets that have regional tracks and when intercept outcomes must be assessed. On the whole, this BMD system is far more sensitive to a change in the rate of completion of the Kill Observation function than it is to a change in the rate of completion of any other function. An analysis of additional statistics collected by the simulator indicated that 15-25% of the RVs assigned to weapons were not hit on the first shot, thereby placing a premium on the timely execution of Kill Observation to provide the defense sufficient time to take a second shot against an RV that has survived a first shot.

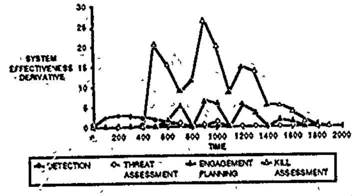


Figure 9. System Effectiveness Derivative Profiles for Select BMC^3 Functions (Lethality Threshold = 0.4)

Another important observation that emerges from Figure 9 is the pronounced oscillation in SED values for select functions over time. In the derivation of the SED as an analysis tool we noted that oscillations often signify the occurrence of processing bottlenecks elsewhere in the system. An analysis of the statistics collected by the simulator revealed that launch requests from the Engagement Planning function arrived at the the Launch Control Complexes (LCCs) faster than they could be processed beginning at 800, 1100, and 1400 seconds; the LCCs became the predominant bottlenecks in the system beginning at those time periods. Naturally, the BMD system is not sensitive to small variations in the processing throughput of the Engagement Planning function when the LCCs have more requests than they could handle, thereby yielding SED values of 0 for that function during those time periods. Threat Assessment precedes Engagement Planning in the overall processing sequence, and its SEDs declined during those time periods as well. Its SEDs never quite reached zero, however, since it is not directly affected by processing queues at the LCCs. Similarly, Kill Observation ultimately provides feedback to the Threat Assessment function, and therefore precedes Engagement Planning in the feedback processing sequence; its SEDs also declined during time intervals indicated. In summary, the effect of processing bottlenecks at the LCCs propagated with diminishing effect back up the processing chain.

SECTION 6. SUMMARY AND CONCLUSIONS

6.1 SUMMARY

We have presented in this paper a method to relate the performance achieved by Battle Management function to the consequent effectiveness achieved by the larger military system served by those functions. The method encompasses a quantitative relationship between function performance and system effectiveness and analysis methods for computing sensitivity from these relationships The relationship captures the operation of an arbitrary BMC^3 system as a time-varying finite-state Markov chain.

We also presented several analysis methods based on the pseudo-costates introduced in Section 2 and on the System Effectiveness Derivative (SED) introduced in Section 3. These two measures offer interpretations that permit the identification of time epoches during a conflict when the BMC^3 system under study is most sensitive to both favorable and unfavorable changes in state, and to processing bottlenecks. These measures therefore support diagnostic analysis of BMC^3 systems, and should prove useful in the development of more efficient Battle Management systems.

6.2 FUTURE MODELING DIRECTIONS

While the analysis methods described here localize the major contributions to system effectiveness to the function level, complementary modeling methods are needed to reveal how a function achieves its performance. Specifically, methods are needed to evaluate the partial derivative term $\frac{\partial \alpha_t(i)}{\partial f_t}$, where $\alpha_t(i)$ denotes the probability that a threat in state i at time t will advance to a favorable state, and f_t denotes the operating condition of function f at time t. We noted in Section 3 that the expression for the relation between function performance and system effectiveness (Eqs. 2-11 and 2-12) can be difficult to evaluate in practice since it presumes we have sufficient knowledge of the system 'plant' to compute this derivative. We frankly doubt that analytic expressions can be found in practice for all but the simplest functions, but hold out the hope that empirically derived regression models may suffice for most applications.

REFERENCES

[1] Wohl, J.G., D. Gootkind, and H. D'Angelo, "Measures of Merit for Command and Control," MITRE Technical Report, MTR-8217, January 1981.

[2] Moore, K., P. Vail, and R. Tenney, "Systematic Evaluation of Command and Control Systems," TR-284, ALPHATECH, Inc., Burlington, MA, April 1986.

[3] Wohl, J., and R. Tenney, "Integrated Analysis Techniques for Command, Control, and Communications Systems; Volume I: Methodology," TR-293, ALPHATECH, Inc., Burlington, MA, November 1986.

[4] Sweet, R., et al. (Eds), Proceedings of the Command and Control Evaluation Workshop, MORS C2 MOE Workshop, Naval Postgraduate School, January, 1985.

[5] Sweet, R., An Evolving C2 Evaluation Tool - MCES Theory," Proceedings of the Ninth MIT/ONR Workshop on C3 Systems The Laboratory for Information and Decision Systems, LIDS-R-1624, M.I.T., Cambridge, December, 1986, pp 183-187.

[6] Mensh, D., and R. Fox, "MCES - A Tool for Integrating Technology into Navy Battle Groups," Proceedings of the Ninth MIT/ONR Workshop on C3 Systems The Laboratory for Information and Decision Systems, LIDS-R-1624, M.I.T., Cambridge, December, 1986, pp 229-236.

[7] Stone, I., "Evaluations of Austere Command Centers using the Modular C2 Evaluation Structure," Proceedings of the 1987 Joint Director of Laboratories Command and Control Research Symposium, June, 1987, pp 313-318.

[8] Athans, M., et al., "C3 Theory: Proceedings of the Fourth MIT/ONR Workshop on Distributed Information and Decision Systems Motivated by C3 Problems Volume IV," The Laboratory for Information and Decision Systems, LIDS-R-1159, M.I.T., Cambridge, October, 1981.

[9] Metersky, M., "A C2 Process and an Approach to Design and Evaluation," Proceedings of the Seventh MIT/ONR Workshop on C3 Systems The Laboratory for Information and Decision Systems, LIDS-R-1437, M.I.T., Cambridge, December, 1984, pp 11-22.

[10] Levis, A., "The Quest for a C3 Theory: Dreams and Realities," Proceedings of the 1987 Joint Director of Laboratories Command and Control Research Symposium, June, 1987, pp 7-12.

[11] Mayk, I., and I. Rubin, "Paradigms for Understanding C3, Anyone?," Proceedings of the 1987 Joint Director of Laboratories Command and Control Research Symposium, June, 1987, pp 93-104.

[12] Bertsekas, D. P., "Notes on Nonlinear Programming and Discrete-Time Optimal Control," The Laboratory for Information and Decision Systems, LIDS-R-919, M.I.T., Cambridge, August, 1981.

[13] Shaw, J.J., P.E. Coughlin, and W.S. Powell, "A Methodology for Relating Battle Management Function Performance to Military System Effectiveness," ALPHATECH, Inc., Burlington, MA, June, 1988.

Database needs for modeling C³I system performance

Keith Johnson[†], Yang-Wei Wang[†], John Mellen[†], Way Kuo[‡*]

[†] Department of Industrial Engineering
[‡] Department of Electrical and Computer Engineering
Iowa State University, Ames, Iowa 50011

ABSTRACT

A generalized concept of system evaluation and an automated C³I evaluation system, OSAM, are described. The goal of the evaluation concept is to permit comparison of different C³I systems with similar missions or goals. OSAM is designed to facilitate the comparison of alternative C³I implementations through the use of several interface programs, local databases and modularity. The concept of *implementation independence* unites these two efforts. This paper discusses implementation independence, the OSAM system itself and relevant databases.

INTRODUCTION

An appropriate *Command, Control, Communications and Information* (C³I) system can greatly enhance the effectiveness of military units in the field [1]. A designer may choose from many options and may be concerned with performance on many levels. Hence, C³I system development is a very complex task which could be greatly facilitated by means of an automated C³I analysis tool [2]. Two difficulties facing automation are:

1. A unified approach to C³I evaluation that will allow the comparison of dissimilar systems with equivalent missions is needed.
2. The extensive body of information required to evaluate C³I systems is not currently available in a standardized, computer-usable form.

A consistent approach is important because if two or more C³I systems cannot be judged against each other on the basis on the same *Measures of Performance* (MOPs), *Measures of Effectiveness* (MOEs), and other criteria, it is not possible to compare one to the other. If information is not in a standardized, computer usable form it will have to be entered manually for each evaluation. This not only increases the time to prepare a model, but also introduces a greater chance for error.

A four phase C³I system performance evaluation approach, the *Overall System Analysis Model* (OSAM), is being developed around the *General Simulation System* (GSS) [3] to overcome these difficulties. A key element of this new approach is the separation of the environmental factors and the mission into an *Implementation Independent Scenario* (IIS), distinct from the procedures, equipment and other implementation details of the simulation. A second major concern of this project is the standardization and codification of information required for C³I performance evaluation.

The the concept of Implementation Independence, the guiding principle for the factorization process mentioned above is discussed first. Then, the OSAM four step procedure is described after which relevant databases and files are reviewed.

IMPLEMENTATION INDEPENDENCE

The basic concept of implementation independence is to modularize the simulation model so that 1) subsequent models may be more rapidly developed and 2) the performance of different systems may be compared. The implementation refers to the equipment, subsystems, procedures, and protocols that comprise the particular C³I system being evaluated. The IIS is the environment, set of external stresses, *command and control* (C²) expectations and mission, which specifies the test to which the implementation is subjected. This factorization allows two or more different implementations to be subjected to the same test.

The concept is rather straightforward, but requires care in practice. The distinction between the IIS and the implementation is not completely clear. For example, shall very-high level error checking (e.g. by a human commander) be considered part of this IIS or the implementation? Also, new systems may present new options which were not previously tested, and some C² functions, such as tar-

*To receive correspondence.

get tracking, are very closely tied to the equipment they utilize. Thus a change in an implementation may dictate a change in the IIS, but such changes must be made with care to retain compatibility with earlier simulations.

The larger classifications are also factored further to facilitate model development. For example, the IIS actually consists of a setting (physical environment), C^3 structure, external stress (e.g. enemy troop movements and electronic warfare), and mission. One may compare C^3I performance in two simulations, one with an offensive and one with a defensive mission. The modularity of the IIS allows the second model to be derived quickly from the first. Likewise, the implementation is modularized with respect to C^3 subsystems, C^3 units and military units. This not only permits rapid design of similar models, but also simplifies modeling due to the close correspondence between actual systems and models, and allows one to obtain MOPs and MOEs for these modular units.

To assure long-term utility and standardization, the principles of modular classification are well-defined. Also, the GSS environment supports this type of factoring very well, through its hierarchical structure and modularity.

The IIS specifies processes that must occur among a given set of entities. These are defined as very-high level GSS modules, referencing yet-to-be-defined submodules. For example, a module may specify that an average of 10 messages per hour of a given size are to pass from Company A to Company B. The corresponding module in the IIS would use a process implying a communication between Company A and Company B, but the actual behavior of this communication process would not be defined.

General system modules actually predict the behavior of C^3I subsystems in a functional manner. For example, for a communications system, given certain environmental conditions, input load, and degradation, a message can be expected to be transmitted with a given delay distribution. The analyst builds a specific model, which specifies performance parameters of a general subsystem model. Further performance characteristics are derived from the stress information in the IIS. Then the specific model will predict the performance of the communication between Company A and Company B.

THE OVERALL SYSTEM ANALYSIS MODEL

The *Overall System Analysis Model* (OSAM), depicted in Figure 1, consists of four major subsystems. Each is capable of operating independently of the others, communicating with them through files and databases. The four subsystems of OSAM are:

1. Scenario Translation
2. C^3I Model Development and Selection
3. C^3I System Simulation
4. Post-Simulation Analysis

Scenario Translation is facilitated by the Scenario Modeler, a FORTRAN program which helps to produce an implementation independent representation of a given scenario, the IIS, and a list of entities required for any C^3I implementation, the Directory. During the scenario translation process, communication requirements, mapping information, and scenario information are required. The Scenario Modeler extracts required information from various databases to eliminate unnecessary and redundant information.

The C^3I Model Development and Selection subsystem is used to develop low-level GSS representations of C^3I systems and subsystems. These models are built using performance data and structure implicit in C^3I component specifications and evaluations. Since C^3I subsystems are essentially modular in nature [1], the C^3I modeling effort is also modular. As new C^3I subsystems and technologies are developed, it will be necessary to develop appropriate corresponding GSS models. C^3I Model Development has a mini-simulation capability to evaluate and check new models for higher-level model development, as well as model editing capabilities, to permit generalization of specific models. Necessary inputs for C^3I Model Development and Selection include management and protocol specifications related to communication requirements, mapping information, equipment operating and performance parameters, and the Directory.

The central component of the C^3I System Simulation, is the *Simulation Control Program* (SCP). Its operation may be modified by the analyst to control the granularity, scope and focus of a particular simulation run. It essentially orchestrates the files generated by the Scenario Translation and C^3I Model Development and Selection subsystems. Since established C^3I models are maintained in the GSS Model Database, they may be referenced by the SCP during the simulation, without actually copying their full descriptions into the model file. Utilities are modules which collect data and note states as necessary to permit eventual MOP and MOE computation.

Finally, Post-Simulation Analysis has four basic purposes: 1) translate the output into human understandable form, 2) generate data files compatible with other parts of OSAM as well as other systems interfaced to it, 3) complete the computation of some MOPs and MOEs, and 4) produce trace and debug information to facilitate the development of models. Since this final step need only interface with the C^3I System Simulation output, which consists of text files, it is therefore loosely coupled to the rest of OSAM.

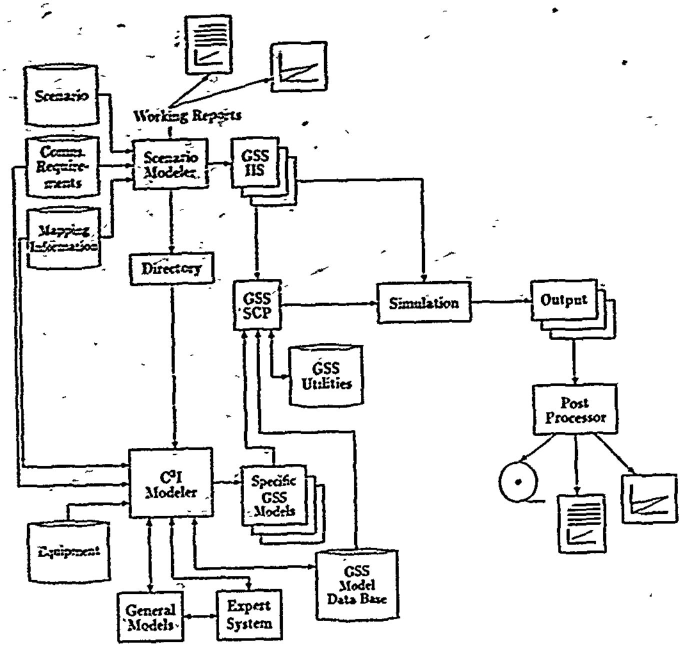


Figure 1: Overall System Analysis Model For C^3I System Performance Evaluation

RELEVANT DATABASES

Databases used by OSAM can be classified into the following categories.

1. *Input* – Those files and databases that exist independent of OSAM, which contain source information needed for C^3I analysis.

2. *Internal* – Those files or databases that are created and used during the four step C^3I system performance evaluation procedure.

3. *Output* – Files generated by OSAM which are to be used by other automated systems as well as analysts

The following information is supplied by external input to OSAM. Since this type of information is common among the C^3 community, only a cursory explanation of each is included in this discussion.

- Scenario Information – The environment, including hostile action such as jamming and other forms of electronic warfare; C^2 structure and particular military units involved; troop movements and involvement.

- Communication Requirements – Needline information between military entities including priority, message length, and transmission rate.

- Mapping Information – Terrain characteristics; elevation; a method of uniquely identifying the location of every entity within the scenario area.

- Management and Protocol Specifications – Communication hierarchy; priority schemes; routing algorithms used.

- Equipment Information – Operating parameters of selected equipment; any available performance information, possibly from field testing or manufacturer reports.

Internal databases merit a more detailed discussion. They can be grouped into the following four areas:

- Implementation Independent Scenarios
- Established Model Database
- Specific Simulation Information
- Past Events Results

The IIS combined with Specific Simulation Information provides the setting for a particular performance evaluation. The *Established Model Database* (EMD) will be initially limited, but will grow as more systems are modeled. The Past Events Results are files generated by the Post Processor from output files. Each of these types of files is discussed below.

Implementation Independent Scenarios

The IIS is a set of data files, resources, processes and models that represent the essence of the scenario, independent of the C³I implementation. Thus, needlines are represented, but the C³I subsystem used to effect those needlines are not. The goal here is to allow a single IIS to serve as input for a number of specific implementations so that their performance may be compared, assuming similar technologies of implementation. This is essentially a high-level GSS representation of the simulations model. The lower levels of the GSS model will be developed independently to represent the particular C³I implementation to be tested.

Since the IIS is a high-level representation, some sort of reference will be needed to assure that entities referred to by the IIS are modeled at the lower level with an interpretation consistent with that of the IIS. The list of entities involved is such a reference, consisting of two parts. The first is a computer-readable checklist which will be used by the C³I Modeler to verify that all entities referenced in the IIS are not only represented in the lower level models, but also represented in a technically consistent manner. That is, processes are modeled as processes, resources as resources, and so on. The second is a printout, which may be used by the analyst, containing brief descriptions of the interpretation of each entity in addition to the technical information mentioned above.

Finally, since the IIS is a computer-oriented set of files, the analyst is likely to have difficulty interpreting it. Because the analyst may wish interim representations of a developing scenario to provide points of reference, the Scenario Modeler will produce working reports, summaries, listings, tables and maps representing the IIS at any stage of development.

Established Model Database

The *Established Model Database* (EMD), is a set of previously verified and validated C³I subsystem models. A specific simulation is a low-level GSS model which accurately depicts the performance of a C³I subsystem. Some common examples are:

- Area Common User Systems
  - Mobile Subscriber Equipment (MSE)
- Data Distribution Systems
  - Joint Tactical Information Distribution System (JTIDS)
  - Enhanced Position Location Reporting System (EPLRS)
- Combat Network Radio Systems
  - Single Channel Ground/Airborne Radio System (SINCGARS)
  - Improved High Frequency Radio (IHFR)
  - Small Unit Radio (SUR)
- Satellite Communication Systems
  - Single Channel Objective Tactical Terminal (SCOTT)

Since these subsystems are used in a modular fashion to build C³I systems, it makes sense to build specific simulation models which represent them, providing these models are general enough to be used in any probable situation. This approach allows the analyst to build models more quickly and with fewer errors than if he or she started from scratch each time.

The challenge is to write these models so that their connection to the IIS is functional and not dependent on subsystem particulars. For example, if there were a need for a data distribution system in an IIS, it should be possible to use either the JTIDS or EPLRS model without modifying the IIS.

Specific Simulations

In some cases, unique models may be built using elementary GSS tools, but in most cases a specific instance of a general model will be used. The specific simulation is a reference to a general model together with specific parameters. For example, in a MSE system, specific parameters would include the number of subscribers and matrix topology.

Past Events Results

Past events results are output from past simulations which can be used as starting points for other simulations. For example, initial simulation could be done at the Brigade or Battalion level, then the output can be used as input for a simulation at the Division or Corps level.

CONCLUSION

In the past, comparing simulation and performance results of different C^3I models was equivalent to comparing apples and oranges. However, the concept of implementation independence allows a unified approach to C^3I analysis. Implementation independence also reduces the complexity and increases the efficiency of C^3I system modeling. Further efficiency is achieved through the judicious use of automated subsystems and specialized databases. This approach used in conjunction with a powerful simulation language, GSS, holds much promise for future modularized C^3I system design.

ACKNOWLEDGEMENT

The work resulting in this paper was supported by the U.S. Army Communications-Electronics Command and Fort Monmouth at Fort Monmouth, N.J., under the contract number DAAB07-86-A014.

REFERENCES

[1] Ricci, Fred J. and Daniel Schutzer. *U.S. Military Communications: A C^3I Force Multiplier.* Rockville, MD: Computer Science Press, Inc. 1986.

[2] Harris, C.J., and I White; Editors. *Advances In Command Control & Communication Systems.* London, United Kingdom: Peter Peregrinus Ltd. 1987.

[3] Prediction Systems, Incorporated. *The General Simulation System Version 10 User Manual.* Manasquan, NJ: Prediction Systems, Inc. 1983,1987.

AUTHORS

Keith Johnson received his BS degree in Industrial Engineering in 1986 from Iowa State University and expects to receive his MS degree in Industrial Engineering shortly. For the past year, he has been a member of a research team developing evaluation procedures for C^3I systems. He will be joining BDM International as an Associate Staff Member in Operational Test Modeling Methodologies.

Yang-Wei Wang was born in Taipei, Taiwan, ROC. He received his BS degree in Electrophysics from National Chiao-Tung University, Taiwan, and MS degree in Electrical Engineering from Iowa State University. Currently, he is a graduate research assistant and Ph.D. candidate in Computer Engineering at ISU. His research interests include distributed packet radio network, random access protocol, and network performance evaluation.

Mr. Wang is a member of IEEE, IEEE Computer Society, and the Honor Society of Phi Kappa Phi.

John Mullen received his BA in Mathematics from the University of Pennsylvania, Philadelphia, in 1968. During six years of service in the U.S. Navy he received extensive training in electronics, computer and communication technology. When he received a MS degree in Industrial Engineering from Iowa State University in 1984, he was awarded the ISU Engineering Research Excellence Award for his completed master's work, in 1984. Currently he is pursuing a Ph.D. in Industrial Engineering with a minor in Statistics at ISU. He is a member of IIE and the honor societies Phi Kappa Phi and Mu Sigma Rho.

Mr. Mullen is a research associate and instructor of Industrial Engineering at Iowa State University. Previous experience includes six years of industrial work in addition to the six years of naval service. He has published several papers on the use of the computer language Forth in operations research and industrial control problems. He currently serves as reviewer for the *Journal of Forth Application and Research.* His research interests include operations research, statistics and computer applications.

Dr. Way Kuo received his BS degree in 1972 in Nuclear Engineering from National Tsing-Hua University, Taiwan, ROC, then served in the Taiwan Marines from 1972 to 1974 with a distinguished service award granted by the Secretary of Navy. He received his MS degree in 1975 in Nuclear Engineering from the University of Cincinnati, an MS degree in 1977 in Industrial Engineering, the MS degree in 1980 in Statistics, and the PhD degree in 1980 in Engineering from Kansas State University.

Dr. Kuo is Professor of Industrial Engineering at Iowa State University. He was previously on the Technical Staff at AT&T Bell Laboratories, a visiting scientist at the Oak Ridge National Laboratory, and a lecturer at the NATO Advanced Study Institute. He has been the principal investigator on projects funded by the U.S. Army, the U.S. Department of Energy, the National Research Council, etc. Dr. Kuo co-authored the text *Optimization of Systems Reliability* published by Marcel Dekker in 1985 and served as a guest editor for the IEEE Journal on Selected Areas in Communications published in October 1986.

RULE-BASED COMBAT OUTCOME MODEL

Joseph P. Provenzano

Jet Propulsion Laboratory
California Institute of Technology

4800 Oak Grove Drive
Pasadena CA, 91109-8099
Section 363 MS 510-264
(818) 397-9991

ABSTRACT

Combat simulations have become essential to conducting many large-scale military exercises. However, it has proven to be very difficult to develop a simulation that has the desired degree of realism, which runs in real time on affordable computers, and which provides results that can be easily understood in postgame analysis. (The players in general do not have, nor desire to have, a detailed knowledge of the simulation.)

It is time to consider a completely new approach to modeling combat outcomes in exercise simulations. The standard approach is one of using mathematical models of combat which essentially translate all factors into attrition multipliers. The resulting attrition is the sole driver of the combat outcomes. This paper addresses the concept of using a rule-based expert system to determine the outcome in a more qualitative manner and focuses on a methodology for embedding the expert system into a full-scale military simulation.

1. OVERVIEW

The idea of using an expert system to determine combat outcomes is a result of contemplating past development efforts, analyzing new requirements, and making observations at the National Training Center, where expert controllers can often predict the outcomes by comparing the opposing plans.

The discussion which follows centers on two key software components. The first is a discrete event simulation referred to as "The Game," which uses conventional simulation techniques to model most aspects of the military operation; unit movement, logistics, air strikes, detailed attrition, etc. The other key software component is a rule-based expert system referred to as "COBRA," which is an acronym for Combat Outcome Based on Rules for Attrition. COBRA is used to determine the outcome of the ground combat engagements.

The work described herein was carried out by the Jet Propulsion Laboratory and was sponsored by the Joint Warfare Center through an agreement with the National Aeronautics and Space Administration, under contract NAS7 918.

The interaction between these two software modules is envisioned to operate in the following manner. The Game controls the simulation, moves units around the battlefield, and determines when a unit(s) has entered the combat zone of an enemy unit(s). The Game then invokes the COBRA module and provides the necessary data on these units. The predicted outcome of this combat situation is determined by the COBRA module, and the rules leading to the determination are recorded for future analysis. (Note that the COBRA module is finished and is not active again until the next call by the Game.) This projected outcome is accurate for describing the progress of the battle until a player (human or automated) intervenes, or until the end of the engagement. At least a few such interventions are expected in most cases, and the COBRA module will be recycled each time. The projection is given to the Game, which produces detailed attrition calculations and provides reports to the players. The attrition and the reports reflect the projection provided by the COBRA module.

One can think of this interaction as COBRA providing the Game with a high-level "script" for describing each combat engagement. The Game will determine the detailed attrition scale factors or whatever is necessary to produce results over time for all weapon systems. This will result in the given percent losses for both sides at the given time for the end of the battle.

2. GAME – COBRA PROCESSING FLOW

Figure 1 is a high-level diagram of the Game – COBRA Process Flow. More detail about this process is provided in the following paragraphs.

1. Start

This is simply the start of the diagram. It represents that two enemy units have come into contact and combat is likely.

2. Combat Decision

This box contains the software control logic for ground combat engagements. If it is a new combat situation (or something has happened which may change the outcome), COBRA is initiated (or recycled) as indicated by the upward arrow. If there has been no player or game intervention since the last COBRA cycle, the current COBRA guidelines are still valid and the detailed losses for this time interval are computed as indicated by the continuation arrow. (Arrival of new forces, changes of mission, and air strikes are examples of interventions which would cause COBRA to be recycled.) If the time for the end of this engagement has been reached, players are informed as indicated by the end arrow.

3. Identify Possible Outcomes

Whenever COBRA initiates processing, it focuses on only the possible outcomes. This focusing serves to limit the search space. The possible outcomes are a function of the "immediate" combat missions of the opposing forces; i. e., exactly what each side is trying to accomplish in this particular engagement. All the possible outcomes that are currently considered are shown in Figure 2.

4. Evaluate Combat Situation

This is the area where all the factors that can affect combat outcome are considered. These factors are assessed in the overall context of a METT-T (Mission, Enemy, Troops, Terrain, and Time) analysis For example, if an attacking unit does not have adequate knowledge of the defending force, COBRA will judge the attacker's intelligence of the enemy as poor. All such assessments will be done in a rule-based framework so that the reasons for the evaluations can be recorded in a manner easily understood by players who have little or no experience with the simulation.

5. Select Mini Combat Scenario

There will be predefined ways that each type of engagement can unfold. The choice depends on the evaluations made in Box 4. Further discussion of this concept and an example are provided in Section 4.

6. Select Combat Outcome

Given the set of possible outcomes, the METT-T evaluation, the mini-scenario selected, the opposing force ratio, and other factors, COBRA will select the combat outcome that makes the most sense in this situation. It will also record the reasons for the choice. More details of this concept are provided in Section 3.

7. Determine Time and Percent Losses

In order to further define the outcome, a reasonable time for the engagement will be determined as well as the overall percent losses for RED and BLUE. This is the end of the COBRA processing for the engagement, and control is passed back to the Game. The projections of the outcome, time of engagement, and percent losses will remain in effect until the end of the engagement or until COBRA is recycled as a result of some intervention.

8. Calculation of Losses

The specific losses for each of the weapon systems during the current combat cycle are computed here. A scale factor which ensures that the overall final losses will be equal to those specified in Box 7 is computed and used in the attrition calculations Note that the losses are "guaranteed to be reasonable" because we have a trace of the reasons back through a METT-T type evaluation.

9. Reports to Players

The reports that are sent to the players will provide information on the attrition that has been calculated and will also be tailored to conform to the mini-scenario and ultimate outcome selected. The idea is to make the reports as realistic as possible by using this type of information (Note that since the Game knows the outcome that will result if no intervention occurs, it will be relatively simple to convey this in the form of a lower level commander's estimate of the expected outcome Such estimates are not available in conventional simulations and will put realistic pressure on players to make timely decisions on the use of limited resources)

10. Player/Game Intervention

The processing in this box simulates the results of air strikes, the receipt of mission changes, etc. (Note that an air strike would cause the force ratio to change and may also affect the evaluated performance of some of the combat subsystems For example, an evaluation of attacker mobility may go from fair to good because of enemy suppression.)

3. IMMEDIATE COMBAT MISSIONS

The emphasis here is on the word immediate. COBRA is interested only in this particular combat situation and not the overall long-range missions of the units involved. In other words, COBRA needs to be given the immediate combat missions. The current list of these missions is as follows.

Attack - Take the Objective

Attack - Destroy the Enemy Unit

Attack - Fix the Enemy Unit

Attack - Bypass the Enemy Unit

Defend

Delay

Withdraw.

Figure 2 shows all possible combinations of these combat missions and the all of the combat outcomes that are currently considered possible for each case. Only one of the outcomes will be selected for each engagement, except where noted by the symbol &.

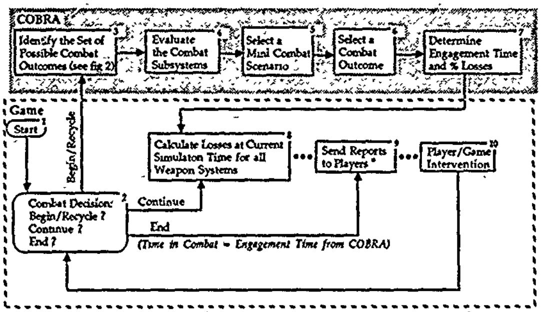
4. POSSIBLE MINI COMBAT SCENARIOS

In order to help determine the outputs mentioned above, it is envisioned that COBRA will map each combat engagement into a type of "mini combat scenario," which is chosen from a predetermined set. These possible mini combat scenarios will depend on the particular type of engagement, i e., on the Immediate Combat Missions of BLUE and RED.

Consider, for example, the case of Attack - Take the Objective vs. Defend We can identify essentially three different ways that this type of engagement could unfold and we define them as the mini combat scenarios for this case

1. Defeat in Detail

This is the way an attacker would like to see the attack develop. In this case the attacker is successful in bringing



* This module uses information about the Mini Combat Scenario and Combat Outcome to "paint a better picture."

(Start) Indicates two or more enemy units should begin combat.

⟶ Indicates the next logical step to be performed.

••• Indicates a time delay before action in the next box on the right of this symbol is performed.

Figure 1. Game - COBRA Processing Flow Diagram

RED

BLUE

	Attack - Take the Objective	Attack - Destroy Enemy	Attack - Fix Enemy	Attack - Bypass Enemy	Defend	Delay	Withdraw
Attack - Take the Objective	BLUE Takes It RED Takes It No One Takes	BLUE Takes It BLUE Dest	BLUE Takes It BLUE Incap	BLUE Takes It & RED Bypass	BLUE Takes It BLUE Repelled	BLUE Takes It RED Moves RED Overrun BLUE Repelled	BLUE Takes It RED Wdraws RED Overrun
Attack - Destroy Enemy	RED Dest RED Takes It	RED Dest BLUE Dest	RED Dest BLUE Incap	RED Dest RED Bypass	RED Dest BLUE Repelled	RED Dest RED Moves BLUE Repelled	RED Dest RED Wdraws
Attack - Fix Enemy	RED Incap RED Takes It	RED Incap BLUE Dest	Slow Attrit & No End Sched	RED Incap RED Bypass	Slow Attrit & No End Sched	Slow Attrit. RED Moves BLUE Repelled	RED Incap RED Wdraws
Attack - Bypass Enemy	BLUE Bypass & RED Takes It	BLUE Bypass BLUE Dest	BLUE Bypass BLUE Incap	BLUE Bypass RED Bypass	BLUE Bypass BLUE Repelled	BLUE Bypass RED Moves RED Overrun BLUE Repelled	BLUE Bypass RED Wdraws RED Overrun
Defend	RED Repelled RED Takes It	RED Repelled BLUE Dest	Slow Attrit & No End Sched	RED Repelled RED Bypass	Slow Attrit & No End Sched	Slow Attrit. RED Moves	RED Incap RED Wdraws
Delay	RED Repelled RED Takes It BLUE Moves BLUE Overrun	RED Repelled BLUE Moves BLUE Dest	RED Repelled Slow Attrit BLUE Moves	RED Repelled RED Bypass BLUE Moves BLUE Overrun	Slow Attrit. BLUE Moves	Slow Attrit. BLUE Moves RED Moves	RED Incap RED Wdraws
Withdraw	RED Takes It BLUE Wdraws BLUE Overrun	BLUE Wdraws BLUE Dest	BLUE Wdraws BLUE Incap	RED Bypass BLUE Wdraws BLUE Overrun	BLUE Wdraws BLUE Incap	BLUE Wdraws BLUE Incap	BLUE Wdraws & RED Wdraws

The rules will also determine engagement time and % losses for BLUE and RED to accompany each outcome.

Figure 2. Possible Combat Outcomes

essentially all of the attacking force on each of the defender's subordinate units – one at a time. Good attacker intelligence and little time for the defender to prepare the position are the types of conditions which would result in the selection of this alternative An attacker with a relatively small force ratio will have a good chance of success if this mini-scenario is selected.

2. Piecemeal Attack

This is the way a defender would like to see an attack develop. In this case the attacker loses command and control and is unable to coordinate the attack. The effect is to bring the attacker subordinates one at a time into the combined fields of fire of the defender forces.

3. One-on-One

Although there are many intermediate cases to the two extremes discussed above, we now consider only one of these intermediate cases. This is defined to be the "One-on-One" case and gives neither side an advantage due to the way the battle unfolds.

5. CURRENT EFFORTS

The ideas presented in this paper are continuously being reviewed and are subject to change as the effort continues. A small prototype has been developed in OPS5 and runs on a Macintosh II. It is being used to demonstrate the concept and also to stimulate ideas for improvement. One example of such an idea recently suggested and currently being investigated is that of using COBRA to select the type of attrition equations to be used rather than specifying the high-level percent losses. For example, the square law, the exponential law and the linear law*, which are three types of attrition equations resulting from three different mathematical representations of the ways that an engagement can unfold, correspond roughly to the three mini-scenarios mentioned above. Using this concept, COBRA would dynamically select the "law" that should be used to simulate the attrition for each phase of the various engagements and record the reasons for these selections.

6. CONCLUDING REMARKS

The Rule-Based approach for determining combat outcome discussed in this paper offers the potential to improve the simulations used for military training and exercises over that achievable using the traditional simulation techniques. These potential advantages include:

1. Handling a broader spectrum of cases;
2. Allowing users to have less knowledge of model details;
3. Considering more than attrition;
4. Assuring reasonable results;
5. Providing mission assessments; and
6. Simplifying postgame analysis.

The challenge lies in actually producing a system which lives up to this potential. There will always be those who will question the output. However, the intent is to provide the reasons for the output and, therefore, the rules will be visible and can be changed when determined to be unrealistic.

* See Reference 2.

ACKNOWLEDGMENTS

The author would like to thank our sponsor, the Joint Warfare Center at Hurlburt Field, Florida, for their support of this effort to discover new ways to develop better training and exercise simulations. The Battle Command Training Program at Ft. Leavenworth, Kansas, has also been very helpful in the support of this effort.

REFERENCES

1. R. W. Davies, *Training Soldiers with Exercises Driven by the Computer*, Internal JPL Memorandum - July 1987.
2. J. J. Schneider, *The Exponential Decay of Armies in Battle*, U. S. Army Combined Arms Operations Research Activity, Fort Leavenworth, KS 66027 - June 1984.

CONTINGENCY PLANNING IN HEADQUARTERS

Ann Needalman
Elliot E. Entin
Dianne Mikaelian
Robert Tenney

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

ABSTRACT

A military Headquarters can be viewed as a multidecisionmakers team supervising a highly dynamic, complex, and uncertain environment. The effects of contingency planning and battle intensity on headquarters effectiveness were investigated through experimental study. Two hypotheses were tested: 1) A plan with multiple options or alternatives is superior to a plan with a single option or alternative; and 2) The value of a multiple-option plan over a single-option plan will increase as the intensity of the battle increases. Subjects played four systematically varied presentations of a wargame within the context of the JANUS Simulation. The measurement of command and control effectiveness was performed through the analysis of two components: the outcome of the wargame and the process by which the outcomes were achieved. Outcome data consisted of event driven data of the location and strength of all units at designated periods throughout the game. These data were used to calculate various measures of effectiveness (MOEs) related to movement and attrition. Process data, collected by trained observers, consisted of frequency counts of discrete events related to communication within the headquarters and number and types of orders given. Observers also made a series of subjective ratings of the overall quality of activity within headquarters. Results confirmed the first hypothesis: headquarters effectiveness, as measured by battle outcome, was higher when a multiple-option plan was used. The second hypothesis was not confirmed: the value of a multiple-option plan did not increase as the intensity of battle increased. Results indicated an unexpected interaction between plan and battle intensity showing that while a multiple-option plan was better than a single-option plan in the low battle intensity condition, the opposite was true in the high battle intensity condition. Consideration was given to findings related particularly to process measures assessing information requests in an effort to understand this counterintuitive effect. It was noted that headquarters that did best in a particular experimental condition were also the headquarters that requested the most information.

INTRODUCTION

The Defense Communications Agency (DCA) has supported research on headquarters effectiveness since 1982. A Headquarters Effectiveness Assessment Tool (Defense Systems, Inc., 1984) comprised of 135 headquarters performance measures was developed for DCA, and a substantial amount of empirical data was collected and analyzed using portions of this tool. The Headquarters Effectiveness Assessment Tool (HEAT) measures are derived from a theory of headquarters decisionmaking that is based on classical adaptive feedback control theory. The decision process is hypothesized to operate in a cycle: gathering information, assessing the situation, generating decision options, predicting the outcome of each option, selecting a response, executing the response, and monitoring its outcome, i.e., back to information gathering.

The purpose of this experiment was to determine the impact of multiple-option contingency planning on battle outcome, and how that impact varies with battle intensity. One of the early premises of HEAT theory was that uncertainty about enemy intentions and capabilities (the so-called "fog of war") forces good headquarters commanders and staffs to consider more than one possible enemy course of action, and to formulate several responses to each, before committing to a particular operation. Often called "contingency planning," one would expect that this approach would yield plans and directives which are more robust to enemy actions than a "brittle" plan where only one enemy course of action, and one plausible response, is considered.

METHOD

Subjects

Nine officers drawn from a course in military physics at the Naval Postgraduate School, along with three line officers from the 101st Airborne Division, served as subjects for the experiment. The subjects were randomly divided into three teams. Each team consisted of three students and one 101st officer who assumed the role of Brigade Commander of Blue Forces. The team and command structure was preserved throughout the experiment.

Apparatus and Wargame Scenario

The JANUS simulation presents two closely related versions of a wargame: the L version developed by the Lawrence Livermore Laboratory, and the T version developed by the Army Training and Doctrine Command Wargaming Laboratory. Each version of the simulation was hosted by a micro-VAX computer, and connected to each computer was a pair of graphics terminals equipped with mouse control. The JANUS facility is located at the TRADOC Wargaming Laboratory, Naval Postgraduate School in Monterey.

In the JANUS simulation, all movement of forces on the screen is accomplished through use of a mouse. Players have individual control of tokens representing one or more weapon systems. The simulation is oriented about "line of sight" and "direct kills" of weapon systems and not Lanchester computation to assess attrition as in other war games.

The wargame scenario depicts a conflict in Iran involving U.S. and Soviet forces. The background for the wargame is part of a larger scenario set in Iran, under development for National Defense University. In this scenario, rebel Iranian forces are threatening to seize Bandar Abbas, on the Strait of Hormuz in the Persian Gulf, and to close the Gulf to shipping. Soviet forces are moving into Iran from Afghanistan in support of the rebel forces, and are driving toward Bandar Abbas. To achieve this goal, they must cross the Jebal Barez mountains and the only road runs through the Bam-Darzin pass.

The friendly commanders (Blue) are assigned a mission of. 1) preventing any enemy forces from entering the Bam-Darzin pass itself; 2) doing as much damage as possible to the enemy forces to blunt their drive on the pass and Bandar Abbas; and 3) avoiding Blue casualties to the greatest extent possible.

Treatment Conditions

The experiment manipulates two independent variables: the presence or absence of contingency planning and the level of battle intensity presented in the wargame. Each independent variable has two levels, with the resulting design being a 2 x 2, yielding four treatment conditions.

The two independent variables are defined as:

1. Planning, which manipulates the number of options or contingency plans within the Operational Plan (OPLAN) provided to subjects. The initial position of Blue (friendly) forces is manipulated to be consistent with the OPLAN. The two levels of planning are as follows:

   a. Single-Option Planning, defined as an OPLAN that has one primary estimate of enemy intent, and one primary course of action tailored specifically to meet that threat, with Blue forces positioned accordingly at the beginning of the game.

   b. Multiple-Option Planning, defined as an OPLAN that has multiple estimates of enemy intent, multiple responses, and a better starting position to meet a variety of enemy threats.

2. Battle Intensity, defined here as a composite of two attributes: 1) the type of tank used by the Orange (enemy) forces, and the Orange offensive capability associated with that type of tank; and 2) the interjection of a secondary task of high importance (an unexpected attack off-screen). The two levels of battle intensity are as follows:

   a. Low Battle Intensity, defined as the use of current T-80 tanks by Orange forces as part of their attack force, and the absence of an off-screen attack.

   b. High Battle Intensity, defined as the use of tanks that are 25 percent stronger than the T-80 tank in armor and fire power; and the need to prepare a battle plan in 20 minutes for another attack taking place off-screen.

Within this 2 x 2 design, the main effects of contingency planning and battle workload were tested, as well as their interaction. Analyses were performed by means of a mixed (between — within) analysis of variance design.

Procedure

Each team played the game under all four experimental conditions. Four different scenarios for the enemy attack were used with four different patterns for the enemy approach. Each scenario appeared unique but was technically similar to the other scenarios. These four battle scenarios were developed so that a team never experienced the same scenario twice. Each team played each scenario once, with single- and multiple-option plans having been developed for each scenario. In addition, each scenario depicted a different surprise attack launched by the enemy (e.g., a helicopter assault to Blue's rear or a flank attack by an armored column). In each case Blue headquarters became aware of the surprise attack 30 to 40 minutes into the game. The surprise attack was designed to highlight any difference between single- and multiple-plan conditions. Each team was assigned to one counterbalanced ordering of the four treatment conditions and scenarios.

Data Collection

Data for evaluation of the experiment came from three major sources: 1) the computer system that supported the simulation, 2) expert observers, and 3) the players themselves. The JANUS system provided event driven data that consisted of periodic updates of the location and strength of all units in the game. The reporting period selected for the experiment was every 15 minutes of game time. This data was used to calculate various measures of effectiveness (MOEs) related to movement and attrition of both Blue and Orange forces over the course of the game.

Two expert observers were positioned within each team to monitor activities and collect additional dependent measures. The measurements collected by the observers were divided into two categories: 1) subjective ratings of the overall quality of activity within headquarters, that were recorded every 30 minutes of wall time, and 2) frequency counts of discrete events related to communication within the headquarters and number and type of orders given.

RESULTS

Outcome Measures and Process Measures

The measurement of headquarters effectiveness in the experiment has two components: the outcomes of the wargame and the process by which these outcomes are achieved. Outcome measures indicate the degree to which the Blue teams in the experiment achieved their game objectives; process measures provide detail on how they went about achieving those goals.

The measures of effectiveness (MOEs) used to evaluate the two hypotheses were derived considering Blue's primary goals. Blue was specifically charged with the defense of the Bam-Darzin pass. This included preventing Orange forces from penetrating the pass, as well as holding the positions around the pass. MOEs that assessed how well Blue deterred Orange's advances and kept them away from the pass area are particularly apropos to these goals. Blue, however, was also urged to destroy enemy forces and to be as efficient as possible (e.g., minimize their own attrition and expenditure of resources) while pursuing their primary goals. MOEs dealing with attrition and performance efficiency were used to assess Blue's accomplishment of their charge.

It is important to note that Orange, played by two experienced staff members, was also given specific goals. Orange forces were to penetrate and take control of the Bam-Darzin pass and to do so minimizing their own losses and expenditure of resources. Thus, Blue was confronted with a motivated intelligent adversary.

In particular, the MOEs are: 1) proportion of orange forces with a five kilometer radius of the Bam-Darzin pass — defined as the number of Orange weapon systems within five kilometers of the pass at end game divided by the total number of weapon systems remaining at end game; 2) Blue stopping efficiency — defined as he percentage of the original distance from the pass still remaining for Orange to traverse at end game weighted by the losses Orange incurred at end game; 3) effective advance — defined as the distance Orange traversed trying to attain the pass weighted by the losses incurred traveling that distance; and 4) Orange losses per kilometer advanced — defined as the attrition Blue was able to inflict on Orange for each kilometer Orange moved toward the pass. Although, the four MOEs, are moderately correlated, there was sufficient unique variance to consider each as a separate measure.

An investigation of Outcome Measures alone can not give a complete picture of headquarters effectiveness. Outcome is the end product of a multitude of procedural operations occurring throughout a threat scenario. Measurements of these operations are called process measures since they identify the flow of activity which precipitated the successful completion of mission objectives. Process measures were drawn from records kept by observers (which contain HEAT measures and other observational data).

Testing Hypothesis 1 — Single Versus Multiple Option Plans

Three MOEs speak directly to Blue's primary mission. They are proportion of Orange forces penetrating within a five kilometer radius of the pass, Blue stopping efficiency, and effective advance of Orange forces. Figures 1 and 2 show that for the two dependent variables, effective advance and Blue stopping efficiency, Blue teams were more effective when they had a multiple-option plan than when they had a single-option plan. The effective advance results in Fig. 3 also demonstrate that after the initial time interval, Blue tended to be more effective over time when employing a multiple-option plan.

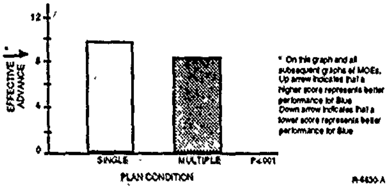


Figure 1. Effective Advance by Plan.

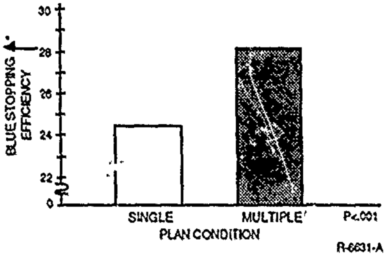


Figure 2. Blue Stopping Efficiency by Plan.

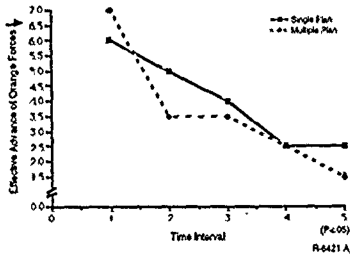


Figure 3. Effective Advance by Plan and Time Interval.

In addition, the MOE weighted more towards attrition showed a stronger pattern favoring a multiple-option plan. Blue was able to inflict a greater loss on Orange for every kilometer Orange advanced, when a multiple-option plan was in effect rather than a single-option plan. These results are shown in Fig. 4, where it can be seen that multiple-option plans result in a 16 percent advantage in attrition per kilometer over a single-option plan.

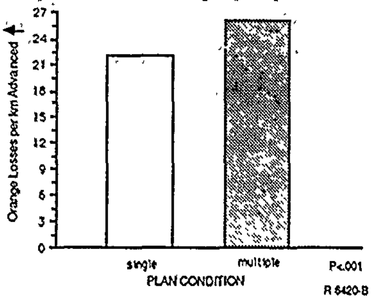


Figure 4. Multiple-Option Plan Results in Greater Effectiveness.

Testing Hypothesis 2 — Plans and Battle Intensity

The second of the two hypotheses to be evaluated implied that the advantage of multiple-option plans over single-option plans would become greater as battle intensity increased. In other words, an interaction between plans and battle intensity was predicted. The results presented in Fig. 5 show that for the two primary MOEs, effective advance and proportion of Orange forces with a five kilometer radius of the pass, as well as for the attrition-oriented MOE, Orange losses per kilometer advanced, the interaction is significant, but not as hypothesized. Blue teams using multiple-option plans are more effective under conditions of low battle intensity, however, are less effective than those using single-option plans under high battle intensity.

The unexpected findings for the plan × battle intensity interaction are puzzling. The expected interaction pattern, as shown in Fig. 6, predicts that a multiple-option plan is somewhat superior in effectiveness to a single-option plan under conditions of low battle intensity. Under high battle intensity, it is predicted that a multiple-option plan would show little change in effectiveness while a single-option plan would show a significant decline in effectiveness. The rationale for such an outcome is reasonably straightforward. A multiple-option plan has prepared for several possible enemy actions or intents while a single-option plan has not. If the enemy departs from the expected action, a multiple-option plan immediately provides well thought-out alternative actions to counter the enemy. A single-option plan has prepared for a single enemy action or intent and when the enemy does something unexpected, a single-option plan has no readily available alternatives. This means during the stress of battle those operating with

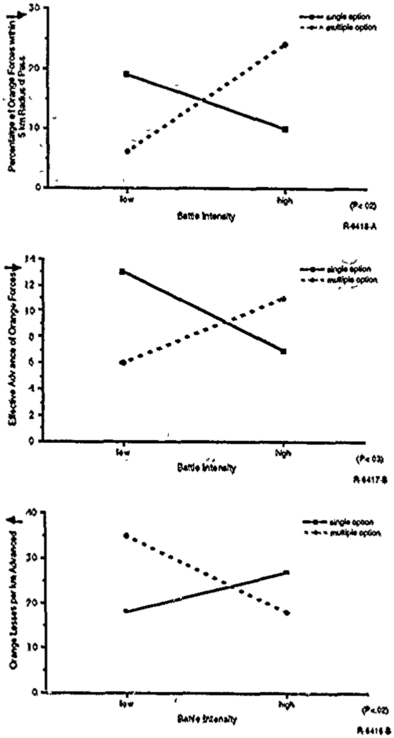


Figure 5. Interaction of Plan and Battle Intensity Disconfirms Expected Outcome.

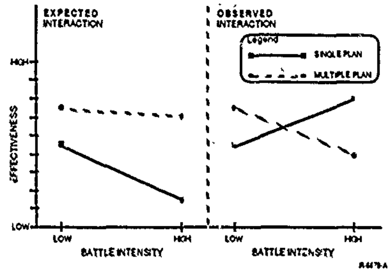


Figure 6. Expected and Observed Interaction Patterns Between Plan and Battle Intensity.

a single-option plan must replan and reorient their forces. Moreover, under the condition of high battle intensity, the time available to do this replanning would be less. Thus, not only must commanders replan and realign their forces, but they must do so under very trying battle conditions.

DISCUSSION

The two primary questions concerning the unexpected interaction that need to be addressed are: why a multiple-option plan was not more effective than it was under high battle intensity, and why a single-option plan was so effective in the same condition. To understand and explain these unexpected interaction results, inspection of the process measure data was undertaken. These process measures included all the interpersonal requests for information occurring within the headquarters (e.g., the number of times: the commander requested information from his staff, the staff requested information from the commander, and the staff requested information from each other). From Fig. 7, it appears that a headquarters' request for information covaried with its effectiveness. Headquarters operating with a multiple-option plan sought more information than headquarters operating with a single-option plan in conditions of low battle intensity and were more effective. In high battle intensity condition, it was headquarters dealing with a single-option plan that requested more information and demonstrated superior effectiveness. It is believed that headquarters that request and obtain the most information gain the best picture of the battle situation and, thus, are best able to fight the enemy effectively.

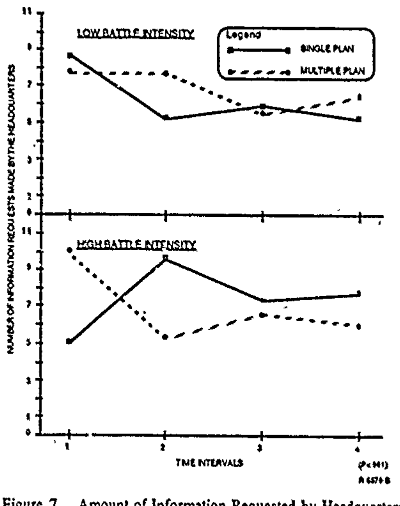


Figure 7 Amount of Information Requested by Headquarters Varied by Experimental Condition Over Time.

The question still remains, why were some headquarters motivated to seek more or less information than other headquarters. We conjecture that the activity level confronting a headquarters and the cognitive limitations inherent in all humans may hold the key. Abstractly, commanders are faced with three general options: they can fight the battle directly, seek information about the battle situation, or think. If the situational demands are low, slack time is ample; then the commander has plenty of time to engage all three options. When situational demands are moderate to high, there is no slack time and commanders must trade off during the options. That is, to spend more time seeking information the commander must put less time and attention in fighting the battle and/or thinking. We hypothesize that multiple-option plan headquarters in the low battle intensity condition enjoyed low situational demand. Our commanders, trained to be active and aggressive used the free time to improve their situational assessment of the battle environment. Headquarters functioning under a single-option plan in low battle intensity faced what for them was a moderate situational demand. They could do the nominal amount of information seeking (fighting and thinking), but no more. They are meeting all their demands and thus would be unwilling to perform one of the options in deference to another for no good reason. Similarly, headquarters working with a multiple-option plan under high battle intensity confront what for them is a moderate situational demand. They, too, are meeting all their demands and would not be motivated to depart from the nominal involvement with the three options. The situation becomes somewhat different for the single-option plan headquarters in the high battle intensity setting. They are confronting a high (maybe very high) situational demand. These headquarters are striving to understand and control the situation to lower the demand. They might wish to think about the problem, but quickly realize that they are employing an internal model of the situation that no longer fits reality. They would like to confront and fight the enemy but with no good assessment of the battle situation, this is difficult. Thus, they engage in the only option that is available in this constrained environment: information seeking. Interestingly, information seeking will help update their internal model and provide a better assessment of the battle.

The above explanation is compatible with principles derived from theories of optimal level proposed by Berlyne (1960), Duffy (1962), Hebb (1949), Leuba (1955), Malmo (1950), and Walker (1964). Such theorists hypothesize that each individual has an optimal level of external stimulation. If the situational demand (or stimulation) is below optimal, individuals strive to adjust the environment to increase the stimulation and bring it closer to optimal. If the situational demand is above the optimal point, then individuals will try to adjust the environment to bring the stimulation level down and closer to their optimal. Commanders of headquarters operating with a multiple-option plan in low battle intensity found their environment under-stimulating, so they sought information to alter the environment to increase stimulation. Conversely, commanders of single-option plan headquarters in high battle intensity found their environment overstimulating. Thus, they strove to alter the stimulation level downward.

Within the constraints of the wargame situation, the only option available to learn about the situation to effect a change was to seek information. At very high levels of stimulation this group would be quite motivated to engage in information seeking.

CONCLUSIONS

Results of the experiment provide support for hypothesis 1. Blue teams were more effective when they worked from a multiple-option plan than when they worked from a single-option plan. Further analysis, however, revealed that for the most part this effect was due to the superior performance of headquarters operating with a multiple-option plan in low battle intensity conditions. Results of an unexpected interaction between plan and battle intensity showed that while a multiple-option plan was better than a single-option plan in low battle intensity condition, the opposite was true in high battle intensity. In this latter condition, headquarters working from a single-option plan enjoyed higher effectiveness than a multiple-option plan. It is surmised that some conditions unique to this experiment interacted with plan and battle intensity to produce this result. It was noted that the multiple-option plan headquarters in the low battle intensity condition and the single-option plan headquarters in the high battle intensity condition sought significantly more information than their counterpart in the same workload situation. It is hypothesized that the additional information gave these headquarters a better picture of the battle and, thus, fostered higher effectiveness. The difference in motivation among the headquarters to seek information was explained by drawing on theories of an optimal level.

REFERENCES

Berlyne, D.E., Conflict, Arousal, and Curiosity, New York: McGraw-Hill, 1960.

Defense Systems, Inc., "HEAT User's Manual," prepared for Defense Communications Agency, July 26, 1984.

Duffy, E., Activation and Behavior, New York: Wiley, 1962.

Entin, E.E., Lentz, P., Mikaelian, D., and MacMillan, J., "Experimental Plan: Effect of Option Planning and Battle Workload on Command and Control Effectiveness in a JANUS Simulation Environment," Technical Report TM-285, ALPHATECH, Inc., January, 1988.

Hebb, D.O., The Organization of Behavior, New York: Wiley, 1949.

Leuba, C., "Toward Some Integration of Learning Theories: The Concept of Optimal Stimulation," Psychological Reports, 1955, 1, pp. 27-33.

MacMillan, J., Entin, E.E., Lentz, P. "Experiment Report: The Effects of Option Planning and Battle Workload on Command and Control Effectiveness," Technical Report, TR-368, ALPHATECH, Inc., January, 1988.

Malmo, R.B., "Activation: A Neuropsychological Dimension," Psychological Review, 1959, 66, pp. 367-386.

Walker, E.L., "Psychological Complexity as a Basis for a Theory of Motivation and Choice, In D. Levine (Ed.), Nebraska Symposium on Motivation (Vol. 12), Lincoln: University of Nebraska Press, 1964.

MATHEMATICAL COMPARISON OF COMPUTER-MODELS TO EXERCISE DATA: COMPARISON OF JANUS(T) TO NATIONAL TRAINING CENTER DATA

Lester Ingber
Physics Department, Code 61IL, Naval Postgraduate School, Monterey, CA 93943

ABSTRACT

Our current project utilizes the powerful techniques of modern nonlinear statistical mechanics to compare battalion-scale combat computer-models (including simulations and wargames) to exercise data. This comparison process is necessary if large-scale combat computer-models are to be extrapolated with confidence to develop battle-management, C^3 and procurement decision-aids, and to improve training.

National Training Center (NTC) data is qualified and used to construct JANUS(T) scenarios. To compare JANUS(T) output data to the NTC exercise data, we develop dynamic mathematical models of each data set, in order to construct a common basis for further comparison. These dynamic mathematical models are at a level of aggregation and in the language of measures of force (MOF) or effectiveness (MOE) appropriate for battle-management and C^2, being stochastic nonlinear generalizations of Lanchester-like mathematical models.

Each data set is fit to several candidate short-time probability distributions, using methods of "fast simulated re-annealing" with a Lagrangian (time-dependent algebraic cost-function) derived from nonlinear stochastic rate equations for MOF's, MOE's, etc. These candidate mathematical models are further tested by using path-integral numerical techniques to develop long-time probability distributions spanning the combat scenario. The best-candidate mathematical models of JANUS(T) data and of NTC data are compared to determine the qualitative as well as the quantitative degree of comparison.

I. INTRODUCTION: C^2 IN TRAINING AND COMPUTER-MODELS

This project addresses the utility of establishing an approach to compare exercise data to large-scale computer-models whose relatively underlying microscopic interactions among men and machines are driven by the natural laws of Physics.

I am Principal Investigator of this project, which is being conducted for the Army Model Management Office (AMMO) by a team of a score of people representing Army TRADOC, NPS, Marine Corps Warfighting Center, and UC Lawrence Livermore National Laboratory.

This Section I outlines some of the issues motivating our general approach. Section II outlines our technical approach to mathematical modeling. Section III then outlines a prototype application to JANUS data. Section IV outlines our general plan of attack. Section V then outlines some specific issues addressed to date. Section VI introduces the main personnel and their tasks in this project.

1. *Necessity of comparing computer-models of combat to exercise data*

In this paper, I will use the term computer-model to include computer simulations as well as computer wargames, the latter involving human participants in real time. Some preliminary studies [1] suggest that JANUS(T) gives similar results when run in batch as a simulation or run with players as a wargame. I.e. it appears that JANUS(T) simulation can be compared favorably to JANUS(T) wargame. In this study, the focus will be to compare JANUS(T) wargame to National Training Center (NTC) data, since both systems then take into account human interactions.

It also should be noted that "large-scale" here refers to battalion-level. If these battalion-level computer-models can be favorably compared, and if consistency can be achieved between the hierarchy of large-scale battalion-level, larger-scale corps-level, and largest-scale theater-level computer-models, then these higher echelon computer-models also can be favorably compared. This could only enhance the value of training on these higher echelon computer-models [2].

The necessity of depending more and more on combat computer-models (including simulations and wargames) has been brought into sharper focus because of many circumstances, e.g.: (a) the nonexistence of ample data from previous wars and training operations, (b) the rapidly shortening time-scale on which tactical decisions must be made, (c) the rapidly increasing scale at which men and machines are to be deployed, (d) the increasing awareness of new scenarios which are fundamentally different from historical experiences, (e) and the rapidly increasing expense of conducting training exercises.

However, the level of acceptance of computer-models in major military battle-management and procurement decisions appears to be similar to the level of acceptance of computer simulations in physics in the 1960's. In physics, prior to the 1960's, theory and experiment formed a close bond to serve to understand nature. In the 1960's, academicians were fascinated with evolving computer technology, but very few people seriously accepted results from computer simulations as being on a par with good theory and good experiment. Now, of course, the situation is quite different. The necessity of understanding truly complex systems has placed computer simulation, together with theory and experiment, as an equal leg of a tripod of techniques used to investigate physical nature.

The requirements necessary to bring combat computer-models to their needed level of importance are fairly obvious. In order to have confidence in computer-model data, responsible decision-makers must be convinced that computer-models model reality, not models of models, or models of models of models, etc. Many people feel that not much progress has been made in the last decade [3] with regard to this issue, despite a general awareness of the problem.

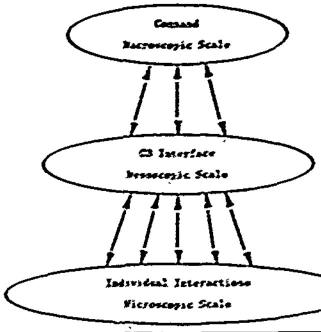
If a reasonable confidence level in computer-models of large-scale combat scenarios could be obtained, there are several immediate payoffs to be gained. More objective data could be presented for procurement decisions, e.g., provided by sensitivity analyses of sets of computer-models differing in specific weapons characteristics. In order to give proper weight to these differing characteristics, their influence within the global context of full combat scenarios would be tested.

2. *Large-scale C^2 and need for mathematical modeling*

Modeling phenomena is as much a cornerstone of 20th century Science as is collection of empirical data [4]. In essentially all fields of Science mathematical models of the real world become tested by fitting some parameters to empirical data. Since the real world is often nonlinear and stochastic, it is not surprising that often this fitting process involves fitting statistical, nonlinear, non-convex functional forms to data.

As in other fields of Science, in the context of modeling combat, reductionist doctrine is simply inadequate to fully understand systems. For example, a threshold is quickly reached at a level of command of any large system, be it physical, biological or social, when a "language" shift is required for effective command and control. A high level commander cannot use a grease-board to track individual units, albeit he might periodically sample his units, but he must rather look at the overall systematics, e.g., aggregated measures of force (MOF) or effectiveness (MOE), attrition, resupply, etc. At this level we properly require command and control (C^2), rather than "supra-battle-management" from commanders. At this level we denote the system as large-scale. (See Fig. 1.)

Figure 1
Scaling C^3 Systems



This issue of refining MOF's and MOE's, e.g., starting at about battalion-level of combat, is relevant to computer-models as well as to actual combat. Merely aggregating data to form MOF's or MOE's does not determine if results from one mission (combat or computer-model scenario) is comparable to another mission. E.g., small differences in tempo or in spatial distribution of FLOT (forward line of our troops), or FEBA (forward edge of the battle area), may cause tables of numbers to appear quite different.

Mathematical models of aggregated data should be expected to uncover "mechanisms" of combat, e.g., like line-firing or area-firing in simple Lanchester theory. More complex missions plausibly will contain more subtle mechanisms as well as weighted contributions of more basic mechanisms. Using this as hindsight, in some systems it may then be possible to specify a figure of merit, some simple set of numbers to capsulize the influence of these mechanisms.

These mechanisms are to be articulated by particular algebraic forms. Indeed, this is the most important sense of Physics, to use mathematical forms to articulate mechanisms and modes of empirical phenomena. For example, it is to be expected that the use of this statistical mechanical approach will facilitate the process of identifying algebraic forms relevant to combat, because in many cases these algebraic forms will be sufficiently similar to other well-known physical mechanics determined in other physics studies.

Awareness of such plausible mechanisms permit the analyst or the field-commander to more easily uncover variations of patterns or common themes unfolding in databases or in real-time. In complex missions, such tools are more than niceties; they can be necessities.

Therefore, to be most useful, computer-model data should be aggregated and then mathematically modeled using variables as close as possible to the level of command to which they are to become decision-aids. The mathematical modeling must by necessity be nonlinear, e.g., offering alternative choices of possible outcomes, upon which human judgement and experience can be brought to bear and be accountable. The mathematical modeling must by necessity be probabilistic, so that expected gains can be humanly weighed with respect to both payoffs and probabilities of payoffs of alternative dynamic (time-dependent) states of the system.

As discussed at a recent conference [5], too often we have weighted the communications aspect of C^3, to the detriment of not properly addressing the command and control (C^2) aspects, i.e., too often only seeking technological fixes to hard large-scale problems. For example, the Soviets give much weight to C^2, and they have structured tabular decision-aids dispersed through their levels of command. However, their relatively rigid political mind-set has fostered the development of these tables by using quasi-linear, essentially deterministic mathematical models fitted to operations data. If we use modern methods of nonlinear stochastic mathematical modeling, using data gained from operations as well as from more advanced computer computer-models base-lined to actual operations, then we can greatly increase our C^2 advantage.

3. Need for mathematical models of aggregated data

The reasons for seeking mathematical models of exercise data correspond to those reasons for seeking mathematical models of computer-models, as described in the above section.

These mathematical models can be used to approach comparison of computer-models, e.g., JANUS(T), to exercise data, e.g., from NTC. I.e., similar mathematical expressions relate to similar mechanisms, whether they exist in the computer-models or in the exercise data. Only if computer-models such as JANUS can be favorably compared to NTC data, can these wargames provide reliable pre- and post-exercise training for NTC commanders.

Such models need to be developed at battalion-level, to drive corps- and theater-level models which require highly aggregated models to run in real-time.

Now, the task in this approach is to find the best mathematical model of each system, i.e., the computer-model system and the exercise-data system. Then, the best mathematical models for each can be compared.

4. Models versus reality

It must be stated that there are still many problems faced by all computer-models of combat, which must be solved before they can be accepted as models of reality.

For example, a very basic problem exists in the quality of acquisition algorithms, i.e., how to construct an algorithm that realistically portrays human attention (pre-attentive as well as selective) and perception, under various combat and weather conditions, night versus day, etc. The Principal Investigator is acutely aware of the influence of attention and perception on complex physical [6-8] and mental tasks [9,10]. Currently, the best combat computer-models model acquisition as serial and logical processes, whereas the human brain acquires data by parallel and associative processes.

Presently, line of sight (LOS) algorithms seem to be the most costly time factor in running JANUS computer-models. Even if more realistic acquisition algorithms are developed, they must be tailored to the needs of real-time computer-models if they are to be used in wargaming and in training. The inclusion of human players in multiple runs of similar scenarios is essential, if a probabilistic mathematical model is to be developed to model exercise data such as that obtained from NTC.

Similarly, in order to develop a computer-model of NTC exercises, to perform the comparison approach in this project, an acquisition or PH (probability of hit) model must be gleaned from the raw data or supplemented by existing algorithms within the computer-model. This is a very difficult problem, requiring subjective, albeit expert, judgment. This is further discussed in Sec. V.

II. TECHNICAL BACKGROUND, GENERAL

1. Problems In Lanchester theory

Quasi-linear deterministic mathematical modeling is not only a popular theoretical occupation, but many wargames, e.g., JTLS (Joint Theater Level Simulation), use such equations as the primary algorithm to drive the interactions between opposing forces.

In its simplest form, this kind of mathematical modeling is known as Lanchester theory:

$$\dot{r} = dr/dt = x_r b + y_r rb$$
$$\dot{b} = db/dt = x_b r + y_b br \qquad (1)$$

where r and b represent Red and Blue variables, and the x's and y's are parameters which somehow should be fit to actual data.

It is well known, or should be well known, that it is notoriously difficult, if not impossible, to use Eq. (1) to mathematically model any real data with any reasonable degree of precision. These equations perhaps are useful to discuss some gross systematics, but it is hard to believe that, for example, a procurement decision involving billions of dollars of national resources would hinge on mathematical models dependent on Eq. (1).

Some investigators have gone further, and amassed historical data to claim that there is absolutely no foundation for believing that Eq. (1) has anything to do with reality [11].

However, although there is some truth to the above criticisms, the above conclusions do not sit comfortably with other vast stores of human experience. Indeed, this controversy is just one example that supports the necessity of having human intervention in the best of C^3 plans, no matter how (seemingly) sophisticated analysis supports conclusions contrary to human judgement. I.e., when dealing with a dynamic complex system, intuition and analysis must join together to forge acceptable solutions.

The use of historical data, to disclaim any truth to the validity of Eq. (1), is in itself the use of inappropriate analysis, as was pointed out in a reply to the previous reference [12]. The aggregation of solitary trajectories from many *different* stochastic combat scenarios does not necessarily form any kind of probability distribution upon which to make statistical judgements. Two combat scenarios, that differ in even only several variables, realistically are going to be quite different scenarios, not least because the very nature of nonlinear nonequilibrium (*open*) competitive systems is to have opposing sides pressed to their extreme, not to their average, capabilities. Given some "maneuvering room" to distort states of an open system, indeed these states will be distorted to more "*favorable*" values.

Therefore, as understood from experience in simulating physics systems, many trajectories of the "same" stochastic system must be aggregated before a sensible resolution of averages and fluctuations can be ascertained. Given two scenarios that differ in one parameter, and given a sufficient number of trajectories of each scenario, then the sensitivity to changes of a "reasonable" algebraic function to this parameter can offer some analytic input into decisions involving the use of this parameter in combat scenarios.

2. Empirical data

Therefore, there are two remaining issues to be resolved. The first is to find a database of a sufficient number of trajectories of the "same" system, upon which mathematical models can be built. The second is to forge an effective approach to mathematically model this data.

The numerous battalion cycles of exercises at the National Training Center (NTC) can provide more trajectories of similar large-scale combat scenarios, than any other source. There appear to be at most about four major categories of exercises, e.g., Blue defender or attacker during night or day.

However, typical of exercises, whose purpose is to train and not necessarily to provide data serving analyses, this data is quite "dirty." [13] Some problems specific to exercises would not occur in actual combat; e.g., multiple laser-targeting does not always determine kills. The present MILES system (Multiple Integrated Laser Engagement System) does not account for range-dependence of probability of acquisition, in addition to not even accurately representing ranges of some weapons systems. There is a tremendous amount of many kinds of data, machine derived as well as derived from human observers in the form of "take-home packages." [14,15] This is a general data-fusion problem which in itself is a formidable task that will require many more years of investigation in many kinds of systems.

The above is not meant to be unconstructive criticism of exercises at NTC. Quite the contrary, while respecting the sensitivity of this data, objective analyses for this project require a complete understanding of these problems.

3. Mathematical Modeling

This brings us to the next issue. What is a "reasonable" mathematical modeling approach?

I believe it reasonable to at least tentatively accept the experience of many commanders, whose intuitions have developed to think in terms of Eq (1). Then, the problem seems to be that the degree of their quantitative, not qualitative, insights is insufficient to detail many combat scenarios. This then becomes the job of analysis, and explicates the purpose as well as the analytic task of mathematically modeling combat data. I.e., a good mathematical model must fit the data, and also be useful as a decision-aid to the commander and decision-maker.

Therefore, we can approach this problem by considering Eq (1) as some kind of zeroth order approximation to reality.

In the late 1970's, mathematical physicists discovered that they could develop statistical mechanical theories from algebraic functional forms

$$\dot r = f_r(r,b) + \sum_i \hat g_r^i(r,b)\eta_i$$

$$\dot b = f_b(b,r) + \sum_i \hat g_b^i(b,r)\eta_i \qquad (2)$$

where the $\hat g$'s and f's are general nonlinear algebraic functions of the variables r and b [16-21]. The f's are referred to as the (deterministic) drifts, and the square of the $\hat g$'s are related to the diffusions (fluctuations). In fact, the statistical mechanics can be developed for any number of variables, not just two. The η's are sources of Gaussian-Markovian noise. The inclusion of the $\hat g$'s, called "multiplicative" noise, recently has been shown to very well mathematically and physically model other forms of noise, e.g., shot noise, colored noise, dichotomic noise [22-24]. At this time, certainly the proper inclusion of multiplicative noise, using parameters fit to data to mathematically model general sources of noise, is preferable to improper inclusion or exclusion of any noise.

The ability to include many variables also permits a "field theory" to be developed, e.g., to have sets of (r,b) variables (and their rate equations) at many grid points, thereby permitting the exploration of spatial-temporal patterns in r and b variables. This gives the possibility of mathematically modeling the dynamic interactions across a large terrain.

4. Support for present mathematical modeling approach

These new methods of nonlinear statistical mechanics only recently have been applied to complex large-scale physical problems, demonstrating that empirical data can be described by the use of these algebraic functional forms. Success was gained for large-scale systems in neuroscience, in a series of papers on statistical mechanics of neocortical interactions (SMNI) [25-36], and in nuclear physics [37-40]. I have proposed that these methods be used for problems in C^3 [22,34,41,42].

Thus, now we can investigate various choices of f's and $\hat g$'s to see if algebraic functional forms close to the Lanchester forms can actually fit the data. In physics, this is the standard phenomenological approach to discovering and encoding knowledge and empirical data, i.e., fitting algebraic functional forms which lend themselves to physical interpretation. This gives more confidence when extrapolating to new scenarios, exactly the issue in building confidence in combat computer-models.

The utility of these algebraic functional forms in Eq. (2) goes further beyond their being able to fit sets of data. There is an equivalent representation to Eq. (2), called a "path-integral" representation for the long-time probability distribution of the variables. This path-integral representation is driven by a "Lagrangian," which can be thought of as a dynamic algebraic "cost" function. The path-integral possesses a variational principle, which means that simple graphs of the algebraic cost-function give a correct intuitive view of the most likely states of the variables, and of their variances. Like a ball bouncing about a terrain of hills and valleys, one can quickly visualize the nature of dynamically unfolding r and b states.

Especially because we are trying to mathematically model sparse and poor data, different drift and diffusion algebraic functions can give approximately the same algebraic cost-function when fitting short-time probability distributions to data. The calculation of long-time distributions permits a clear choice of the best algebraic functions, i.e., those which best follow the data through a predetermined epoch of battle. Thus, dynamic physical mechanisms, beyond simple "line" and "area" firing terms, can be identified. Afterwards, if there are closely competitive algebraic functions, these insights can be more precisely assessed by calculating higher algebraic correlation functions from the probability distribution.

It must be clearly stated that, like any other theory applied to a complex system, these methods have their limitations, and they are not a panacea for all systems. For example, probability theory itself is not a complete description when applied to categories of subjective "possibilities" of information [43,44]. Other non-stochastic methods are likely appropriate for determining other types of causal relationships, e.g., the importance of reconnaissance to success of missions [15]. I feel that these statistical mechanical methods are appropriate for comparing these stochastic large-scale combat JANUS and NTC systems, the details of our studies will help to determine the correctness of this premise.

5. AI versus physics

It is useful to contrast this approach of mathematically and physically modeling data to that of artificial intelligence (AI). Actually, the two approaches should be viewed as complementary. Given a codified knowledge base of a large system, AI techniques can effectively manage this knowledge. In contrast, these physics techniques seek to discover knowledge embedded in the data. Without having this knowledge first extracted, AI tools are typically poor at managing data to extrapolate to new scenarios. However, once these statistical physics techniques formulate and algebraically functionalize the knowledge, AI techniques can be very beneficial to create heuristics based on the probabilistic mathematical models, to enhance the efficiency of data retrieval.

6. Current state of calculations

Recently, two major computer codes have been developed, which are key tools for use of this approach to mathematically model combat data.

The first code fits short-time probability distributions to empirical data, using a most-likelihood technique on the Lagrangian. CPT Steve Upton, USMC, while he was my thesis student at NPS, and I developed code for two-temperature fast simulated annealing to fit many parameters, e.g., usually entering as coefficients of polynomials of variables in such systems as Eq. (2) [45,46]. This code was modeled after other methods of simulated annealing [47] and fast simulated annealing [48].

An algorithm of very fast simulated re-annealing has been developed to fit a empirical data to a theoretical cost function over a D-dimensional parameter space [49]. The annealing schedule for the Temperatures T_i decrease exponentially in time t, i.e., $T_i = T_{i0}\exp(-c_i t^{1/D})$. No rigorous proofs have been given. It is expected that the obvious utility of this algorithm will motivate such proofs. However, actual fits to data are a finite process, and often even only heuristic guides to algorithms that obviously fit many classes of data are important. Heuristic arguments have been given here that this algorithm is faster than the fast Cauchy annealing [48], where $T_i = T_0/t$, and much faster than Boltzmann annealing [47], where $T_i = T_0/\ln t$.

The second code develops the long-time probability distribution from the Lagrangian fitted by the first code. CDR John Connell and LCDR Charles Yost, USN, while they were my thesis students at NPS, and I developed a Cauchy-driven Monte Carlo code for a one-variable Lagrangian [50]. Recently, I have developed a multi-variable Cauchy-driven Monte Carlo code for nonlinear multivariate problems.

A more robust and accurate histogram path-integral algorithm to calculate the long-time probability distribution has been developed by other researchers to handle nonlinear Lagrangians [51-53], and they are presently working to create a code to process several variables. The Monte Carlo code we developed can handle many variables, but does not seem as accurate as theirs for similar times of computer processing for one variable. However, the Monte Carlo code is faster for many variables, and does not have high memory requirements as does the other code. Therefore, it could be used on a PC, e.g., on the battlefield to help predict upcoming scenarios, after the commander has made a judgement as to what kind of scenario (equivalent to choosing a prefitted Lagrangian) is unfolding [42].

For this first study, we propose developing the histogram algorithms for two or four variables. For this project, Dr. Wehner has developed a two-dimensional algorithm, capable of handling complex boundary conditions. We hope to extend these methods to higher dimensional spaces.

III. TECHNICAL BACKGROUND: JANUS(L) PROTOTYPE MATHEMATICAL MODEL

To illustrate some of the issues, consider a conventional combat scenario, e.g., of the kind we have mathematically modeled using JANUS(L) simulation data.

There are variables, spatial-dimensions, and parameters that must be processed by such calculations. Typically people have considered only a few variables, e.g., one or two, in one or two dimensions, with several parameters; or, they have considered limiting cases of huge/infinite number of variables/dimensions. These problems require breaking new ground into the nonlinear nonequilibrium stochastic realm of 10, 20 or 30 dynamic variables. I believe this number is barely large enough to give reliable analysis/aids to decision-makers, yet barely small enough to be able to process good scientific calculations. We must avoid handling too many variables which leads quickly to data overload of machines as well of humans, and we must avoid doing too simplistic modeling which is at best unreliable for complex systems.

First consider a simple example, one that we have fit using JANUS(L) data. We aggregate the time-dependent attrition of Red and Blue forces, starting in an initial simple configuration, on a flat terrain. (See Fig. 2.)

The composition of the scenario is as follows:

Blue Force (Defense)

3 task forces

each task force composed of:

12 M60A3 tanks

4 M901 vehicle-mounted TOW

total force: 48

"pop-up" parameter: time to defilade = 15 sec

Red Force (Offense)

3 task forces

each task force composed of:

30 T72 tanks

9 BMP infantry fighting vehicle with mounted 30mm gun

total force: 117

No "breakpoint" was modeled; i.e., forces engaged until one was completely annihilated. The Conflict Simulation Center of the UC Lawrence Livermore National Lab provided us with 20 trajectories of this scenario, spatially aggregated over the entire terrain, each of about 12 minutes duration, with data provided about every half-minute. (See Fig. 3.)

Figure 2

Initial Configuration of Red and Blue Forces

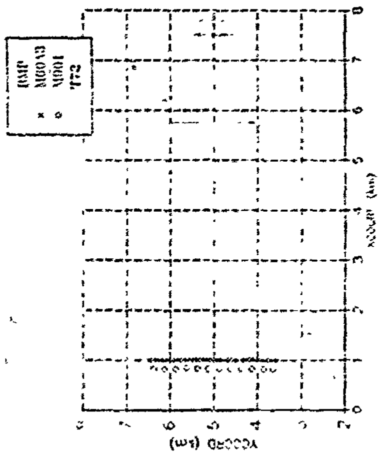
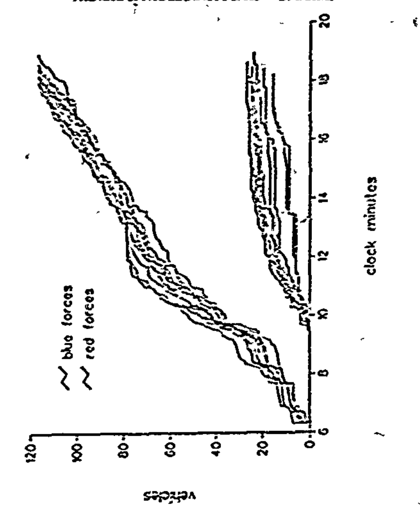


Figure 3
Attrition of Red and Blue Forces — 20 Battles



Consider the scenario mapped out in time and in the two spatial dimensions, where the spatial dimensions are coarsely grained into a three by three grid of cells, e.g., labeled from (1,1) through (3,3). (See Fig. 4.) Consider that Blue forces have strategically set themselves into a defensive posture on the West German border at some site, here to be located in the middle left column cell. Consider Red forces have strategically set themselves into an offensive posture on the East German border, here located throughout the right column of three cells. Consider that all microscopic stochastic algorithms defining each person's and machine's interactions at given distances and velocities have already been programmed into the simulation.

Figure 4
Conventional Red Versus Blue Scenario

		R
B		R
		R

There will be long ranged interactions, e.g., via artillery, as well as short-ranged ground interactions. Each cell defines an independent set of Blue and Red stochastic mesoscopic variables. E.g., in cell (2,3) there may be two kinds of Blue forces/weapons, etc., interacting with the other Blue and Red variables at all cells.

To appreciate the magnitude of the problem being presented, consider a relatively simple mathematical model of Blue and Red each possessing only one type of force/weapon. Further simplify the problem by considering that the rate of change of each variable in each cell is driven by time-independent algebraic functions: a drift-force with terms proportional to Red attrition, and terms proportional to Red attrition multiplied by Blue attrition; a multiplicative noise term composed of a constant background, and terms proportional to Red or Blue attrition. I.e., consider four intra-cell parameters per Blue and Red variable per cell. In addition, for simplicity, consider only nearest-neighbor (NN) interactions between cells, effected by adding linear terms terms to each drift proportional to Red and Blue attrition. Thus each NN requires an additional 4 parameters per cell. I.e., each corner cell has 2 NN, the middle cell has 4 NN, the others have 3 NN each. Then this defines a 168-dimensional parameter-space of coefficients in a mathematical model-Lagrangian defined by an 18-dimensional variable-space in two spatial-dimensions, to be fit by a maximum-likelihood algebraic function of the short-time probability distribution of the variables to combat simulation data. Of course, for well-known scenarios, intuition gained by working with experts in combat analysis will greatly reduce the number of meaningful parameters to be considered.

The cells serve to aggregate the appropriate mesoscopic variables, here to be considered as the spatial-temporal attrition of Blue and Red units during the course of the battle. This then describes a classic pattern-recognition problem, to describe the spatial-temporal evolution of these variables.

Cell 1:

Rate	Intra-Cell	Inter-Cell
$\dot{r}_1 =$	$x^r_b b_1 + y^r_{br} b_1 r_1 + z^r \eta_{r_1} + z'^r r_1 \eta'_{r_1}$	$+x^r_b b_2 + x^r_r r_2$
$\dot{b}_1 =$	$x^b_r r_1 + y^b_{rb} r_1 b_1 + z^b \eta_{b_1} + z'^b b_1 \eta'_{b_1}$	$+x^b_b b_2 + x^b_r r_2$

(3)

Cell 2:

Rate	Intra-Cell	Inter-Cell
$\dot{r}_2 =$	$x^r_b b_2 + y^r_{br} b_2 r_2 + z^r \eta_{r_2} + z'^r r_2 \eta'_{r_2}$	$+x^r_b b_1 + x^r_r r_1$
$\dot{b}_2 =$	$x^b_r r_2 + y^b_{rb} r_2 b_2 + z^b \eta_{b_2} + z'^b b_2 \eta'_{b_2}$	$+x^b_b b_1 + x^b_r r_1$

(4)

$r_{1,2}$: red attrition, number of casualties in cell 1,2

$b_{1,2}$: blue attrition, number of casualties in cell 1,2

$\eta_{1,2}$: uncertainty, white noise in cell 1,2

$\eta'_{1,2}$: uncertainty, multiplicative noise in cell 1,2

x, y, z, z': parameters to be fit to data

$x_{1,2}$-terms: attrition due to direct "line" firing

$y_{1,2}$-terms: attrition due to "area" firing

$z_{1,2}$, $z'_{1,2}$-terms: uncertainty in physics and C^3 information

Note that, in general, the x's, y's and z's may be time-dependent, but in this first set of studies, they are taken as constants. Although this statistical mechanics approach can process this time-dependence, it greatly adds to the resources necessary to fit the data.

The z' terms include the interesting physical mechanism, describing the uncertainty in attrition during each short interval of time as being proportional to the total force at the beginning of the interval. This effectively introduces a highly nonlinear log-normal behavior, but presents no additional problems for our quite general calculational procedures.

To see the utility of using the Lagrangian, consider just simple additive noise:

$$\dot{r} = x^r_b b + y^r_{br} br + z^r \eta_r$$

$$\dot{b} = x^b_r r + y^b_{rb} rb + z^b \eta_b$$

$$r = \sum_i \theta^r_i R_i \, , \, b = \sum_j \theta^b_j B_j \qquad (5)$$

where the z's and θ's are constants, and where the R's and B's represent specific aggregates of units or weapons systems. We do not arbitrarily assign weapons effectiveness index/weapons unit value (WEI/WUV) to systems.

The Lagrangian provides a dynamic algebraic cost function mathematically equivalent to the above rate equations. If we assume that the noise η_r is uncorrelated with the noise η_b, then

$$L = \frac{(\dot{r} - x_r^b b - y_r^b b r)^2}{2 z^{r2}} + \frac{(\dot{b} - x_b^r r - y_b^r r b)^2}{2 z^{b2}}$$

$$\dot{r} = \Delta r / \Delta t \,, \Delta r = r(t + \Delta t) - r(t)$$

$$\dot{b} = \Delta b / \Delta t \,, \Delta b = b(t + \Delta t) - b(t) \tag{6}$$

In terms of L, the probability P of obtaining a change in r and b during Δt is

$$P = (2\pi \Delta t)^{-1} (z^r z^b)^{-1/2} \exp(-L \Delta t) \tag{7}$$

Nonlinear multiplicative noise induces a Riemannian geometry in the space of variables, and more care needs to be taken in defining the Lagrangian. The ideal size of Δt can be derived from the Lagrangian [51,52].

We have fit the relatively simple JANUS(L) tank scenario described above (the z's) using the one-cell aggregation mathematical model of Eqs. (5) and (6). We found that the simple form of Eq. (5) gives a lower algebraic cost-function than algebraic cost-functions involving more complicated forms for the drifts, defined in Eq. (2), e.g., Padé approximates of ratios of polynomials of r and b, or those involving multiplicative noise, e.g., including the z' terms in Eqs. (3) and (4). We know of no other studies that have compared such nonlinear stochastic mechanisms. In future studies the θ', the weights given to different units or weapons systems, will be fit simultaneously with the z's.

Given this form for L, we use the path-integral to calculate the long-time distribution of variables r and b at time t, given values of r and b at time t_0. To compact the notation, is is useful to use $M = \{r, b\}$ as a vector of these variables. Also for compactness, the boundary conditions confining the possible values of r and b are omitted.

$$P[M_t | M_{t_0}] = \int \cdots \int dM_{t-\Delta t} dM_{t-2\Delta t} \cdots dM_{t_0+\Delta t}$$

$$\times P[M_t | M_{t-\Delta t}] P[M_{t-\Delta t} | M_{t-2\Delta t}] \times \cdots P[M_{t_0+\Delta t} | M_{t_0}]$$

$$P[M_t | M_{t_0}] = \int \cdots \int DM \exp\left(-\sum_{s=0}^{u} \Delta t L_s\right)$$

$$DM = (2\pi m_0 \Delta t)^{-1/2} \prod_{s=1}^{u} (2\pi m_s \Delta t)^{-1/2} dM_s$$

$$dM = dr\, db$$

$$m = \det \begin{bmatrix} z^{r2} & 0 \\ 0 & z^{b2} \end{bmatrix} = z^{r2} z^{b2}$$

$$M_0 = M_{t_0} \,, M_{u+1} = M_t \tag{8}$$

This is obviously impossible to calculate in closed form, and we therefore must use numerical methods. For simplicity of presentation here, Eq. (8) is written in a "prepoint discretization," which disguises more complex algebra arising from a Riemannian geometry induced by nonconstant multiplicative noise [20,41].

It must be emphasized that the output need not be confined to complex algebraic forms or tables of numbers. Because L possesses a variational principle, sets of contour graphs, at different long-time epochs of the path-integral of P over its r and b variables at all intermediate times, give a visually intuitive and accurate decision-aid to view the dynamic evolution of the scenario. (See Fig. 5.)

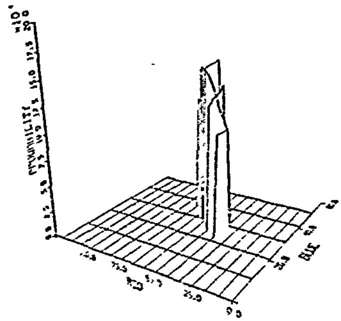
After L has been fit to data, i.e., using our method of very fast simulated re-annealing, Eqs. (4) and (5) can be immediately presented to interface with analysts more familiar with this Lanchester like mathematical representation of the physical system. It is also possible to devise an objective "measure of fit" of the best-candidate theoretical probability distribution, P^T, to the empirical distribution, P^E, of data from the system (JANUS or NTC). For example, the "information" or "entropy" difference might be measured as

$$I^{TE} = \int P^T \ln(P^T / P^E) \tag{9}$$

where the integral goes over the space of the variables.

Figure 5
The Decision-Aid

The horizontal axes represent Red and Blue forces at one time slice, nine minutes into the tank battle. The vertical axis represents the long-time probability of finding values of these forces. The Lagrangian used to fit the data represents a Lanchester drift with additive noise. In general, the probability will be a highly nonlinear algebraic function, and there will be multiple peaks and valleys. For clarity of presentation, all points whose probability is less than half maximum have been set to zero.



IV. GENERAL PLAN OF ATTACK

1. Establish variables and parameters for mathematical modeling

a. The existing NTC attrition delineates the variables and rough ranges of parameter values for some classes of possible mathematical models. I.e., we are fitting Lanchester-like coefficients in short-time probability distributions to the data.

b. Other computer codes are calculating the long-time probability distribution of the underlying variables, e.g., MOF's.

c & d. Steps 1a and 2b above are similarly performed for the data obtained from scenarios on JANUS(T) which at least contain the NTC terrain and similar exercises.

2. Qualify NTC database

Specifically, "unqualified" NTC data means that the data has not yet been made consistent with the take-home packages. Total kills in the take-home packages are typically a factor of 2 to 4 higher than those recorded on magnetic tape, due to influences of the controllers, etc. Missing movement orders of some units have not been reasonably filled in (e.g., only ~ 1/3 of units seem to be tracked). Resurrections and fratricides have not been addressed, with respect to how they should affect JANUS play, etc.

a. The NTC database for particular missions are being qualified. Computer codes have been developed to facilitate this qualification. This requires using the information in "take-home packages," written summaries of missions to qualify the data recorded on magnetic tape. All qualified data is inserted into an INGRES database, to facilitate the transfer of data, e.g., initial force structures and movement orders, directly into JANUS(T).

b. Driven by the requirements of 1 above and 3 below, the archived database gleaned in (a) is qualified to facilitate both mathematical modeling of aggregate data and construction of JANUS(T) scenarios similar to NTC exercises.

3. Construction of computer-model scenarios

Using NTC terrain and the database gained in 2 above, computer-model scenarios are constructed on JANUS(T). In order to do this a PH algorithm for NTC is being developed, for all combinations of targets and shooters, static and moving. Also, PK's consistent with those at NTC, using the MILES recording devices, are put into the JANUS(T) database.

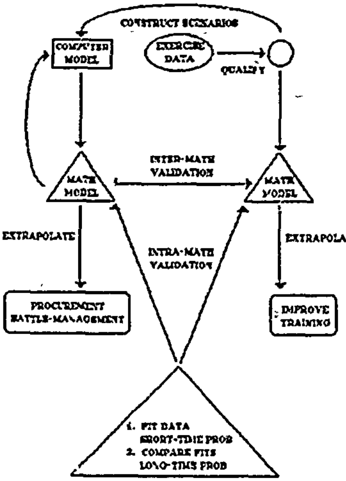
4. Baseline NTC and JANUS(T)

Steps 1a through 1d will be iterated, using data obtained from steps 2 and 3. The objective is to determine which physical input parameters of JANUS(T) likely need to be modified, in order that essentially similar dynamic mathematical models can be obtained from both the NTC database and the attrition and and force data output from the computer-model. A computer-model scenario will be constructed that will reasonably statistically conform to the NTC exercises, e.g., to give the same total kills at the end of the battle.

5. Compare NTC and JANUS(T)

This process should serve at least two purposes. We will learn to what degree existing computer-models have to be modified to conform to exercise-reality. Also, quantitative insights may be gained on how to improve the data-collection process at NTC exercises.

Figure 6
Mathematical-Model Comparison Process



V. QUALIFICATION OF NTC DATA & JANUS SCENARIOS

This section is a summary of just a few issues to date, that we have considered with respect to the qualification of NTC data and the construction of JANUS scenarios. Because the NTC program is highly sensitive, we do not plan on releasing our results piece-meal, but rather we will wait until the completed report is finished.

We immediately face a potential competition between two scientific issues in our project:

(a) If we tweak JANUS to an arbitrary degree just to fit selected NTC data, then we risk destroying the good Physics in JANUS, perhaps its primary asset, just to fit this selected data. Then, the resulting computer-model cannot be trusted to be extrapolated to missions not included in the original fit.

(b) However, if we don't change JANUS to at least include features/data present at NTC, albeit not present in today's view of actual combat, then we are not giving JANUS a "fair chance," and also we are not taking advantage of perhaps the sole source of existing exercise data that contains (somewhat) similar replications—which is essential for comparing/modeling stochastic systems.

The issue probably is one of degree. We all seem to agree that present JANUS PK's should be changed in our study to the MILES PK's. But what about PH's and acquisition algorithms? Although JANUS is not confused about how to acquire data, how to assess PH's, etc., our human generation is very uncertain on how to understand, much less model, these processes in real humans, at NTC or in any other complex real-life setting.

While computer modelers must be complimented for establishing a logically neat and fast serial ordering of acquisition, determination of priorities, aim-hit, kill, acquire again, etc., most of these modelers would be among the first to admit that their algorithms do not necessarily completely or correctly reflect human processing. For example, a novice in any combative discipline might be trained to perform these processes in serial order, but especially in the heat of an intense battle against multiple threats, any experienced combatant most certainly performs many of these mental-physical processes in parallel, not just serially, and constrained at least by limited capacity of selective attention, by short-term memory, and by the ability to patternize information.

Many computer modelers in other academic disciplines now look to neural nets or to statistical mechanics of neocortical interactions, to better understand how the human brain most likely processes patterns of information. Perhaps someday, combat computer-models will incorporate neural nets and statistical mechanical neural processors to help model acquisition, but they certainly don't come close now.

Even given a better acquisition and PH models, people at different levels of proficiency will have different acquisition and PH processes, which are very much environment-dependent. They are not so conveniently separated in humans, and they are mixed differently in different people.

To emphasize our lack of understanding of these issues, consider the questions: Should soldiers be trained to separate these mental processes, perhaps markedly improving their abilities to react quickly, flexibly and "intuitively" to multiple threats? People could be better trained to handle patterns of multiple threats, but at what cost and for how long must they be trained? Do we have enough knowledge to implement this training, much less to code this knowledge into our computer-models?

I argue/offer these points, referencing my own published physics-neuroscience (the SMNI series above) and karate research (also referenced above), as well as my 30 years' of experience in training thousands of hand-to-hand combatants, many training intensely for several years. I'm sure that there are some other experts with as much experience who would disagree with me on any given point.

The bottom line is that, at NTC we undoubtedly have quite a mix of many levels of training and natural mental-physical abilities. These issues are at the heart of differences to be expected between JANUS data and the NTC data. We must keep this perspective as we attempt to compare JANUS to NTC data.

Our project is to compare JANUS to NTC data, and to provide the Army with a "what-if" capability of JANUS for NTC commanders. To do this, we must leave intact as much of the algorithmic guts of JANUS as possible, while possibly changing some databases relating to physics, e.g., PK's. This focus is an essential ingredient of our project: We are probing the consistency and utility of NTC data by enforcing the disciplines of constructing a computer-model and a mathematical model, in the process of also comparing JANUS data to NTC data.

Still, we probably will be forced to enter into a grey area, in part because we must somehow qualify NTC data to the extent we can construct similar JANUS scenarios, and in part because we are tasked to prepare JANUS for "what-if" scenarios. To do this, we likely will have to modify the acquisition and/or PH attributes of JANUS, to at least come close to achieving the same time-dependent attrition in both systems.

We plan to change the more physical PH's (e.g., relatively more removed from internal cognitive processes than is JANUS acquisition), scaling these tables to at least match total kills in JANUS to those in NTC take-home packages. This approach has the advantage of simultaneously and consistently developing a PH algorithm for the NTC mission, while qualifying the NTC time-dependent kill data. In doing so, I am aware that some of this scaling is really "sweeping under the rug" ("renormalizing" is a nicer technical term) otherwise-necessary modifications to the JANUS acquisition algorithms which certainly don't precisely match acquisition at NTC.

It remains to be seen if our approach, of mathematical-model comparison of JANUS data and NTC data, can provide objective scientific proof that these human factors can be folded into a stochastic description, so that, at least at the level of aggregation at which a commander views the combat, it is indeed reasonable to state that JANUS data can be favorably compared to NTC data.

Comparison versus Validation

Many people object to considering NTC exercises as representing true combat; some even object to it being considered true training. E.g., what tactics does a creative commander practice/apply at NTC when a TOW's range is artificially cut off within a tank's range? [Is proper training for commanders only endorsed if only actual combat tactics are practiced? Are commanders trained to be flexible enough to quickly adapt to different weapons' systems and opposing-force tactics?] Many people object to having the worth of a computer-model judged against such data. [Are our combat computer-models flexible enough to accommodate different weapons' systems and tactics?]

NTC has real people and real machines, albeit they may be so constrained by non-combat exercise rules, some Red-but-painted-Blue equipment, unrealistic weapons' ranges as portrayed by laser devices, etc. Most important, NTC has (barely) enough similar replications of such combat-like exercises, that it uniquely can serve as a source of data to begin to fill an important gap: We can no longer put off at least forging an approach to compare computer-models to real-life data of some sort. This discipline may simply point out serious differences between the computer-model data and the exercise data. Given the intense discussions these issues always seem to raise, it is clear that the Army must be informed of these discrepancies if they exist.

VI. METHOD OF SOLUTION AND PERSONNEL

A collection of technical approaches and personnel are participating in this project. Mr. Hue McCoy, TRAC-MTRY, is the Technical Director. MAJ Rick Halek, AMMO, is the Administrative Director.

1. Mathematical Modeling

To fit theoretical parameters to empirical data, very fast simulated re-annealing computer codes are used to fit various trial Lagrangians to data, i.e., a short-time probability distribution, determining the multivariate (deterministic) drifts and diffusions (fluctuations).

CPT Stephen Upton, USMC, at the MC Warfighting Assessment Division, Quantico, and Ms. Lois Brunner of the NPS Mathematics Department and C^3 Group, are establishing bounds on mission-specific weights to aggregate force units. We intend to find the the precise weights as part of this fitting process.

To calculate long-time probability distributions from these fitted short time distributions, computer codes have been developed to perform the nonlinear path-integral of the Lagrangians derived above. This is essentially a mathematical intra model checking process, ensuring that the algebraic functional forms fit above make good physical sense. Dr. Mike Wehner in Division B at LLNL has developed these computer codes.

2. Database qualification

Mr Mike Uzelac, of BDM, at LLNL, Conflict Simulation Center (CSC), is providing qualified data to make this total project possible. The CSC is working closely with NPS personnel so that the data is used efficiently for both mathematical modeling and for construction of computer-model scenarios. CSC has developed an "Analyst Workstation" which can serve to efficiently scan patterns of NTC data to qualify it, as well as to provide a "script" for JANUS(T) play to replicate NTC scenarios. They have already done this with a sample of NTC data.

3. Construction of JANUS(T) scenarios

TRAC-MTRY is constructing a PH algorithm for NTC. These personnel are using the chosen NTC scenarios to construct the needed JANUS(T) scenarios, given the proper data from part 2 above. LTC John Miller, USA, Chief TRAC-MTRY at NPS, is being tasked for this purpose. LTC Vernon Bettencourt will become Chief TRAC-MTRY in July 1988. TRAC-MTRY has JANUS(L) and JANUS(T). Key personnel at TRAC-MTRY are MAJ Hirome Fujio, USA, and MAJ Bernie Galing, USA.

CPT Mike Bowman, USA, has joined this project as my thesis student. He and MAJ Fujio also have worked with TRAC-WSMR analysts, Mr. Gary Flack and Mr. Al Kellner, to develop a software interface between the NTC INGRES database and the JANUS(T) database.

I expect that many more NPS students can find interesting theses in any of the above three major aspects. This manpower will inevitably contribute to this project. For example, two of my NPS M.S. thesis students are directly addressing very current scenarios requiring joint operations. LT Jack Gallagher, USN, is studying trade-offs in using Navy Tomahawk missile support of Army-Air Force air-land scenarios, versus using tactical nuclear weapons. The Army JANUS(T) computer-model is being modified to study the Tomahawk scenarios, and LLNL JANUS(L) computer-model, baselined to the same JANUS(T) scenarios, is being modified to study the tactical nuclear scenarios. LCDR Roy Balaconis, USN, is studying trade-offs in using Tomahawk versus SLAM support of these air-land scenarios. We plan to use statistical mechanics algorithms, developed for our NTC project, for sensitivity analyses of the data provided by these combat computer-models.

REFERENCES

[1] Shugart, P.A "An inquiry into the utility of wargames," Report No. ATRC-WD-1, White Sands Missile Range, NM. U.S. Army TRADOC Analysis (TRAC), 1987.

[2] Bartlett, G.T., "Battle Command Training Program." *Phalanx* 21, 1988, pp. 18-20.

[3] Comptroller General, "Models, Data, and War: A Critique of the Foundation for Defense Analyses," Report No. PAD-80-21, Washington, D.C., U.S. General Accounting Office, 1980.

[4] Jammer, M., *The Philosophy of Quantum Mechanics*. New York, NY: Wiley & Sons, 1974.

[5] University, C^3CM Joint Test Force and National Defense, Thinking Red in Wargaming, Albuquerque, NM, 1987.

[6] Ingber, L., *The Karate Instructor's Handbook*, Solana Beach, CA. PSI-ISA, 1976.

[7] Ingber, L., *Karate, Kinematics and Dynamics*. Hollywood, CA. Unique, 1981.

[8] Ingber, L., *Elements of Advanced Karate*. Burbank, CA. Ohara, 1985.

[9] Ingber, L., "Editorial: Learning to learn." *Explore* 7, 1972, pp. 5-8.

[10] Ingber, L., "Attention, physics and teaching." *J Social Biol Struct* 4, 1981, pp. 225-235.

[11] Dupuy, T.N., "Can we rely on computer combat simulations." *Armed Forces J.* August, 1987, pp. 58-63.

[12] Hollis, W.W., "Yes, we can rely on computer combat simulations." *Armed Forces J* October, 1987, pp. 118-119

[13] National Security and International Affairs, "Need for a Lessons-Learned System at the National Training Center," Report No. GAO/NSIAD-86-130, Washington, D.C.. U.S. General Accounting Office, 1986.

[14] Levine, R.A., Hodges, J.S., and Goldsmith, M "Utilizing the data from the Army's National Training Center: Analytic Plan," Report No. N-2461-A, Santa Monica, CA. RAND, 1986.

[15] Goldsmith, M and Hodges, J "Applying the National Training Center Experience. Tactical Reconnaissance," Report No. N-2628-A, Santa Monica, CA: RAND, 1987.

[16] Dekker, H., "Quantization in curved spaces," in *Functional Integration: Theory and Applications*edited by J.P Antoine and E. Tirapegui New York: Plenum, 1980, pp. 207-224

[17] Grabert, H. and Green, M.S., "Fluctuations and nonlinear irreversible processes." *Phys. Rev. A* 19, 1979, pp. 1747-1756.

[18] Graham, R., "Covariant formulation of non-equilibrium statistical thermodynamics." *Z. Physik* B26, 1977, pp. 397-405.

[19] Graham, R., "Lagrangian for diffusion in curved phase space." *Phys. Rev. Lett.* 38, 1977, pp. 51-53.

[20] Langouche, F., Roekaerts, D., and Tirapegui, E., *Functional Integration and Semiclassical Expansions*. Dordrecht: Reidel, 1982.

[21] Schulman, L.S., *Techniques and Applications of Path Integration* New York: J. Wiley & Sons, 1981.

[22] Ingber, L., "Nonlinear nonequilibrium statistical mechanics approach to C^3 systems," in *9th MIT/ONR Workshop on C^3 Systems. Naval Postgraduate School, Monterey, CA, 2-5 June 1986*Cambridge, MA: MIT, 1986, pp. 237-244.

[23] Fox, R.F., "Uniform convergence to an effective Fokker-Planck equation for weakly colored noise." *Phys. Rev. A* 34, 1986, pp. 4525-4527.

[24] van der Broeck, C., "On the relation between white shot noise, Gaussian white noise, and the dichotomic Markov process." *J. Stat. Phys.* 31, 1983, pp. 467-483.

[25] Ingber, L., "Towards a unified brain theory." *J. Social Biol. Struct.* 4, 1981, pp. 211-224.

[26] Ingber, L., "Statistical mechanics of neocortical interactions. I. Basic formulation." *Physica D* 5, 1982, pp 83-107.

[27] Ingber, L., "Statistical mechanics of neocortical interactions. Dynamics of synaptic modification." *Phys. Rev. A* 28, 1983, pp. 395-416.

[28] Ingber, L., "Statistical mechanics of neocortical interactions. Derivation of short-term-memory capacity." *Phys. Rev. A* 29, 1984, pp. 3346-3358.

[29] Ingber, L., "Statistical mechanics of neocortical interactions. EEG dispersion relations." *IEEE Trans. Biomed. Eng.* 32, 1985, pp. 91-94.

[30] Ingber, L., "Statistical mechanics of neocortical interactions: Stability and duration of the 7±2 rule of short-term-memory capacity." *Phys. Rev. A* 31, 1985, pp. 1183-1186.

[31] Ingber, L., "Towards clinical applications of statistical mechanics of neocortical interactions." *Innov. Tech. Biol. Med.* 6, 1985, pp. 753-758.

[32] Ingber, L., "Statistical mechanics of neocortical interactions." *Bull. Am. Phys. Soc.* 31, 1986, pp. 868.

[33] Ingber, L., "Applications of biological intelligence to Command, Control and Communications," in *Computer Simulation in Brain Science: Proceedings, University of Copenhagen, 20-22 August 1986*edited by R. Cotterill London, Cambridge U., 1988, p. to be published.

[34] Ingber, L., "Statistical mechanics of mesoscales in neocortex and in command, control and communications (C^3): Proceedings, Sixth International Conference, St. Louis, MO, 4-7 August 1987." *Mathl. Comput. Modelling* 11, 1988, pp. 457-463.

[35] Ingber, L., "Mesoscales in neocortex and in command, control and communications (C^3) systems," in *Systems with Learning and Memory Abilities: Proceedings, University of Paris 15-19 June 1987*edited by J. Delacour and J.C.S. Levy Amsterdam: Elsevier, 1988, p. to be published.

[36] Ingber, L. and Nunez, P.L., "Multiple scales of statistical physics of neocortex: Application to electroencephalography." 1988, pp. submitted.

[37] Ingber, L., "Nuclear forces." *Phys. Rev.* 174, 1968, pp. 1250-1263.

[38] Ingber, L., "Riemannian corrections to velocity-dependent nuclear forces." *Phys. Rev. C* 28, 1983, pp. 2536-2539.

[39] Ingber, L., "Path-integral Riemannian contributions to nuclear Schrödinger equation." *Phys. Rev. D* 29, 1984, pp. 1171-1174.

[40] Ingber, L., "Riemannian contributions to short-ranged velocity-dependent nucleon-nucleon interactions." *Phys. Rev. D* 33, 1986, pp. 3781-3784.

[41] Ingber, L., "Statistical mechanics algorithm for response to targets (SMART)," in *Workshop on Uncertainty and Probability in Artificial Intelligence. UC Los Angeles, 14-16 August 1985*Menlo Park, CA: AAAI-RCA, 1985, pp. 258-264.

[42] Ingber, L., "C^3 decision aids: Statistical mechanics application of biological intelligence," in *1987 Symposium on C^3 Research. National Defense University, Washington, D.C., 16-18 June 1987*Washington, DC: National Defense University, 1987, pp. 49-57.

[43] Zadeh, L., "A computational theory of dispositions." *Int. J. Intelligent Sys.* 2, 1987, pp. 39-63.

[44] Goodman, I.R., "A probabilistic/possibilistic approach to modeling C^3 systems part II," in *1987 Symposium on C^3 Research: National Defense University, Washington, D.C., 16-18 June 1987*Washington, DC: National Defense University, 1988, pp. 41-48.

[45] Ingber, L. and Upton, S., "Stochastic model of combat," in *1987 Symposium on C^3 Research: National Defense University, Washington, D.C., 16-18 June 1987*Washington, DC: National Defense University, 1987, pp. 59-63.

[46] Upton, S.C., "A statistical mechanics model of combat," M.S. Thesis, Monterey, CA: Naval Postgraduate School, 1987.

[47] Kirkpatrick, S., Gelatt, C.D. Jr., and Vecchi, M.P., "Optimization by simulated annealing." *Science* 220, 1983, pp. 671-680.

[48] Szu, H. and Hartley, R., "Fast simulated annealing." *Phys. Lett. A* 122, 1987, pp. 157-162.

[49] Ingber, L., "Very fast simulated re-annealing." 1988, pp. to be published.

[50] Connell, J., Ingber, L., and Yost, C., "Statistical mechanical virtual neural computer," in *1987 Symposium on C^3 Research National Defense University, Washington, D.C., 16-18 June 1987*Washington, DC: National Defense University, 1987, pp. 65-68.

[51] Wehner, M.F. and Wolfer, W.G., "Numerical evaluation of path-integral solutions to Fokker-Planck equations. I." *Phys. Rev. A* 27, 1983, pp. 2663-2670.

[52] Wehner, M.F. and Wolfer, W.G., "Numerical evaluation of path-integral solutions to Fokker-Planck equations. II Restricted stochastic processes." *Phys. Rev. A* 28, 1983, pp. 3003-3011.

[53] Wehner, M.F. and Wolfer, W.G., "Numerical evaluation of path integral solutions to Fokker Planck equations. III Time and functionally dependent coefficients." *Phys. Rev. A* 35, 1987, pp. 1795-1801.

MODELING AND SIMULATION OF ARMY DIVISION HEADQUARTERS USING PETRI NETS

Kenneth R. Madsen

ALPHATECH, Inc.
111 Middlesex Turnpike
Burlington, MA 01803

ABSTRACT

Results of a research effort focusing on the organizational and functional relationships that exist in an Army Division level command post structure are presented. The study integrates the information contained in five US Army documents and develops a functional flow of the situation estimate and planning processes. An IDEF formulation captures the hierarchical nature of the organization as well as explicit dependencies between its functions and tasks. The headquarter decision processes are then modeled using Stochastic, Timed Attributed Petri Nets (STAPN)s and the model is implemented on a Macintosh computer using Micromodeler, a STAPN modeling tool. Simulations performed to examine the sensitivity of the headquarters' process measures to automation, existence of contingency plans, and staffing flexibilities are presented. Finally, these results are analyzed in light of the command post structure and used to propose some hypotheses for future testing in headquarters exercises or war gaming situations.

1. INTRODUCTION

This paper describes a study conducted under contract with the Defense Communications Agency's Center for Command, Control and Communications Systems under the Wartime Headquarters Improvement Program. This program is in its second phase of the development of the Headquarters Effectiveness Assessment Tool (HEAT) HEAT was designed to enable a team of trained observers to objectively assess and quantify headquarters performance and effectiveness, to report their results and diagnosis, and to provide information to DCA to help them plan for improvements to C^3 systems and equipment to support future headquarters activities.

The purpose of this study was to develop a prototype Petri net simulation of the Army division-level headquarters and to examine its sensitivities to automation, contingency planning, and staff size and flexibility. The study began with a visit to an Army command post exercise (Cascade Peak IV), and continued with a detailed analysis of available Army training documents (listed in References Section). That analysis produced an organizational and functional decomposition of the Army division command post system and a Petri net simulation of the estimation and planning process within the main command post Section 2 of this paper traces the research effort that lead to that simulation. Section 3 describes the baseline and excursion models and Section 4 analyzes the simulation findings.

2. THE ARMY DIVISION COMMAND POST STRUCTURE

ORGANIZATIONAL STRUCTURE

The Army division organizational structure is well documented in ARTEP 71-100-MTP [1]. This structure starts with the three division command posts (DTAC, DMAIN, and DREAR). These three command posts are designed to support the commander in each of the three areas of the combat operations battlefield. close, deep and rear The close battle is fought to defeat committed enemy forces The deep battle is meant to disrupt or destroy enemy follow-on forces. The rear battle is fought to preserve one's own options by protecting uncommitted reserves and supply forces The main command post (CP) is the hub of the division headquarter structure, and should be capable of supporting the commander in any of the three battlefield areas at all times. All intelligence (G2) and operations (G3) tasks are also performed in the main CP When the commander is physically located at one of the other CPs part of that support may then reside in that CP. The TAC CP is the smallest of the three and is designed primarily to support the close battle. To decrease the size of the main CP much of the personnel (G1) and some of the logistics (G4) tasks reside in the rear CP. The rear CP provides a place for these elements to perform many of the supervisory and sustaining tasks that are necessary but not critical to the immediate battle area.

FUNCTIONAL RELATIONSHIPS

While the ARTEP contains a detailed description of the organizational structure of the Army command post system, it does not describe the functional relationships of its cells and elements. R&D Associates, in support of the Battle Command Training Program Office, conducted a functional analysis of the division headquarters and reported their analysis in a draft technical report [2] This analysis included postulated combat functions for each headquarters element (command, personnel, intelligence, operations and logistics) and a decomposition of those functions into their tasks and subtasks. RDA also developed a series of functional flow diagrams which describe the internal information flow for the commander and each of the four major staffs (personnel, intelligence, operations and logistics).

THE MILITARY DECISIONMAKING PROCESS

Army Field Manual (FM) 101-5 describes military decisionmaking as "both an art and a science" [3] The commander must continually make critical decisions under uncertain situations based on incomplete and questionable information The military decisionmaking process is an orderly process that includes recognizing the problem, gathering facts and making assumptions, developing possible solutions, analyzing and comparing those solutions, and selecting the best solution This process (diagramed in Figure 1) begins with receipt of the mission by the commander.

This mission is usually assigned by a higher headquarters, but it may be developed by the commander Based on staff inputs about the current situation, the commander then conducts

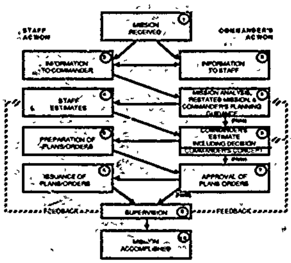


Figure 1. Military Decisionmaking Process.

a mission analysis. This analysis results in his identification of the tasks to be performed, the purpose of those tasks and the constraints involved. Often the major constraint is time, so the commander must allocate the time that his staff may use to develop their estimates. As a rule, the commander will allocate two thirds of the available time to his subordinate units leaving one third to his staff. If he feels that time will not allow a complete estimate of the situation, the commander can complete the decisionmaking process on his own and issue verbal orders to his subordinate units (via the path marked with "note" beside the arrows in Figure 1).

When the commander completes his mission analysis, he restates the mission and issues planning guidance to his staff. The staff officers prepare their staff estimates and return to the commander with their recommended courses of action. The estimate process involves considerable information exchange among the staff and will be discussed in detail in a later paragraph. After receiving the staff recommendations the commander completes his estimate of the situation and decides on a course of action to accomplish the mission. He announces his decision and his concept, or visualization of the entire operation, to his staff and subordinate commanders. The staff prepares plans and orders to implement that concept, and presents them to the commander for his approval. After approval is received, the staff issues the plans and orders to subordinate units. The commander and his staff continue to receive feedback and to supervise the subordinate units as they make their plans and execute the orders. The military decisionmaking diagram is the basis of the Petri net simulations developed in this study and will be referenced frequently throughout this paper.

THE ESTIMATION PROCESS

During the staff estimation process, staff officers collect and analyze relevant information and develop the most effective solution possible with the time and information available. Figure 2 illustrates the process as defined by CGSC Student Text 100-9 [5].

The process begins with the receipt of the commander's mission statement, initial planning guidance and time analysis. Following the commander's mission analysis the G3 staff establishes the area of operations and provides that information to the G2 staff. G2 analyzes the weather and terrain in the area of operations and provides it to G3 and G4. Each staff then analyzes the effects of weather and terrain on their functional area. G2 determines the disposition, composition, strengths and weaknesses of the enemy situation while G3 analyzes the same factors for the division's own forces (own situation). The own

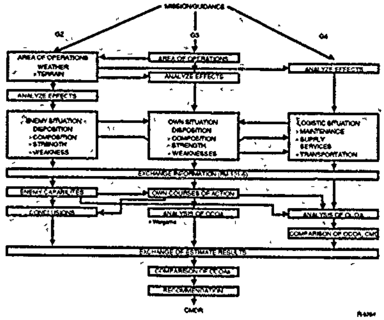


Figure 2. Staff Estimation Process.

situation analysis relies on G4's input of the logistic situation (maintenance, supply, services, transportation, etc.). After an exchange of information among the staffs, G2 analyzes the enemy capabilities and provides the information to G3 and G4. At this point G3 enumerates possible plans or courses of action. Each staff analyzes these options in light of its functional area. During this analysis G3 wargames each course of action and identifies advantages and disadvantages of each option. After an exchange of analysis results, G3 compares the courses of action, develops their recommendation and briefs the commander.

This section has summarized our research of the literature describing the Army division command post system. The resulting functional decomposition adequately defines the decisionmaking process and information flow within the main command post. The following section will describe the computer models that were developed based on this decomposition.

3. ARMY DIVISION COMMAND POST MODELS

The previous section described the organization and functional flow of the Army division command post structure. It also described the process by which that organization reaches decisions and generates plans and orders for subordinate units. This section will describe the simulation techniques used to model the structure and procedures of the main command post.

IDEF ALPHA REPRESENTATION

The ARTEP describes, in a hierarchical fashion, the organizational decomposition of all three Army division command posts. Each CP is composed of a number of cells that are organized along functional lines. For example the plans cell is responsible for generating future division plans (among other duties) while the current operations cell is responsible for monitoring and dealing with the current situation. Each cell is composed of various organizational elements such as operations, intelligence, and logistics. While the current operations cell is predominantly composed of G3, it, like most cells, includes other supporting elements as well. As discussed in Section 2, each element has functions to perform and tasks within each function. While the ARTEP details each task by element, by command post, it does not contain a graphical representation of this hierarchy.

IDEF ALPHA is an ideal hierarchical, graphical tool for representing the Army division headquarters structure. IDEF

ALPHA, developed by ALPHATECH, Inc.[6], is an Apple Macintosh-based graphical tool that allows the user to build, save, modify, and print hierarchically decomposed IDEF diagrams. IDEF (ICAM Definition) was developed as a structured approach for graphically relating manufacturing tasks and functions for the ICAM (integrated computer-aided manufacturing) community. The approach has since been adapted for use by the C^3 community.

This study included a complete categorization of the Army division command post structure into the IDEF format using IDEF ALPHA. The IDEF representation decomposed each division command post into its cells, elements, functions, and tasks as defined in the ARTEP and expanded by the RDA report. Due to space limitations the IDEF formulation could not be incorporated into this paper.

PETRI NET REPRESENTATION

One of the more intriguing bases for C^3 modeling is the field of Stochastic, Timed, Attributed Petri Nets (STAPNs) [7]. Invented in the early 1960's (by Petri) to characterize concurrent operations in computer systems (described as networks), Petri nets have been extended over the years to capture almost all of the important aspects of large man/machine organizations (such as attributes, timing relations, and stochastic events). Their greatest appeal is their conceptual simplicity. Also, quite natural behavioral variables accompany each simple element of a STAPN. In addition, the topology of a STAPN model automatically determines a number of relations between these variables.

Micromodeler

The Army division headquarters decisionmaking process was modeled using Micromodeler, a graphical simulation tool based on the STAPN technology discussed above. Micromodeler was developed by ALPHATECH, Inc. as a prototype modeling tool for the Air Force Foreign Technology Division [8]. Micromodeler currently runs on the Apple Macintosh computer, but ALPHATECH is developing its successor (C^3 Modeler) to run on a Sun Workstation. With Micromodeler, the analyst can build and reconfigure models constructed from Petri net elements, then run event stepped simulations based on these models.

Micromodeler uses the Petri net primitives (transitions and places) with one addition: the box. A box is a Petri net element which contains places, transitions and other boxes inside it. The purpose of boxes is to hide detail and to allow building of models in a top-down, modular fashion. Models are built by drawing them on the computer screen using icons. Micromodeler translates these drawings into a simulation. Figure 3 shows the screen layout of Micromodeler.

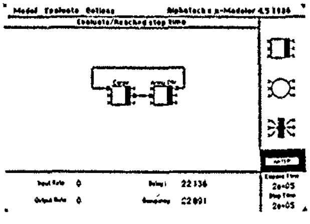


Figure 3 Micromodeler Screen Layout.

The large rectangular window is called the drawing window. This is the portion of the screen where the Petri net simulation is drawn and later run. This screen happens to be the top-level screen of the model used in this study. The pictures, or icons, located on the right-hand side of the screen are the Petri net elements (box, place and transition). The small window located below these icons is the title box. This window contains the name of the current view ("ARTEP" in Figure 3) and can be helpful in identifying the portion of the model pictured and its relationship to the rest of the model. The bottom area contains statistical data collected during the simulation run. Any box in the drawing window can be exploded to reveal its contents. Figure 4 shows the screen that results from exploding the "Army Div" box in Figure 3. Note that the title of this screen, or view, is "Army Div"; thus tying it to the previous screen.

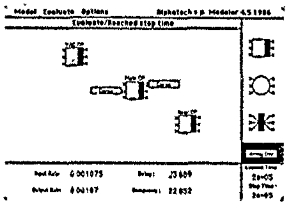


Figure 4. Exploding "Army Div" Icon.

Micromodeler is a unique tool for simulating the headquarters process, but it does have limitations. As the icons on the screen of Figure 4 indicate, the analyst can only connect three inputs and three outputs to any Petri net element (box, place or transition). This limitation will be noted later in the staff sizing discussion of Section 3 of this paper. Another limitation is the lack of multiple token attributes in Micromodeler. The use of multiple attributes would provide for a much richer simulation which could include an assessment of information quality or content as well as its mere existence. Efforts are currently underway at ALPHATECH to eliminate these and other limitations of Micromodeler.

COMMAND POST SIMULATIONS

The Baseline Simulation

The military decisionmaking process (Figure 1) served as the basis of the Petri net simulations for this study. Figure 5 duplicates the process in the Micromodeler format.

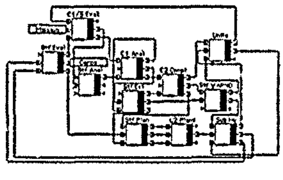


Figure 5. Petri Net Simulation.

The STAPN primitives behind each box simulate the information flow and tasks and subtasks within each box. The content of these boxes include all tasks discussed in Section 2 and illustrated in the military decisionmaking (Figure 1) and staff estimation (Figure 2) diagrams. The tokens in the system (not visible in Micromodeler representations) represent the information, plans and orders flowing between the commander and his staffs.

The baseline Petri net simulation was designed as a sequential planning model. In this simulation the headquarters staff is involved in developing a single plan from the initial input of battle and intelligence information to the completion of the planning process. This is somewhat unrealistic because actual headquarters staffs are continually monitoring the situation and updating their estimates of the situation in a process of "rolling planning". Sequential planning was modeled, however to assure that no resource contention was involved in the baseline simulation. It also provided a convenient measure of staff efficiency — planning time. Planning time, as measured by the micro modeler simulation, was the complete cycle time of the planning activity. Parallel planning would have prevented the measurement of this baseline efficiency.

This timing information resulted from the individual task times that are inputs to the Micromodeler program. The baseline times used in this study were defined in FM 101-5 as typical planning times required for a division staff to react to a change in mission within an ongoing operation. These times are shown in Figure 6 in tabular form.

OPR	ACTIVITY	TIME ESTIMATE
Cdr	Mission Analysis	20 min
Cdr	Estimate of Situation	75 min
G2	Characteristics of Area	
G2	Enemy Situation	
G3	Own Situation	
G3	Relative Combat Power	
G2	Enemy Capabilities	
Cdr	Own Courses of Action (C/As)	60 min
G3	Analysis of Own C/As	
G3	Comparison of Own C/As	
Cdr	Decision	
G3	Warning Order	30 min
G3	Preparation of OPORD	
All	Staff input to OPORD	90 min
Cdr	Approval of OPORD	
G3	Distribution of OPORD	60 min
G3	Acknowledge OPORD	
TOTAL DIV HQ		5.5 hrs
Subordinate Hq Planning		11 hrs
TOTAL		16.5 hrs

Figure 6. Typical Planning Times.

The table also incorporates the commanders 1/3, 2/3 rule (discussed earlier) for allocating planning time to his subordinate headquarters. The results of the baseline simulation will be discussed in Section 4 of this paper.

The Excursion Simulation

The baseline simulation incorporated several assumptions: 1) sequential planning (one plan being worked at a time), 2) no automation to speed up the tasks, 3) no contingency plans in existence that may shorten the planning process, and 4) no resource contention (tasks could always be completed when the required information was available). The study then examined three major excursions from this baseline: automation, contingency plans and staff sizing/flexibility. These excursions required varying degrees of changes to the baseline simulation described above.

AUTOMATION

In 1985 the Combined Arms Operations Research Activity (CAORA) conducted an analysis of the G3 section of the Army corps and division main command posts [9]. The purpose of their analysis was to identify opportunities for aiding the performance of the G3 during tactical operations through the use of computer applications. The assessment was based on the near-term (five-year) automated environment of main CPs and current U.S.Army doctrine. As in the RDA study, a structured functional analysis was performed to identify specific G3 main tasks and products. Additionally, CAORA assessed and prioritized aiding opportunities, and recommended research priorities to develop those aids. Figure 7 contains a partial list of the high-priority G3 analytic aiding opportunities identified in the CAORA study.

UNIT MOVEMENT PLANNER
FORCE MOVEMENT ANALYZER
AIR MOVEMENT ANALYZER
AIR MOVEMENT PLANNER
TERRAIN MANAGEMENT
TIME ANALYZER
COMPARE ALTERNATE COURSES OF ACTION
FORECAST UNIT STATUS
ROUTE EVALUATION
RELATIVE COMBAT POWER

Figure 7. Analytic Aiding Opportunities.

The items on that list relate to specific tasks simulated in this study using Petri nets. Those tasks were then identified as automation candidates for the automation excursion. The purpose of the excursion in this study was to determine the possible sensitivities of automation on the overall planning process; not to examine individual aids themselves. Therefore, the STAPN transition delay (task) times identified as automation candidates were shortened for the automation excursion. Based on previous findings using HEAT, a maximum 4:1 multiplier was assumed in time savings for each task that was automated with an intermediate step of 2:1 time savings. These automation assumptions were first introduced only to G3 functions, then extrapolated to G2, G4 and commander tasks. The results of the automation excursions will be discussed and compared with the baseline results in Section 4 of this paper.

CONTINGENCY PLANS

The purpose of contingency planning is to account for situational uncertainty. The importance of contingency planning has been identified as a critical issue in the HEAT Program and the early program experiments have been dedicated to the hypothesis that contingency planning improves the effectiveness of a headquarters operation. This study addressed the question of contingencies using the Petri net simulation of the Army division headquarters. FM 101-5 indicates that the commander may choose to short-circuit the decisionmaking process and issue verbal orders to the subordinate units in time-critical situations According to FM 101-5 this decision would be made after the commander completes his mission analysis (step 3) While none of the Army documents available addressed such an alternative, we hypothesized that the commander could interrupt the normal process at the same point (step 3) to implement a contingency plan.

Previous HEAT work indicates that, while commanders often consider situational contingencies, they do not always share these contingencies with their staff; nor do they always have the time to develop written plans to address all contingencies considered.

Figure 8 addresses these situational contingencies and associated planning possibilities. This overlay to the military decisionmaking process interrupts the process after the commander's mission analysis (step 3) by determining whether a contingency plan exists to cover the current situation. If none exists, the process continues as usual with the staff estimates (step 4). If the commander does have an existing contingency plan, he then determines whether it is one known only to himself (implicit) or a written contingency plan with corresponding orders already in the field (explicit). If either type of contingency plan exists the staff estimate is no longer needed. The commander knows what he should do, has made his decision and is ready to proceed to develop plans and orders. If the contingency plan is an implicit one the formal preparation and approval of those orders must still be accomplished. If, on the other hand, the contingency plan is an explicit one, the commander can issue verbal orders to the field units to execute the appropriate orders that have already been written to address the current situation. The results of the contingency plan study will be presented and compared with the baseline in Section 4 of this paper. A combination of automation and contingency planning will also be examined.

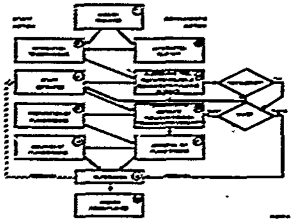


Figure 8. Military Decisionmaking Process with Contingency Plans.

STAFF SIZING

The CAORA study report included a list of the personnel who work in the G3 section of a heavy division main command post. That list, which was derived from the current table of organization and equipment (TOE) and from Army Field Circular 101-55 (not available for this study), totaled to 33 people in both the current operations [7] and plans (26) cells. No other reference to staff sizing was found in the documents available for this study. Presumably the time estimates discussed above were based on a standard staff size as defined by the TOE.

The purpose of this study was to examine the sensitivities of the division main command post efficiency to perturbations, therefore, staff sizing was altered on a multiple basis to examine that sensitivity. The baseline study assumed that any task could be accomplished once the required input information was available. For the staff sizing study, the baseline Petri net simulation had to be altered to model the fact that a staff member must also be available to perform that task. That staff member would be unavailable to perform other tasks during the duration of the task he was committed to. He then became available to perform some other task in the system. In addition to information inputs, each task transition now included a staff input. Recalling the basics of Petri nets, the transition can only fire when tokens are available in each of the input places. Therefore, a staff token must be available in the staff input place before any task can begin. The same staff token(s) is also used to enable other tasks performed by the same staff; either from the same Petri net place or from another place within the staff loop.

For the initial staff sizing study a separate loop of t a-sitions and places was developed for each of three staffs: the commander (and his immediate staff), G3 and G2. Due to limitations of Micromodeler the G4 staff could not be simulated separately. G4 continued to operate as in the baseline (no staff contention). Every task had already been identified by the staff that was to perform it (G2, G3, G4 or Commander). With the exception of G4 tasks all task transitions were now served by one of these three staff loops. The loops were then loaded with tokens to represent the staff size multiples (1,2,..6). One token was considered to be a single unit that could perform each task in the estimated time length. The existence of multiple tokens allowed multiple tasks to be performed simultaneously, thus speeding up the process and permitting parallel activities.

To stress the system so that parallel activity was needed, the program driver was also changed for this study. Rather than drive the program with input information only after the previous planning cycle was complete, the inputs were now sequenced by clock time. The rate of those inputs was then altered (from once every four hours to once every thirty minutes) to increase the activity level. The baseline efficiency measure was found to be inadequate for the staff sizing study because of the varying information input rates. Therefore, a new measure, the percent of plans completed was calculated to relate the headquarters output to the input volume for each case.

A final change was then made to examine the effects of cross training among G2 and G3 personnel. Any G2 and G3 staff tokens could now serve both G2 and G3 tasks. This increased staff flexibility should allow the G2 and G3 staffs to keep up with an increasing workload until both staffs are saturated. The results of the staff sizing and cross training studies will be discussed in Section 4 of this paper.

4. ANALYSIS AND RESULTS

Previous sections of this paper have developed the organizational, procedural and theoretic basis of the Army command post study. This section will present the results of that study, analyze and compare those results, and suggest any future work to be done as follow-up.

BASELINE CONFIGURATION

The baseline configuration was discussed in Section 3 of this paper. The efficiency measure for the baseline, automation and contingency simulations (planning time) was computed by dividing the total battle time by the total number of headquarters outputs and converting to hours. The total battle time was made large to let the simulation noise figures settle. Each transition (task) time was set as an exponential delay rather than a constant delay, so a short battle may have given spurious results. The baseline simulation resulted in an average planning time of 18 hours over the period of 127 days (total battle time). During that time the staff generated 167 plans. These numbers are not important in themselves, although it is comforting that the steady state planning time (18 hours) closely followed the typical time (16 1/2 hours) given in FM 101-5 and used as simulation inputs. The difference can be accounted for by the additional unit preparation, movement and battle time that was simulated within the "units" box of the Petri net model. The importance of the results lies in the sensitivities to be found in the excursions to be discussed in the following paragraphs.

EXCURSION CONFIGURATIONS

Automation Study

The automation excursion examined the sensitivities of shortening selected task times for the G3 staff, then to all staffs (G2, G3, G4, and commander) involved in the estimation and planning process. Not all tasks were assumed to benefit from automation. Task selection was based on the list of analytic aid opportunities found in Figure 7. In the first excursion, selected G3 task times were cut in half while all other task times remained constant. The impact on planning time was negligible, as seen in Figure 9.

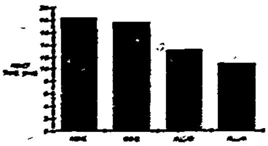


Figure 9. Automation Study Results.

This finding was not surprising because, while there is a great deal of parallel activity in the estimation process, there are also numerous bottlenecks. Each time information is exchanged among staffs or provided to the commander, the process stops until all staffs have completed their work. Therefore, automating only one staff causes them and their computer-aided estimates to have to sit and wait for the other staffs to catch up.

Applying the same automation assumptions to all staffs reduced the planning time from 18 hours to 13 hours, a 28% improvement. The final test of this excursion study was to go to the 4:1 improvement limit that was identified in earlier HEAT studies as the maximum possible. Cutting the selected task times of all staffs by four resulted in an average planning time of 11 hours, a 39% improvement to the baseline (18 hours) but only a 15% improvement on the 2:1 assumption (13 hours). Clearly there is a point of diminishing returns, partially due to the fact that not all tasks can be automated.

Contingency Study

The contingency plan excursion examined the impact of varying degrees of pre-planning on the efficiency of the Army headquarters. Section 3 discussed the two types of contingency plans considered and the rationale for their selection. The Petri net model was run for both implicit and explicit contingencies with the probability of their existence ranging from 20% to 100%. A stochastic decision rule determined in each situation whether a plan existed to cover that situation. As Figure 10 indicates, the impact of implicit (known only to the commander) contingency plans was minimal.

Even if all situations (100%) were covered by his mental planning only a 11% planning time improvement to the baseline was realized. Explicit (written) contingency plans had a much greater impact with a 50% improvement limit. Written contingency plans produced the greater impact primarily because of the additional short-cuts that they permitted. Not only was the staff estimation process by-passed (as with the implicit plans), but the formal plan preparation and approval was eliminated at the subordinate headquarters as well as the division command post. The results do not address the pre-planning time required to have these written plans available, but only the impact of their existence. While the validity of these assumptions has not been established, their sensitivity warrants further study.

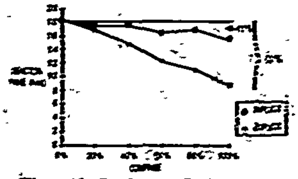


Figure 10. Contingency Study Results.

Automation and Contingencies

The final excursion using the sequential planning model was a combination of automation and contingencies. The results of these tests appear in Figure 11.

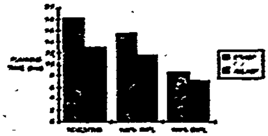


Figure 11. Automation and Contingencies.

This chart assumes that either no situations (the first pair of bars) or all situations are covered by a contingency plan and compares the results of no automation (light-hatched bars) to a 2:1 automation time reduction in all staffs (dark-hatched bars). The new information is in the two right-hand dark-hatched bars and shows the improvement when automation is added to contingency planning.

The combination of automation and <u>implicit</u> contingency planning results in an average 12 hour planning time, or a 33% improvement to the baseline. The corresponding combination using <u>explicit</u> contingency plans results in an average 7 hour planning time, or a 61% improvement to the baseline. This, the lowest cycle time of the study, represents a 2 hour, or 22% improvement to the the explicit contingencies alone (9 hours) and a 4 hour, or 31% improvement to the automation results (13 hours). The headquarters efficiency seems to be more sensitive to contingency plan availability than to automation improvements.

Staffing Study

The baseline, automation and contingency studies assumed that adequate staff was always available to perform a task when the input information was available. The staffing study removes that assumption by requiring that shared staff members also be available. To stress the staff, this study also increased the information input rate to force parallel activities. The assumption here is that each staff token in the Petri net simulation can perform each task in the time derived from FM 101-5. However they can now only work on a single task at any one time, possibly forcing other tasks to be put off until staff members (tokens) are available. A staff multiple of two (or two staff tokens) meant that two such staffs were available in the staff loop and could thus perform two different tasks simultaneously. Multiples of four and six were also used to examine the sensitivities.

Because of the parallel nature of the simulation and the varying information input rate, average planning time could no longer be used as the primary measure of headquarters efficiency. Instead, percent of plans completed was calculated and plotted (Figure 12) to illustrate the system sensitivity to staff sizing over a range of information input rates. The chosen measure, percent plans complete, was calculated by dividing the number of staff inputs by the number of completed staff plans and converting to a percentage.

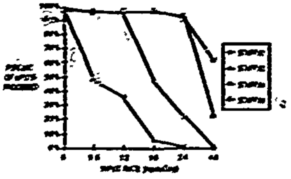


Figure 12. Staff Sizing Results.

Figure 12 compares these percentages for staff multiples of 1, 2, 4 and 6 over a range of input rates from six inputs per day (every four hours) to 48 inputs per day (every half hour). With four-hour input intervals all staff sizes performed at near 100% efficiency. However, as the rate increased the single staff quickly became bogged down until, at 24 inputs per day (hourly), they were unable to produce any plans. This complete saturation is a result of the non-optimal nature of the staff loops and clearly does not represent sound management practice. Staff tokens tended to get stuck in a single task area, continuing to work off the stack of inputs rather than throwing away all but the latest and processing it alone. Micromodeler cannot currently support LIFO (last in, first out) strategies, but work is now under way to improve its capabilities. As Figure 12 indicates, adding more staff tokens moves the knee of the curve to the right. The absolute values of these results are not significant, but the comparison reveals that staff sizing does impact headquarters efficiency.

A close examination of the staff sizing simulations revealed that, when approaching the saturation point, the G2 staff usually saturated first. This phenomenon resulted from the fact that G2 had critical collection planning tasks to perform as well as situation estimation duties. This finding lead to the final staffing study — the impact of cross training on headquarters efficiency. For this study, the staff loops of G2 and G3 were connected so that any available staff token could service any G2 or G3 task. The result of this increased staff flexibility is shown in Figure 13 for a staff multiple of two (a total of four G2 and G3 staff tokens).

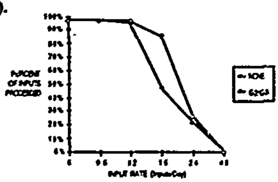


Figure 13. Staff Cross Training Results.

The bottom curve represents the case shown earlier with no cross training. The top curve shows the efficiency improvement realized by making the same number of staff members available to perform a greater variety of tasks. Again, the result is a further movement of the knee of the curve to the right.

STUDY CONCLUSIONS

The purpose of this study was to develop a prototype Petri net simulation of the Army division main command post and to examine its sensitivities to automation, contingency planning, and staff size and flexibility. The results reveal significant sensitivities in all three areas. Shortening selected task times (automation) in a single staff produced minimal payoff due to the data dependencies in the estimation process. Automation of all staffs equally produced better results, but the degree of automation must be considered carefully. The effects of specific automation candidates were not examined in this study. A high payoff was found for written contingency plans while the effect of implicit (known only to the commander) contingencies was minimal. Increased staff size and flexibility also produced a high payoff under high workload conditions.

This study was a first attempt at simulating a real headquarters process using IDEF and Petri net modeling techniques. The results are encouraging but, by no means, conclusive. The models used, while based on official Army training documents, have yet to be validated with experiment or exercise data. Efforts are now under way to begin this validation with the help of Army personnel. It is hoped that these efforts will produce a more realistic model with more precise time estimates. Such a model would be a powerful tool for setting up experiments and developing measures of efficiency for the Army division command post operation.

REFERENCES

1. Army Training and Evaluation Program, Mission Training Plan for Division, ARTEP 71-100-MTP, Hq Department of the Army, Initial Coordinating Draft.

2. Information System Projects Support (ISPS) Program, BCTP Support Operations Comprehensive Plan Addendum, ARTEP Task Analysis, RDA-TR-144800-006 (Preliminary Draft - advanced copy).

3. Staff Organization and Operations, FM 101-5, Hq Department of the Army, Washington, DC, May 1984.

4. HEAT Users Manual, Defense Communications Agency, Prepared by Defense Systems, Inc. under contract #DCA-100-84-C-0047, July 1984.

5. A Guide to the Estimate of the Situation, CGSC Student Text 100-9, U.S. Army Command and General Staff College, Fort Leavenworth, KS, July 1985.

6. AUTO IDEF Users Manual, TR-358, ALPHATECH, Inc., 111 Middlesex Turnpike, Burlington, MA 01803, Prepared under contract to Armstrong Aerospace Research Laboratory, Contract No. F33615-86-C-0545, October 1987.

7. Systematic Evaluation of the Air Defense Mission, TR-284, ALPHATECH, Inc., 111 Middlesex Turnpike, Burlington, MA 01803, Prepared under contract to BDM Corporation, McLean, VA, Subcontract No. BDM-S562-0300512, Prime Contract No. DCA100-85C-0063, April 1986.

8. Micro-Modeler User's Manual, TR-358, ALPHATECH, Inc., 111 Middlesex Turnpike, Burlington, MA 01803, Prepared under contract to Air Force Foreign Technology Division, Contract No. F33657-81-C-2150, March 1987.

9. Technical Report CAORA/TR-13/85, ACN 70376 G3 Analysis, US Army Combined Arms Operations Research Activity, Studies and Analysis Directorate, Vol I Main Report, Vol. II. Appendices, Fort Leavenworth, KS.

APPLYING
THE MODULAR COMMAND AND CONTROL EVALUATION STRUCTURE (MCES) TO CONCEPT DEFINITION

DR. RICKI SWEET

SWEET ASSOCIATES, LTD.

The goal of this analysis is to consider command and control needs in military operational management with respect to global-scale warfare. The challenge is to define an operational concept for C2 beyond the current purview of the Specified and Unified Commands. The analysis addresses the question: "What command and control architecture is required to provide the necessary responsiveness to national level political/military objectives?" The emphasis is on the required command and control architecture, both for the effective management of a global level conflict and for the responsiveness to national political/military objectives.

From a methodology viewpoint, this analysis represents the application of the MCES to command center conceptual definition. It provides the requisite analytic support to formulate the system requirements that lead to instantiation of the concept. In contrast to the focus upon command and control architectures as system and technology driven, this analysis emphasizes support to the decision makers in the command and control process. Nevertheless, it is intended that the SuperCINC work will lead to the identification of information/equipment/C2 support required at the systems level to support NMCS / SuperCINC.

The goal of the SuperCINC analysis is to consider command and control needs in military operational management capability with respect to global scale warfare. Specifically, these deficiencies arise from the lack in capability to support the management of war with several simultaneously occurring crises at different levels going on globally as well as potentially within a single theater. Thus, the management challenge is to define an operational concept for C2 beyond the current purview of the Specified and Unified Commands.

This problem was addressed under funding from McDonnell Douglas Corporation, Washington Studies and Analysis Group. The analysis addresses the question: "What C^2 architecture is required to provide the necessary responsiveness to national political/military objectives?" The C^2 architectural concept defined to enhance the organizational capability for such management must ensure effective warfighting capability and responsiveness to the designated objectives.

In contrast to the focus upon C^2 architectures as system and technology driven, this analysis emphasizes the decision makers in the C^2 process. Still, it is intended that this analysis will lead to the identification of information/equipment/C2 support required at the systems level to support NMCS, both without and including the SuperCINC.

From a methodological standpoint, the SuperCINC work represents the application of the MCES to command center conceptual definition. It provides the analytic support to specify the system requirements which will lead to instantiation of the concept.

C^2 CONCEPTS FOR SUPERCINC ARCHITECTURE

As a result of this analysis, several issues have been identified relating to the supporting C3I for the SuperCINC architecture. The C^2 concepts that have surfaced are: (1) the command authority of the SuperCINC; (2) the location, classification and evolution of SuperCINC supporting C2 Nodes; (3) customized interfaces for C^2 decisionmakers; and (4) level of detail of required information. The first of these concepts is the guide to the definition of the required organizational architecture for SuperCINC.

1. The command authority of the SuperCINC leads to several axioms upon which the systems level architecture must be built.

   - The supporting command center(s) will be permanent.

supporting C3I for the SuperCINC architecture. The C^2 concepts that have surfaced are: (1) the command authority of the SuperCINC; (2) the location, classification and evolution of SuperCINC supporting C2 Nodes; (3) customized interfaces for C^2 decisionmakers; and (4) level of detail of required information. The first of these concepts is the guide to the definition of the required organizational architecture for SuperCINC.

1. The command authority of the SuperCINC leads to several axioms upon which the systems level architecture must be built.

   o The supporting command center(s) will be permanent.

   o Functional operational authority is delegated by NCA under pre-determined activating conditions, which are yet to be specified.

   o The nature of this authority and especially its limits must be determined.

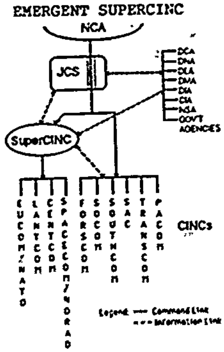
     -Five star permanent authority with respect to all CINCs.

     -An "Emergent SuperCINC" with authority over specified CINCs, with said authority itself contingent upon a set of conditions being satisfied, see Fig. 1.

2. The location, classification and evolution of supporting C2 Nodes are derivative of the SuperCINC concept development. Consideration of these concepts result in the following limiting conditions:

FIGURE 1.

EMERGENT SUPERCINC



O Location.

- SuperCINC, as an entity, must be located at a major information node.

- This node must be able to provide, in equal detail, information from a number of different commands. Accordingly, topics to be resolve include:

  o Security of information,

  o Vulnerability of that information to C3CM,

  o Redundancy, and

  o Recency.

- The desideratum is to locate the C2 Node as close to the central geographical region of interest as possible. However, this may not be possible. The location is dependent upon:

  o Availability of the most technologically appropriate command center, and

  o Availability of the most survivable, reliable communication links.

O Classification.

There is a direct relationship between the permanency of authority and the type of supporting C2 Nodes that must be available, e.g., the "Emergent SuperCINC" must be supported by a Specific SuperCINC C^2 Node (SSC2N). Feasible combinations of C2 nodes can be postulated between sets of information to handle the C2 requirements. The resulting information synthesis should be located at the most adequately equipped C2 node, which may thereafter be enhanced. For the Universal SuperCINC, a C2 node must be identified, together with alternatives, based upon such topics as succession of command, redundancy and survivability of functionality.

- The Candidate C2 Nodes may be classified according to employment concept

  - o The Specific SuperCINC has command authority only with respect to the CINCs where the scenario overlaps their mandate.

  - o The Universal SuperCINC has command authority with respect to all the CINCs.

- o Evolution.

  Since the SuperCINC concept has never been a design concept, existing command centers and C2 nodes are not likely to meet all requirements.

  - o Existing and programmed C2 systems viewed in this perspective, result in residual requirements which must be remedied by upgrading existing systems to implement the SuperCINC supporting C3I system.

  - o The final SuperCINC architecture and its supporting C3I operational systems will evolve through conceptual, development, test, deployment, or perhaps operational phases.

  - o This evolution is more likely to be built if, after identifying the long-term goal, we begin to implement changes in operational and organizational concepts, information flow patterns, employment concepts and the like.

  - o These relatively immediate changes will be followed by the longer lead time aspects of developing and acquiring supporting hardware and software.

  - o At each stage of this evolution, it is critical that the criteria for evaluation evolve with the SuperCINC architecture and its mission.

3. "Customized interfaces" for C^2 decisionmakers.

   In order for the SuperCINC to operate effectively and efficiently in the fast moving and information rich environment, an atmosphere which enhances personal decision making style must be available. This atmosphere has begun to be referred to as customized interfaces for individual use.

- o These interfaces may be related to all aspects of the C3I system's static components. These make up the physical entities of the system, namely, hardware, software, people, and the facilities that house them.

- o Of particular note is the question, "To what and in what specifics does the decisionmaker need access? Concerns to be addressed include:

  - Secure, reliable, enduring communications to superordinate, lateral and subordinate organizations;

  - Conferencing capabilities within the command center; and

  - Idiosyncratic needs, vis-a-vis visual display.

4. Level of detail of required information.

   Finally, there are two competing issues that effect the utility of information within the C2 process.

- o Information is and must continue to be made available to the decisionmaker at a number of different levels of detail.

  - This requirement results in greater dimensionality than is readily usable at a single time by a single decisionmaker.

- It must be provided in terms of potential flexibility to enhance the information base upon which the selection and execution of options is grounded.

O We have long advocated a minimum essential approach to providing information to the decisionmaker. However, if one looks across time and at these idiosyncratic needs, one has a major design issue with which to grapple.

- We suggest that a suitable approach is to build upon the information needs derived from decisions required during the accepted operational scenario.

- The individual SuperCINC in conjunction with his direct supporting staff will determine what information is to be pulled from the data sources and the form in which it is to be presented for decisionmaking.

O The second issue relates to who is to provide the minimum essential information to the decisionmaker. It is likely that in the process of selecting the required minimum essential information elements, a filter is introduced which emphasizes a particular set of conclusions.

- If the supporting staff is known to or believed by the decisionmaker to be credible, the pre-filtering is taken as appropriate.

- If the decisionmaker does not feel that the data presented to him is credible, for whatever reason, he will often request access to a greater level of detailed data than in that supplied.

- Clearly, to the extent that he must access this more detailed data, the less time he has to engage in the other equally demanding aspects of his job.

INSIGHTS FROM THE APPLICATION OF THE MCES

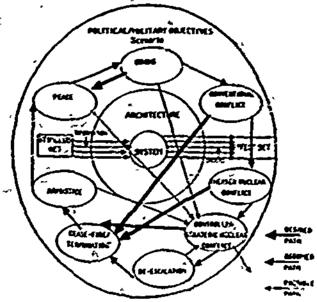
The SuperCINC study is a top-down analysis focussed upon organizations, their functions, facilities and procedures. The analysis draws its direction from the decision-makers' objectives. These in turn are derived from the ultimate military/political objective: deterrence.

PROBLEM SPECIFICATION AND SCENARIOS: MODULE 1.

First, the representative conflict scenarios from crisis to global nuclear war were described, see Fig. 2. The phases of

FIGURE 2.

POSSIBLE SCENARIO SEQUENCE



the scenario are seen from the general military perspective as sequential in nature. Certain paths through these phases may be termed "desired paths". These inevitably lead to a de-escalation in the level of conflict. It is the effectiveness of a given architecture in maximizing these desiderata that is an ultimate essential component of the evaluation.

The scenario not only reflects the decision-makers' objectives, but also provides the set of events and decisions which must be supported for the SuperCINC Architecture to carry out the chosen mission. Thus, the concept defined for the SuperCINC architecture is based upon scenario-driven decision support requirements, namely: information needs specifically engendered by pre-real time planning as well as the time sensitivities implicit the operational mission. The required information needs of the SuperCINC relate to: Capabilities (Plans, Resources, etc.); Intelligence Assessments (Threat, Response); and Foreign Country Issues. This focus highlights critical information needs and the corollary minimum essential elements for architectural specification.

CONCEPTUAL ALTERNATIVES AND STATICS: MODULE 2.

In the system bounding of the SuperCINC study, specific attention was given to organizational structure, information flow, and specific design dimensions, see Fig. 3.

ORGANIZATIONAL STRUCTURE. Three conceptual SuperCINC alternatives were developed. Each was capable of providing the necessary C2 structure in support of operational responsibilities above the CINC level, involving the integrated activities of several CINCs. This role may be constituted to correspond to that of military CINC of CINCs. The candidate organizational structures, dependent upon who the SuperCINC is, were described:

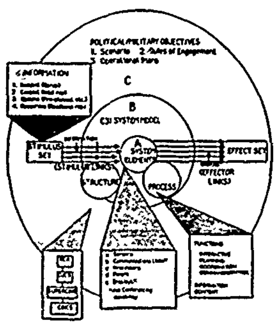
(1) JCS, as a whole acting as SuperCINC,

(2) A member of JCS acting as Single SuperCINC, and

(3) Another organizational entity acting as Single SuperCINC. This non JCs Single SuperCINC may be one of the ten CINCs of the Unified and Specified Commands, a senior military officer, active or retired, or a designated/appointed SuperCINC from among a set of suitable candidates with the appropriate military background.

FIGURE 3.

C^2 SYSTEM BOUNDING



INFORMATION FLOW. The information flow requirements may be seen as arising out of the relationships between the SuperCINC and other specified organizations.

The NCA. The SuperCINC must respond to the NCA by filtering the large amount of information, providing a concise summary of events, producing options for the NCA and responding to the queries of the NCA.

Internal to the SuperCINC. The SuperCINC requires information to generate options for the NCA. The perspective must be global in nature and support national level goals. This information must be accessed from the individual CINCs, the services, and a wide variety of government agencies including the intelligence agencies.

The CINCs. - The CINCs requires not only direction and authority but some perspective of the global nature of the conflict. In some cases, direct cross-feed of information between CINCs may be required to ensure that actions by one CINC do not directly conflict with actions of another. Additionally, direct contact with the Allies takes place at the CINC level and requires the CINC's ability to coordinate within his theater of operation from a global perspective.

The services. As suppliers of men and materiel to the CINCs, the services require information to ensure that the needs of the CINCs will be met under a fast breaking scenario. Additionally, the services must provide, through the SuperCINC, accurate information on capabilities.

DESIGN DIMENSIONS. Information flow requirements for SuperCINC must be based upon the architectural design decisions made during conceptual definition. For our purposes, these design issues relate to the interdependent dimensions: organizational; geographical or spatial; and temporal.

Organizational. The SuperCINC has been taken to represent a new level of military organization, below the JCS, which has operational authority in conflicts which are likely to involve operations that transcend the currently defined purviews of the Unified and Specified Commands.

In both the "SuperCINC as a group of decisionmakers" and the "Single - JCS" roles, SuperCINC is in a direct line of authority between the NCA and the CINCs. For the former, information is received from a variety of information sources. In the latter, the JCS plays a conduit role between the information sources, the SuperCINC and the CINCs.

For the "Single SuperCINC - non-JCS", there are direct information links to SuperCINC from the JCS and the information sources.

Geographical. - If the SuperCINC commands several CINCs only, then the geographical location of each CINC's area of responsibility (AOR) and the relationship to the others' become critical considerations. The AORs may be geographically contiguous, but this is not necessary.

The SuperCINC may have command authority only over the CINCs where the scenario in effect overlaps their mandate; then the NCA maintains its command authority over all other CINCs.

When command authority for specific CINCs is delegated by NCA to SuperCINC, JCS continue to act in it current role.

Under some circumstances, it may be assumed that the SuperCINC will have command authority over all the CINCs.

Temporal. - The point at which the SuperCINC assumes command is unlikely to be totally event driven.

The rationale is that it would be extremely difficult to orchestrate events, if a given pre-determined action triggers the SuperCINC. Therefore, the SuperCINC should be in play from the beginning of hostilities at any level.

In contrast to this view, the SuperCINC could be postulated to rise to operational command based upon a pre-determined set of criteria well before the beginning of hostilities.

SUPERCINC FUNCTIONALITY: MODULE 3.

The functional aspects of C3I System Model of the appropriate military organization(s) to remedy the deficiencies were grouped into: C^2 Functions; Interactive Functions, external and internal; Planning Functions; and Coordination Functions.

INTEGRATION OF STATICS AND DYNAMICS: MODULE 4.

The integration of statics and dynamics is, in this analysis, equivalent to the definition of the conceptual architectural alternatives. The initial specification is provided in Module 1. This is refined in Module 2 and finalized for the first iteration of the evaluation process in Module 4, see Fig. 4.

The conclusion of any MCES analysis is assumed to have three possibilities: implementation of results, do nothing, or reanalysis. If reanalysis is the outcome, the architectural concept definition will continue to be refined, leading ultimately to the devvelopment of the concept preparatory to the design and acquisition stages of the PPBS.

SPECIFICATION OF MEASURES: MODULE 5.

Measures or Evaluative Criteria, derived from mission needs and independent of specific concepts, were identified and explicated. They may be used to evaluate the capability of candidate Architectures for SuperCINC to remedy the identified deficiency in terms of the mission of managing global scale warfare. The measures are intended to provide values which allow the determination of significant increases in performance, effectiveness as well as policy and force effectiveness.

The Measures of Performance (MOPs) or - equivalently - the A or Sub-System Level Evaluative Criteria can be divided into three categories; those that measure:

0 Component performance characteristics;

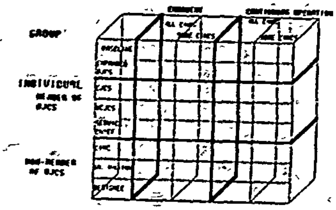
0 Characteristics of the information flow in the C^2 system; and

0 Design characteristics.

The Measures of Effectiveness (MOEs) or - equivalently - the B or System Level Evaluative Criteria assess the extent to which the system performance meets the requirements for a given scenario or set of scenarios. Three classes of MOEs have been defined that are applicable to different aspects of the SuperCINC architectures. They deal with:

FIGURE 4.

ARCHITECTURAL DIMENSIONS



- 0 Errors, both static and dynamic;
- 0 Adaptiveness imbedded in the system architecture; and
- 0 Ability of the system to evolve over time in view of logistics, acquisition procedures, and changing technology.

The Measures of Force and Measures of Policy Effectiveness (MOPEs and MOPEs) or equivalently - the C or Architectural Level Criteria assess the effect of SuperCINC on overall military effectiveness.

CONCLUSION

There is no current implementation of the SuperCINC concept. However, current requirements for command centers are being developed to support a variety of potential missions. This requirement to include command center mission flexibility makes the C2 node to support a SuperCINC a feasible realization.

Since the evaluation approach is based on executing a scenario, an adaptation of a war gaming exercise would be the appropriate means for eliciting expert judgement. Estimates will have to be made both by expert commanders as to what their needs would be in responding to the stimuli produced by the scenario and by expert staff as to what the attributes of the various processes would be with current comparable equipment or with planned enhancements.

Under the information intensive and fast moving conditions of the global-scale war scenario, NCA is providing policy, oversight, guidance, and direction functions. Therefore, the delegation of command authority by NCA to another SuperCINC is likely. In such a context, day to day resource allocation and war-fighting management, as well as other selected functions, are delegated to a subordinate level SuperCINC in order to effectively and efficiently provide more flexible mission oriented alternatives.

Reviewing the architectural dimensions proposed in this study, it is easily seen that some of these dimensions represent more feasible architectures than do others. Accordingly, the determinants of the identified architectures for subsequent analysis are temporally, geographically and organizationally based.

1. Temporal.

   The command authority concept is the guide to the definition of the required architecture for SuperCINC. The functional operational authority which is delegated by NCA to the SuperCINC may be classified in terms of a <u>temporal</u> dimension. It may be either continuous or emergent, that is, activated according to a set of as yet unspecified pre-determined conditions. The continuous or permanent authority is most likely to be at the five star level.

2. Geographical.

   Two <u>geographically-related</u> conditions are possible, for either the permanent or the emergent SuperCINC, but not equally likely. On the one hand, under the universal classification, this authority will be extended over all CINCs. On the other hand, the authority will be extended over that sub-set of the CINCs whose AOR is operant with respect to the scenario in play. This latter option leads inevitably to resource allocation conflicts between the CINCs included and those excluded from the SuperCINC command authority. Accordingly, it is judged that this latter option is a less likely concept than the former.

   Other geographic factors relate to the location of thehe SuperCINC supporting C2 Nodes.

   The C2 Nodes must be located at a major informational node with timely, reliable, secure, and redundant communication capability.

   This node must be able to provide, in equal detail, information from a number of different commands.

The C2 Node should be close to the central geographical region of interest.

The location is dependent upon availability of the technologically appropriate command center.

3. Organizational.

The options available were: Specific SuperCINC C^2 Node (SSC2N) and the Universal SuperCINC. According to employment concept, the Specific SuperCINC has command authority only with respect to the CINCs where the scenario overlaps their mandate. The Universal SuperCINC has command authority with respect to all the CINCs.

In addition, there is a direct relationship between the permanency of authority and the type of supporting C2 Nodes that must be available. For example, the command authority derived organizational architecture definition of SuperCINC points out that due to the necessity of an appropriate readiness posture, the supporting command center(s) will be permanent.

In summary, the selected architectural alternatives to be subsequently analyzed are:

1. Functional operational authority which is continuous and with authority over all CINCs.
2. Functional operational authority which is emergent and with authority over all CINCs.
3. Functional operational authority which is continuous with authority over that sub-set of the CINCs whose AOR is operant with respect to the scenario in play.
4. Functional operational authority which is emergent with authority over that sub-set of the CINCs whose AOR is operant with respect to the scenario in play.

In order to refine and quantify the architectures, scenarios, and the evaluative criteria, five tasks must be completed: identification of the joint operational doctrines to be associated with the selected architecture(s); expansion of the set of evaluative criteria; application of the evaluation approach to the candidate architectures; specification of SuperCINC requirements; and identification of the conditions for employment of the selected architecture.

Their satisfactory conclusion will provide the basis for an implementation plan to remediate the problem of managing global scale warfare. Thereafter, priorities for the program elements which will address the required fixes may be set and the necessary resources mobilized. Finally, a sequenced set of programs, phrased in terms of several evolutionary contexts, must be developed. These contexts focus the evaluation upon how well the SuperCINC functions, at the various stages of the military system life cycle.

REFERENCES

Rona, Thomas P. "Weapon Systems and Information War", Boeing Aerspace Company, July 1976.

Rona, Thomas P. "Air C^2: The New European Perspective", Boeing Aerospce Company, December 1983.

Sweet, Ricki and Levis, Alexander H., "SuperCINC Architecture: Command Structure For Global-Scale Warfare, Final Report", Sweet Associates, Ltd., December 1987.

Sweet, Ricki and Levis, Alexander H., "SuperCINC Architecture Concept Definition and Evaluation: An MCES Application", SIGNAL, July 1988.

Sweet, Ricki with Mensch, Dennis, Gandee, Pat, Stone, Ingabee, and Briggs, Kevin, "The Modular C^2 Evaluation Structure (MCES): Applications Of and Expansion To C3 Architectural Evaluation", Naval Postgraduate School, September, 1986.

USE OF THE IFFN TESTBED FOR EVALUATING AIR C^2

James E. Haile

Identification Friend, Foe, or Neutral

ABSTRACT

The Identification Friend, Foe, or Neutral (IFFN) Joint Test Force (JTF) located at Kirtland AFB, NM, has developed a testbed that is composed of high fidelity, real time man-in-the-loop simulators designed to replicate the NATO Central Region Integrated Air Defense System. The purpose of the test is to assess the ability of this air defense system to correctly identify and engage enemy aircraft. The testbed represents the largest real time command and control (C^2) simulation which consists of 59 medium and high fidelity tactical consoles and over a million lines of code. The OSD-sponsored testbed development and test is scheduled to run through July 1989. After this the testbed will become the Theater Air Command and Control Simulation Facility (TACCSF) operated by the USAF Tactical Air Warfare Center. The facility will be used by both Army and Air Force commands to resolve joint operational issues and support acquisition requirements.

INTRODUCTION

One of the most serious problems facing the NATO air defense forces is the correct and timely identification of aircraft in a tactical air environment which includes large numbers of friendly and hostile aircraft, electronic warfare threats, and surface-to-air, and air-to-air missiles that operate beyond visual range (BVR). Numerous studies have concluded that the current electronic identification capabilities are too slow, individually unsuitable for positively identifying hostiles and friends, have insufficient range, and are subject to interference from electronic countermeasures and other environmental factors.

Using real time, man-in-the-loop simulations of Army, Air Force, and NATO C^2 and weapons systems, IFFN was designed to: assess the baseline capabilities within the NATO Integration Air Defense C^2 system to perform the identification function; identify deficiencies in the performance of the identification function; and, define potential near-term procedural and equipment changes to overcome the deficiencies.

TESTBED DESCRIPTION

The overall operational scope of the test is summarized in Figure 1. It focuses on the NATO Integrated Air Defense System within the Lauda Battle Management Area. Consistent with defense planning guidance, a standard 14/10 air scenario is used with the NATO air defenses defending against a Warsaw Pact three wave attack. Although NATO offensive air and support operations are not represented with manned simulators, they are explicitly flown in the air scenario. In addition, the operations focus on a 1989 time frame using current AAFCE air defense plans and air space control procedures.

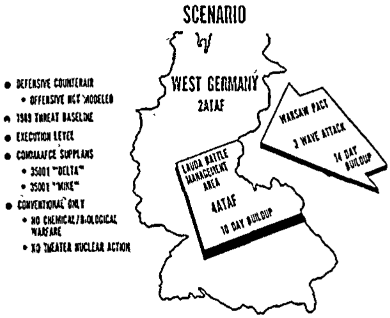


Figure 1

The overall testbed configuration representing the Lauda Integrated Air Defense Systems is depicted in Figure 2. There are three types of weapons systems: PATRIOT, HAWK Army surfaceto-air missile systems, and the Air Force F-15. Associated with these weapons are their respective command, control, and communications (C^3). In the case of the PATRIOT and HAWK there are battalion (BN) fire direction centers (FDCs) with a brigade FDC (TSQ-73) controlling the two BNs. For the F-15's, the primary C^2 is a CRP (Modular Control Equipment) and a NATO E-3A. Controlling the overall Army and Air Force is the NATO CRC (German

Air Defense Ground Environment equipped). Every node has its appropriate voice and digital communications replicated. A total of 59 high and medium fidelity simulators comprise the testbed. The high fidelity simulators are used for data collection and are run by actual field operators. These are depicted in Figure 2 by the large boxes. The lower fidelity simulators are operated by trained technicians and are used to load the high fidelity simulators. These are represented by the smaller boxes.

Overall Test Configuration

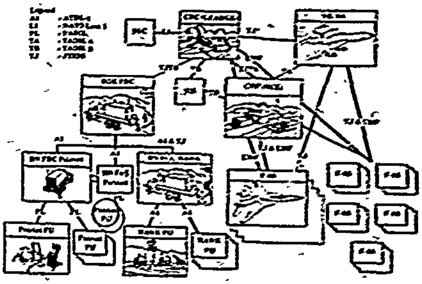


Figure 2

The overall testbed design is depicted in Figure 3. A central simulation system is used to generate the comprehensive air scenario with appropriate tactics, airspace control procedures, and terrain. This scenario is then provided to four different types of simulation through a communications subsystem. These simulators are: PATRIOT, HAWK, F-15, and air C^2. All simulators are located at Kirtland AFB, NM, with the exception of the PATRIOT which is at Ft Bliss, TX.

TESTBED DESIGN

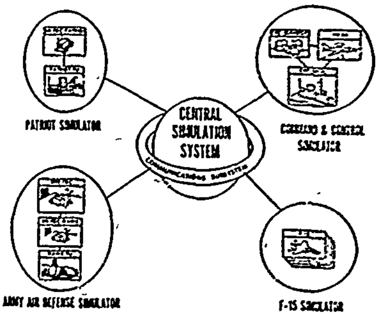


Figure 3

Figure 4 depicts the PATRIOT Tactical Operations Simulators at Ft Bliss, TX. The consoles depicted replicate a PATRIOT fire unit; another is used to replicate the PATRIOT BN FDC. These high fidelity systems are used by the Army for prototype software development.

PATRIOT SIMULATOR

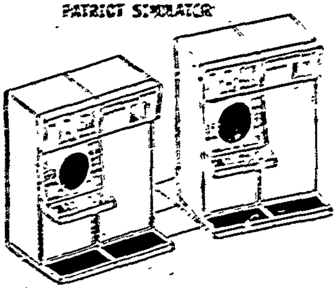


Figure 4

Next in Figure 5 is a picture of the HAWK fire unit simulator. All key functions of the HAWK system are replicated and eight are used in a BN.

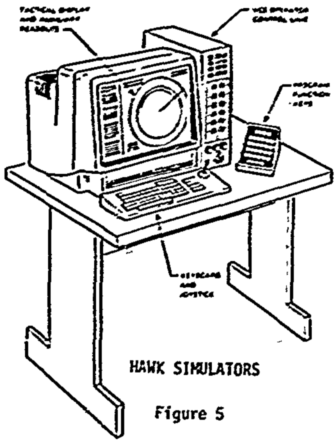


Figure 5

Figure 6 is the F-15 simulator. It uses an actual F-15 stick and throttle (MSIP) with appropriate switches. The upper screen is the F-15 heads up display which projects airspeed, altitude, attitude, radar target box, and missile parameters. The lower screen includes all the flight instruments, radar, Electronic Warfare Warning System (EWWS), Tactical Electronic Warfare System (TEWS), and Joint Tactical Information Distribution System (JTIDS). This flight simulator is unique in that hundreds of friendly and enemy tracks can be presented to the pilots in a realistic jamming environment.

F-15 SIMULATOR

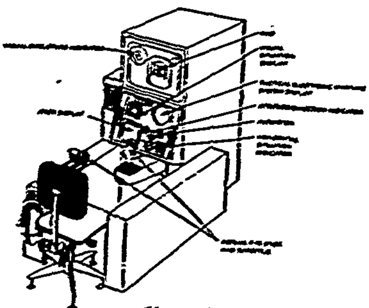


Figure 6

Figure 7 is the IFFN tactical command and control display (TCCD). The right screen is a touch screen that commands system functions. The left display is in color and portrays appropriate air situations. The special function keys on the left are software reprogrammable and are set up with a template. Thirty of these TCCD's are used in the IFFN testbed.

TACTICAL COMMAND AND CONTROL DISPLAY

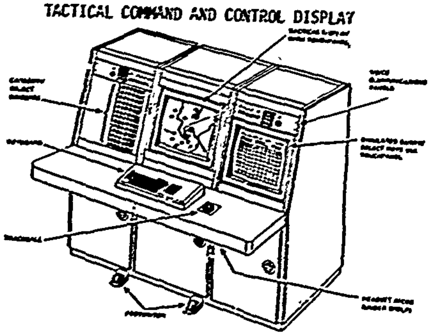


Figure 7

Figure 8 portrays the two simulation rooms in the IFFN testbed. Situated in the Air Command and Control Simulation room are the 30 TCCD consoles which are configured as follows: 5 for the NE-3A, 8 for the CRP, 12 for the CRC, 2 for the HAWK BN FDC, 2 for the Brigade FDC, and 1 for the Sector Operations Center. All of these consoles are software reprogrammable so other configurations could be addressed. The fire unit simulation room contains: 4 F-15's, 8 HAWKs, 6 PATRIOT test loaders, 5 F-15 test loaders, and 2 test loaders for the special information systems. In addition, there are the four consoles at Ft Bliss for the PATRIOT fire unit and BN FDC.

IFFN TESTBED

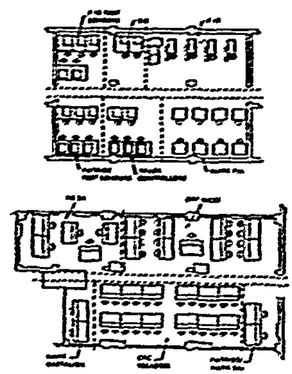


Figure 8

Figure 9 summarizes some of the overall testbed features that makes it the largest air C^3 simulation in the world.

TESTBED FEATURES

- 59 HIGH TO MEDIUM FIDELITY CONSOLES
  - SIMULATING 12 TYPES OF TACTICAL SYSTEMS
  - USING 7 TYPES OF DATA LINKS
- 1.2 MILLION SOURCE LINES OF FORTRAN CODE
- COMPUTERS
  - 18 CONCURRENT 3250/60/65 COMPUTERS
  - 21 ARRAY PROCESSORS
  - 462 MILLION INSTRUCTIONS PER SECOND CAPACITY
- EXERCISE CAPACITY
  - 2000 X 2000NM WITH DMA TERRAIN DATA
  - 2000 ACTIVE AIRCRAFT
  - 128 MAXIMUM EXERCISE PARTICIPANTS

Figure 9

IFFN TESTING

The IFFN JTF to date has conducted three operational tests involving the PATRIOT, PATRIOT BN FDC, and the F-15. Five more operational tests are scheduled through 1988 with the HAWK, HAWK BN, Brigade FDC, NE-3A, and CRP. The grand finale is the ninth test in which the CRC controls all the fire units and subordinate C^2 nodes. This testing will show how each element with its associated communications both individually and collectively contributes to the air defense process. Testing to date is showing that complete situation awareness is difficult to obtain, there are limitations in correctly identifying aircraft, and there is a clear need for integrated Army/AF/NATO operational testing and training under simulated wartime conditions.

FOLLOW-ON TESTING

In 1989, the IFFN JTF will complete its OSD directed testing. At that point, the USAF Tactical Air Command (TAC) and Army Training and Doctrine Command (TRADOC) have agreed to assume ownership of the testbed. TAC will be the executive and the name of the facility will change to Theater Air Command and Control Simulation Facility (TACCSF). Users will be the Army, Air Force, OSD, and NATO. The new mission will encompass:

- Concepts, tactics, and procedures development
- Developmental and operational testing
- C^2 systems integration and training.

In 1990, four tests are planned for TACCSF. The test issues have major operational significance to an air defense in the NATO Central Region and directly involve C^3. The four tests are:

- Alternate F-15/HAWK/PATRIOT employment options
- F-15 JTIDS Net Management
- MCE CRC/CRP Employment concepts
- F-15 and PATRIOT protection of high value airborne platforms.

In 1991, TACCSF is planning to conduct additional tests which are:

- Radar improvements and addition of ESM to E-3
- Forward area air defense (FAAD) C^2 connectivities
- Implications of MX XV.

These issues require some enhancements to the testbed; hence, they have been scheduled for the second year of TACCSF.

Other C^2 test issues that are being considered for 1992 are:

- Alternate airspace control procedures for new C^2 systems
- Anti-tactical ballistic missile C^2 architecture
- C^2 procedures for the attack of time sensitive ground targets
- C^2 in support of offensive air missions
- Passive sensor systems for the CRC.

There is a general agreement that for TACCSF to realize its full potential as a NATO air C^2 testbed it must add simulators for portions of the offensive air operations and C^2. This will enable TACCSF to explicitly address the new NATO ACCS concepts of combined offensive/defensive air operations.

SUMMARY

In summary, the IFFN testbed development is on schedule and demonstrating viable capability. IFFN test results to date have produced significant operational insights that are being addressed by the Army, Air Force, and NATO operational communities. The Army, Air Force, and NATO see great benefits in using the TACCSF to investigate a host of very complex C^3 issues starting in 1990.

A Testbed Activity to Enhance C3 Support to Crisis Management

C. J. Pasquariello, Center for C3 Systems (C4S),
Defense Communications Agency (DCA)

S. H. Starr, Washington C3I Division,
The MITRE Corporation

Abstract

It has long been recognized that our current C3 support to crisis management operations is subject to significant limitations (e.g., limited flexibility, responsiveness, and capacity). Although programmed C3 systems should ameliorate some of those deficiencies, these enhancements are being pursued on a piecemeal, fragmentary basis. Consequently, there is a need for a systematic, integrated approach to address residual crisis management issues by adapting emerging technologies and applying new system concepts.

In recognition of this need, DCA has initiated a testbed activity to enhance C3 support to crisis management. The testbed activity will draw upon and orchestrate a broad spectrum of C3 evaluation tools to address key issues. These include computer models, manned simulator facilities, exercises, and command center evaluation tools. In order to minimize the resources and time associated with this activity, every effort is being made to capitalize on existing and programmed tools.

This paper provides a context for the testbed activity by describing the crisis management process and trends that are affecting its evolution. It then identifies and discusses key C3 system issues associated with crisis management activities. Based upon the characteristics of those issues, a set of evaluation tools is identified and a concept is formulated for orchestrating those tools. The paper concludes with a discussion of the near-term steps that are being pursued in the program.

A. Background

As background for the paper, this section provides a definition of a crisis, introduces and discusses a conceptual model for a crisis, broadly describes the C3 system and process that support crisis management, and summarizes trends that are influencing the evolving system.

1. Definition

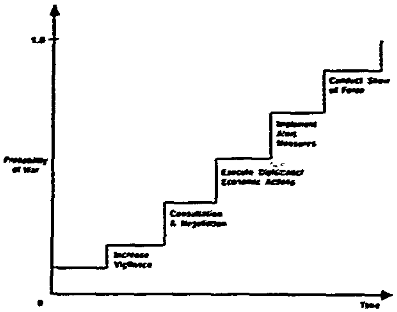
The spectrum of operational environments is frequently subdivided into five categories: day-to-day (or peacetime), crisis, conventional conflict, selective nuclear release, and general nuclear response. Consistent with JCS usage (reference 1), a crisis is defined as: ". . . an incident or situation involving a threat to the United States, its territories, and possessions that rapidly develops and creates a condition of such diplomatic, economic, political, or military importance to the U.S. Government that commitment of U.S. military forces and resources is contemplated to achieve U.S. national objectives."

2. Conceptual Model of a Crisis

In 1977, Tom Belden (reference 2) developed a conceptual model for indications and warning activities that provides a useful perspective for analyzing crises. Central to the conceptual model is the realization that analyses of crisis response should encompass the adversaries, their allies, and interested parties. The crisis "state" of any participant can be characterized by a "decision stairway" that depicts the participant's probability of war as a function of time.

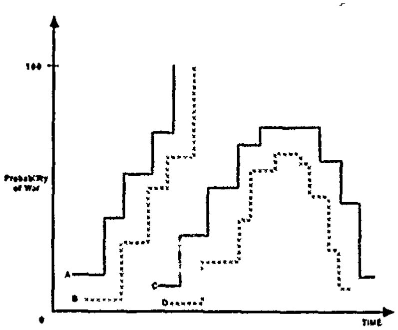
Figure 1 depicts such a decision stairway for a hypothetical situation in which a participant evolves from a day-to-day posture, through a crisis, to a military conflict. Transitions in the probability of war occur as a hypothetical event (e g., a border incursion) stimulates increased vigilance; adversary response prompts consultation and negotiation with allied foreign affairs and military authorities; increasing tension prompts the issuance of diplomatic responses (e g., sternly worded notes of protest) or economic actions (e.g., imposition of sanctions); the degenerating situation results in the implementation of alert measures (e.g., moving to higher levels of DEFCON); leading to a show of force (e.g., mobilization of ACE Mobile Forces to the troubled spot), and ultimately resulting in a transition to conventional conflict with the adversary. Note that transitions in the decision stairway can be prompted by political, economic, or military actions.

Figure 1. Stylized "Decision Staircase" for a Crisis Participant.



This technique can readily be extended to characterize and analyze complex, multi-participant crises. For example, Figure 2 conceptually characterizes the major interactions of the Arab-Israeli War in 1973, where the actions of the Arabs (participant A) stimulated responses from the Israelis (participant B), ultimately resulting in a conventional conflict. These events subsequently precipitated crisis responses in the United States (participant C) and the Soviet Union (participant D) as they monitored the evolving crisis and supported their national interests in the egion. The latter nations found themselves in an escalating crisis situation when the Israelis crossed the Suez Canel and threatened to annihilate the Egyptian Third Army. That secondary crisis was contained as the Arab-Israeli conflict ultimately transitioned to a stable cease fire.

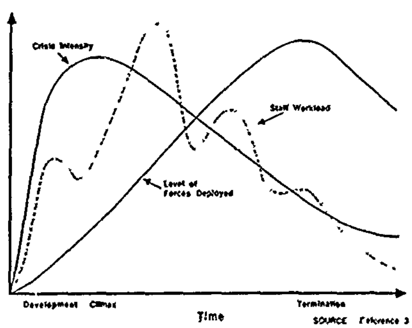
Figure 2. Simplified Decision Stairways for Four-Participant Crisis.



This perspective of the problem suggests that attempts to analyze broad C^3 support to crisis management must be able to deal with the dynamic interactions of the many participants involved in the multiple facets of a crisis (e.g., military operations, diplomatic actions, intelligence gathering).

Figure 3 provides a preliminary characterization of U.S. crisis behavior patterns, based upon analyses of ten major crises from 1958 (i.e., Lebanon) to 1975 (i.e., the Mayaguez) and six major exercises from 1973 (i.e., Night Train) to 1976 (i.e., Elegant Eagle) (reference 3). These results give rise to three major observations. First, the curve labeled "crisis intensity" suggests the general shape of the "decision stairway" for the sample set, beginning with a rapid buildup of the typical crisis which gradually tapers off as the crisis is resolved. Second, it is interesting to note that the level of forces deployed lagged the crisis intensity and tended to peak well after crisis intensity had slacked off considerably. An extreme example of this behavior was manifested in the Mayaguez crisis where Marines landed after the safety of the ship was no longer in doubt. Third, the oscillatory nature of the staff workload may be indicative of significant bottlenecks in the command and control process. Curves of this nature are frequently descriptive of a closed loop control system characterized by appreciable time delays and peak demands that exceed the instantaneous capacity of the system. It should be noted that the curve on staff workload was derived solely from exercise data because no comparable information was collected and maintained for actual crises.

Figure 3. Observed Crisis Behavior Patterns.

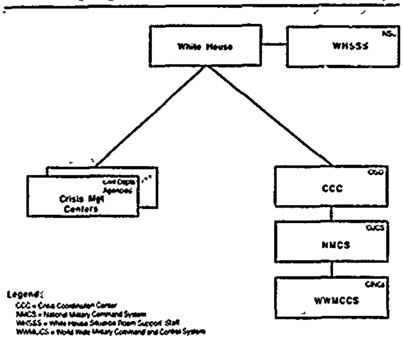


These observations suggest that crisis management measures of merit should be employed that characterize the intensity of the crisis, the timeliness of crisis actions, and the staff workload. In order to evaluate these measures of merit, evaluation tools are needed that reflect the capabilities and limitations of the C^3 system that supports crisis management actions and the process that these personnel employ.

3. C^3 System to Support Crisis Management

Since crisis management can require the participation of many U.S. government (and allied) organizations, a complex C^3 system has evolved over the years to support it. Figure 4 depicts a critical sub-set of that system. The system is headed by representatives of the National Command Authority (NCA) in the White House. They are provided operational support by the White House Situation Room Support Staff (WHSSS). Reporting to the White House is a chain of organizations from the defense community. They are led by the Secretary of Defense (or his representative) operating from the newly created Crisis Coordination Center (CCC) in the Pentagon. He is assisted by the Chairman, Joint Chiefs of Staff, and his supporting organization, operating out of the National Military Command Center (NMCC) in the Pentagon. They, in turn, are supported by the involved Unified and Specified Commanders-in-Chief (CINCs) operating from appropriate nodes of the Worldwide Military Command and Control System (WWMCCS).

Figure 4. Subset of Crisis Management System.

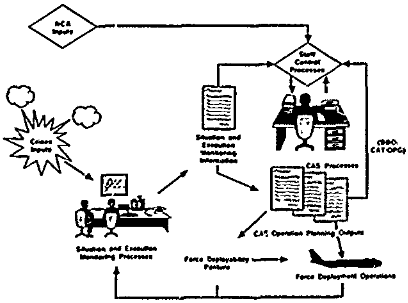


Since most crises have important diplomatic and economic ramifications, crisis management generally involves participants at crisis management centers supporting a broad range of civil departments and agencies (e.g., State, Treasury, Federal Emergency Management Agency (FEMA)). In addition, support is forthcoming from the intelligence community from sites such as Langley, Virginia, and Ft. Meade, Maryland.

4. Crisis Management Processes

Figure 5 provides a high level characterization of the processes that are performed at the NMCC in support of crisis management. Cells are established to support situation monitoring and execution monitoring. The situation monitoring process involves the monitoring and evaluation of worldwide events, military operations, and force status (e.g., friendly, other parties). Similarly, the execution monitoring process subsumes the monitoring/assessment of military operations in progress, the assessment of the adequacy of operations, and the response to requests (e.g., forces, other support). The information from these cells is provided to the crisis control cell and the crisis action system cell. The crisis control cell establishes crisis procedures and makes action assignments; monitors crisis processes, actions, and status; and prepares/presents options, plans, and requests to senior decisionmakers. In support of the crisis control cell, the crisis action system reviews and assesses the situation, preliminary deployment planning, and closure estimates; and prepares and reviews warning, alert, operations, and executive orders.

Figure 5. Crisis Action Processes.



5. Trends

There are a number of trends in the area of crisis management that affect the process and the system that is emerging to support it. First, there is an appreciation that the threat is growing. The volatility of world events gives rise to nearly constant tension and greater risk of terrorism and low intensity conflict. In response to that perceived threat, new policy and force structure is emerging. In particular, increased emphasis is being placed on the enhancement and use of Special Operations Forces. In consonance with these trends, the management of crises is becoming more complex. The increased scope of recent crises has involved the participation of additional players in both the civilian and military communities. In addition, in several recent crises (e.g., Iran hostage, Achille Lauro, Persian Gulf), the political ramifications and volatility have forced high-level decisionmakers to address extremely complex issues, frequently under very tight time constraints.

From a C^3 systems perspective, several trends have become apparent that will affect crisis management. It is widely recognized that the C^3 system that has been deployed to support crisis management has significant limitations (e.g., limited flexibility, responsiveness, and capacity--see Figure 3--). Several programs have been initiated to ameliorate those limitations, most notably the WWMCCS Information System (WIS). However, these programs have been beset with significant difficulties. WIS has experienced major delays and is undergoing a major management review. In addition, many of the enhancements that are being made to individual C^3 nodes tend to reflect a limited appreciation of the total crisis management problem and are being imperfectly integrated, coordinated, and evaluated. In spite of these obstacles, there is considerable optimism that the rapid emergence of new technology in the areas of processing, display, data base management, and decision aids has the potential to enhance C^3 support to crisis management.

B. Objectives and Guidance

This section outlines the objectives of the DCA testbed activity and describes the factors that are guiding and focusing the effort. The most significant of these factors include the management guidance that has emerged from recent senior advisory boards, programmatic guidance to minimize resource needs and time, and a perception of the key issues that remain to be addressed.

1. Objectives

In view of the trends in crises and the supporting C^3 system, DCA/C^4S is initiating a distributed testbed activity to adapt emerging technology and new system concepts for the improvement of crisis management. To achieve that objective, the testbed activity is being designed to realize four sub-objectives.

- **Rapid prototyping.** The testbed activity will provide a vehicle for rapidly developing and evaluating prototype systems to support crisis management. Emphasis will be placed on prototyping new, improved decision support systems to assist staff members in situation development, crisis assessment, course of action development, and execution.
- **Requirements refinement.** The testbed activity will provide a mechanism for enhancing communications between the operational and systems development communities. This will facilitate the translation of operational needs into technical requirements whose programmatic implications (e.g., resources, schedule) are more clearly understood.
- **Analytical evaluation.** The testbed activity will subsume the development of analytical tools that provide a means to evaluate the performance and effectiveness of existing and proposed crisis management procedures and supporting systems.
- **Test and evaluation.** The testbed activity will include test and evaluation activities to extend the findings of rapid prototyping efforts and to facilitate the transfer of this technology into fielded systems.

2. Management Guidance

These objectives have been selected to respond to management guidance that has emerged from four recent efforts: the Packard Commission (reference 3), the DCA Technology Conference (reference 4), the Defense Science Board (DSB) Task Force on C^2 Systems Management (reference 5), and a DSB Summer Study on Technology Base Management (reference 6). The Packard Commission was convened by President Reagan to explore options to streamline the acquisition of DoD systems. It recommended that DoD reduce the time and risk associated with acquisitions by expanding its use of rapid prototyping and acquiring systems in an evolutionary manner. The proposed testbed activity provides one vehicle to implement this philosophy.

The DCA Technology Conference was convened in the Fall of 1986 to identify technologies that promise to enhance substantively C^3 system capabilities. It was concluded that a distributed rapid prototyping testbed is feasible and desirable to support the development of C^3 systems that supported multiple C^3 nodes. The proposed testbed activity will draw on that technology to achieve its objectives.

The DSB Task Force on C^2 Systems Management issued a report in July 1987 to revisit key issues that it had assessed ten years earlier. It called out the need to develop an overarching DoD C^3 architecture, to pursue a well-focused command and control research program, and to extend efforts to exercise and evaluate C^3 systems in a mission context. With respect to the latter area, it specifically noted that "the combination of operational exercises and user-oriented testbed activities represent major opportunities to maximize current-system readiness and assess command and control performance." The mix of tools envisioned in the testbed activity echoes that philosophy.

The DSB Summer Study on Technology Base Management advocated 6.3A Advanced Technology Transition Demonstrations (ATTDs) as "an extension of the Packard Commission prototyping recommendation." The objective of an ATTD would be to demonstrate "proof-of-principle" in an operational vice purely laboratory environment. Programmatically, it is anticipated that a representative activity would last approximately three years at a total program cost of $10-100M. The proposed testbed activity may be an attractive candidate for an ATTD.

3. Program Guidelines

In order to minimize the time and resources for the crisis management testbed activity, the following guidelines have been adhered to in structuring the program. Initially, the activity will focus on core command center initiatives (e.g., enhancements to the WHSSS, CCC, NMCC). Preliminary activities will assist in refining system requirements and explore the feasibility of accelerated application of emerging technology. To minimize the possibility of duplication of effort, the testbed activity will be configured to tie in with the testbed programs of the Services and DARPA. In addition, it will complement existing crisis management activities in several areas. In support of plans and architectures (e.g., JCS C^2 Master Plan, Wide-Area Surveillance Architecture), the testbed activity will evaluate and validate proposed concepts. In conjunction with exercises and evaluations (e.g, JCS CPXs, OSD Serial Exercises), the testbed activity will identify new, improved decision support systems that can be assessed in the context of a quasi-operational environment.

4. Key Issues

Several key crisis management issues have emerged in recent years that will be addressed by applying the testbed activity resources. These issues can be aggregated into five major areas: data/data management, information systems, communications/ connectivity, decision aids, and systems evaluation.

In the area of data/data management, there are pervasive problems for all of the major crisis management nodes. It is recognized that the data required to support operators and decisionmakers is highly diffused and there is a need to retrieve, manipulate, and display it more effectively. In particular, there is a need for a data fusion capability that can consolidate/integrate available information into a complete, consistent, and coherent picture of friendly and adversary status.

In the area of information systems, most of the crisis management nodes lack effective automated message handling systems to format, catalog, or sort/route incoming or outgoing messages. In addition, all of the nodes lack required multi-level security features. This is particularly important for crisis management where selected information tends to be very sensitive and "close hold."

Problems in communications/connectivity are endemic in the crisis management community. As noted above, the sensitivity of selected data mandates the use of limited distribution communications to restrict its dissemination. In addition, there is concern that it will prove difficult to connect reliably and rapidly to a Joint Task Force that has been deployed to respond to a crisis. Overall, there is need for an overarching crisis communications architecture to specify the appropriate interfaces and information exchanges among the nodes of the system.

There is wide-spread interest in developing and evaluating decision aids to support personnel throughout the crisis management system. These decision aids have the potential to support many key tasks including data base management and display, assistance in selecting courses of action, and institutional implementation of selected courses of action.

Finally, there is a continuing need to evaluate the crisis management process from a total systems point of view, capturing the interactions of the key participants. Prior efforts have tended to view the problem from a fragmented, sub-systems perspective and have failed to identify and ameliorate many of the C^3 issues that have hindered actual crisis operations.

C. Testbed Activity Resources and Concept

Due to the breadth of issues associated with the crisis management area, it was rapidly concluded that no single tool would be adequate to satisfy the full set of program objectives. Consequently, an orchestrated set of tools was sought drawing on computer models, testbed facilities, exercises, and assessment tools. To the extent feasible, efforts were made to identify existing tools that could be modified to satisfy program needs. This section describes the key resources that were identified to support the program and the concept that has emerged to synthesize them into a consistent activity.

1. Key Resources

a. Computer Models

Computer models were sought to support two needs: to perform preliminary analyses of crisis management performance subject to hypothesized variations in procedures, supporting C^3 systems, and threat; and to stimulate candidate manned simulator testbeds.

To meet the first need, a search for a viable computer model identified the Action Processing Model (APM). This batch-processed, FORTRAN model was designed and implemented in the late 1970's by IBM under guidance from DCA (reference 3). This tool models crisis management operations, detailing interactions among key command centers (e.g., NMCC) and forces. It requires the analyst to input the system architecture (e.g., resources, procedures) and a crisis scenario and it characterizes measures of performance that include workload capacity, information quality, and timeliness. Efforts are currently underway to explore the desirability and feasibility of upgrading APM to include recent advances in simulation technology (e.g., knowledge-based techniques to capture cognitive factors).

To support the need for a testbed stimulator, initial attention has focused on the Research, Evaluation, and Systems Analysis (RESA) model. This model is an outgrowth of work performed at NOSC over the last decade. It currently consists of 250K lines of FORTRAN instructions that encompass an extensive naval data base (e g., 400 ships/subs, 50 aircraft, 60 weapons) and multiple sub-models (e.g., kinematics, sensors/systems, wide-area surveillance, and weapons/damage). Programs are underway to augment the data base with information germane to Army and Air Force operations (e.g., terrain, missiles, Corps to company symbology). This stimulator could be used to provide the operational context that would underlay a simulated crisis.

b. C^3 Testbed Facilities

C^3 testbed facilities were assessed to identify environments in which the crisis management performance of human operators could be assessed when subjected to the stresses of a simulated crisis. The facility in question had to satisfy several stringent criteria:

- Flexibility. The testbed had to be flexibly reconfigurable to provide a context for assessing the performance of rapid prototypes of programmed and planned crisis management sub-systems (e.g., decision support systems).

- Realism. The testbed had to support the realistic emulation of controlled situations that could stress the crisis action system.

- Accessibility. The facility had to be available for substantial blocks of time so that statistically meaningful numbers of test replications could be run.

- Evolvability. The testbed had to be capable of evolving to a multi-site facility to deal with the distributed nature of the crisis management system.

After evaluating a number of candidate facilities against these criteria, the MITRE/DCA Command Center Engineering Laboratory (CCEL) was selected as the nucleus for the Crisis Management Testbed Facility. This facility consists of extensive ADP resources (e g., micro VAX with graphics processor and complement of intelligent terminals), workstations (e.g., Apollo, PC XT & AT workstations), a fiber optic local area network, video equipment (e.g., large screen wall projector), and extensive software resources (e.g., various compilers, data base management systems). The facility has been used in prior programs to develop and evaluate prototypes of an Executive Briefing System, an "Electronic Notebook" (based on commercial products such as advanced hypertext software and emerging digitized video), and the Automated Emergency Action Message Preparation and Dissemination System (AEPDS). Upgrades to the laboratory are programmed that should enhance its utility as a testbed for C^3 support to crisis management.

The Joint Directors of Laboratories Simulation Network (JDLNET) was also assimilated into the program in order to extend the testbed to deal with distributed C^3 issues. JDLNET is evolving to support joint activities relating to distributed C^3 systems using the Defense Integrated Secure Network (DISNET), a subnetwork of the Defense Digital Network. Early participants on the network include NOSC, CECOM, RADC, and the Naval Postgraduate School. This capability is discussed in detail in reference 5.

The Rand Strategy Assessment System (RSAS) has been developed under the sponsorship of the Director, Net Assessment, OSD (reference 8). The system has been designed to combine selective features of political-military wargaming and analytic modeling. One of its distinctive features has been the use of decision models to represent selected military command-level and political level decisionmaking processes in game play. Since many crises involve high level political, military, and diplomatic decisionmakers from several nations (see discussion in Section A.2), there is considerable interest in adapting the RSAS decision models to the needs of the DCA/C^4S testbed. It is anticipated, however, that those models will have to be tailored extensively to respond to the specific issues of interest in the DCA program.

c. Exercises

To facilitate the translation of new concepts and systems into the operational community, the testbed activity will take advantage of planned tests and exercises (T&Es). These T&Es will provide an opportunity to extend the assessment of key products to reflect many of the factors that characterize a realistic operational environment.

Three classes of exercises have been identified that are well suited to the needs of the program: 1) selected JCS Command Post Exercises (CPXs): these exercises are conducted biannually to train participants in the use of the crisis management system; 2) OSD/JCS No-notice Interoperability Exercises (NIEX): as the title suggests, these activities are conducted with limited advance warning to assess the readiness of joint forces to interoperate effectively; and 3) OSD Serial Exercises: these are a sequence of relatively limited exercises that have been conducted to validate new concepts of operation (e.g., to explore the concept of industrial readiness states that would correspond to traditional defense conditions (DEFCONs)).

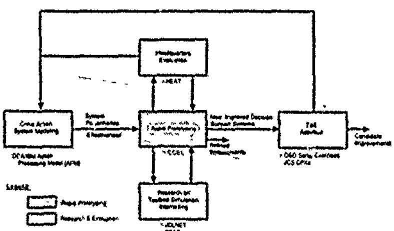
d. Assessment Tools

One of the major problems in employing the proposed testbed is the difficulty in objectively assessing and quantifying a C3 facility's performance and effectiveness. In recent years a Headquarters Effectiveness Assessment Tool (HEAT) has been developed and applied by DCA and several Services (e.g., reference 9). The tool utilizes a framework that is based on the processes that a headquarters performs (e.g., monitor, understand, plan-develop alternative actions, plan-predict consequences, direct, inform). Efforts are contemplated to adapt this framework to meet the evaluation needs of the crisis management testbed activity.

2. Testbed Activity Concept

Figure 6 illustrates the concept that has emerged to synthesize the individual evaluation resources into a consistent testbed activity.

Figure 6. Testbed Activity Concept.



A representative application of the testbed activity might begin with a preliminary analysis of a crisis management issue using an upgraded version of the APM computer model. That tool could be used to perform a preliminary assessment of the impact of a proposed system or change in operational concept on crisis management effectiveness. This assessment would include sensitivity analyses to evaluate variations in effectiveness as a function of key external variables (e.g., scenario).

If the findings of that assessment were promising, the next step would be to analyze the issue in greater depth using man-in-the-loop simulation techniques. The results of the APM analysis would be used to design the experimental test matrix for the system to be tested in the CCEL. If appropriate, a rapid prototype of the concept would be fabricated and integrated into the CCEL. A number of controlled replications would then be conducted to assess the concept to desired levels of statistical confidence. A variant of HEAT would be used to evaluate the results of the test runs. As one application, the HEAT results would be employed to upgrade the APM. If the issue in question involved distributed C^3, the activity would draw upon JDLNET and modified modules from the RSAS to emulate the other nodes in question.

There are several possible applications of the results of the man-in-the-loop simulation. First, they could serve to refine the requirements for the system under test to facilitate its acquisition. Conversely, they could provide proof-of-principle for the system, particularly if the system in question were a new, improved decision support system. For the latter case, the system could then be further assessed in the context of a T&E. That would provide the opportunity to evaluate the system in a more operationally realistic environment and to facilitate the technology transition process. If the results of the T&E were positive, the candidate improvement could then move into the appropriate acquisition stage. The results of the T&E would also be used to refine further the APM.

D. Near-Term Activities

In the near-term, two complementary testbed activities are envisioned: rapid prototyping and research and evaluation.

1. Rapid Prototyping

Several candidate rapid prototyping activities are under consideration for near-term implementation.

- **NMCC Enhancements.** There is great interest in developing and evaluating Joint Operational Tactical System (JOTS) variants to enhance the fusion of information in the NMCC. The proposed test would be undertaken to identify information requirements and to assess function enhancements.

- **CCC Applications.** The CCC currently employs paper copies of Emergency Action Packages (EAPs) for operators to refer to during crisis operations. There is interest in automating these packages and enhancing their utility by incorporating a natural language interface and expert system technology.

- **White House Upgrades.** There is interest in enhancing the interface of the WHSSS with many of the other nodes of the crisis management system. When JDLNET is fully integrated into the testbed activity, it could be used to demonstrate proof-of-concept of proposed capabilities to exchange information and finished products.

2. Research and Evaluation

Three research and evaluation activities are envisioned to satisfy the needs of the testbed activity. First, consideration is being given to updating and applying the APM computer model to support the generation and validation of crisis action system architectures and plans. Second, several activities are under consideration to support testbed stimulation and internetting. These include refining RESA to use as a testbed stimulator and developing a plan for evolving to a distributed testbed capability. Finally, there is interest in adapting the HEAT methodology to support testbed evaluations.

E. Summary

A concept for a testbed activity has been formulated that promises to provide a sound foundation to address C^3 support to crisis management from a total systems perspective. An incremental approach is being pursued to evolve the needed infrastructure while developing key near-term products. Foremost among these near-term products are rapid prototypes (e.g., JOTS variants, automated EAPs), analytical models (that can be employed to support the generation and validation of crisis management architectures and plans), and systems interfaces (e.g., interfaces between the WHSSS and the CCC). Ultimately, the pace and scope of the activity will be driven by the availability of resources.

REFERENCES

1. Joint Operation Planning System (JOPS), Volume IV (Crisis Action System), 1985.

2. Belden, T.G., "Indications, Warning, and Crisis Operations," International Studies Quarterly, March 1977.

3. Arababi, M., et al., "Design of a Simulation Model for NMCS Crisis Processes," IBM internal white paper, 1977.

4. "A Quest for Excellence," Final Report to the President by the President's Blue Ribbon Commission on Defense Management, June 1986.

5. "Command, Control, and Communications Technology Assessment: Conference Proceedings," The MITRE Corporation, MTP-87W00001, July 1987.

6. Report of the Defense Science Board Task Force on Command and Control Systems Management, July 1987.

7. Report of the Defense Science Board 1987 Summer Study on Technology Base Management, December 1987.

8. Davis, P.K., et al., "Analytic War Gaming with Ivan and Sam," SIGNAL, July 1988, pp. 59-63.

9. Feld, P., "Supporting the Evolutionary Acquisition of C^2 Systems," SIGNAL, June 1988, pp. 211-217.

USE OF SIMULATION TO IMPROVE CRISIS MANAGEMENT AT THE NATIONAL MILITARY LEVEL

B. Moore
Advanced Technology Office
Defense Communications Agency
Reston, Virginia 22090-5087

H. W. Williams
Command Center Engineering Department
The MITRE Corporation
McLean, Virginia 22102

ABSTRACT

THE CHAIRMAN OF THE JOINT CHIEFS OF STAFF (CJCS) IS THE PRINCIPAL MILITARY ADVISOR TO THE NATIONAL COMMAND AUTHORITIES (NCA). IN TIMES OF CRISIS, THE CJCS PROPOSES MILITARY COURSES OF ACTION TO THE NCA AND REPRESENTS THE MILITARY STAFF IN THE CHAIN OF OPERATIONAL COMMAND WITH RESPECT TO THE UNIFIED AND SPECIFIED COMMANDS. THE NATIONAL MILITARY COMMAND CENTER (NMCC) IS THE FOCAL POINT FOR DEFENSE RELATED INFORMATION SUPPORTING THE CJCS AND THE JOINT CHIEFS OF STAFF (JCS). WITHIN THE NMCC, THE CRISIS ACTION SYSTEM (CAS) IS ACTIVATED TO CONDUCT TIME-SENSITIVE PLANNING DURING A CRISIS SITUATION. THIS PAPER DISCUSSES THE INCREASING FREQUENCY OF CRISIS, THE CHANGING NATURE OF CRISIS, AND DESCRIBES HOW WARFARE SIMULATION WILL BE USED AT A DCA NODE ON THE JDL-SPONSORED SIMULATION NETWORK (JDLNET) TO STUDY HARDWARE, SOFTWARE, AND PROCEDURAL IMPROVEMENTS TO THE CRISIS ACTION SYSTEM (CAS). USING THE NAVAL OCEAN SYSTEMS COMMAND (NOSC) RESEARCH, EVALUATION, AND SYSTEMS ANALYSIS (RESA) SOFTWARE PACKAGE AS A CRISIS EVENT DRIVER - AND OTHER JDLNET NODES PARTICIPATING AS THE SUPPORTED AND SUPPORTING COMMANDERS -- PROTOTYPING OF CRISIS ACTIONS SYSTEM (CAS) IMPROVEMENTS WILL BE ACCOMPLISHED IN A LABORATORY SETTING.

1. THE SPECTRUM OF CRISIS

Potential crises confronting the United States range from local natural disasters to global nuclear holocaust. For natural disasters, a variety of local, State, and Federal government departments and agencies exist to confront, contain, and control them. At the international level, threats and crises are orchestrated by the National Security Council system which consists of representatives from the Departments of Defense and State, The Central Intelligence Agency, the National Security Council, and invited individuals with specialized knowledge. Some of these crises can be controlled on an economic level, some can be contained at the diplomatic level, and some require attention at the military level. Figure 1 illustrates the spectrum of crisis and relates the nature of the crisis to the consequences of the event. Increasingly, the interest, attention, and application of US military forces is directed toward crisis management of terrorism and low intensity conflict. SecDef Weinberger noted " ... these forms of ambiguous aggression have become so widespread that they have become 'the warfare of choice' over the last 40 years." [1]

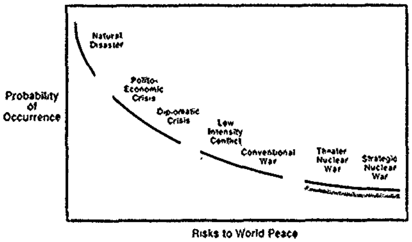


FIGURE 1. INTERNATIONAL CRISES: PROBABILITY VS RISK

As shown in Figure 2, our Government has a variety of offices and agencies that monitor and participate in the management of crisis. The White House Situation Room Support Staff and the National Security Council coordinate the flow of information to the President from the civilian and defense-related crisis management organizations. The Crisis Coordination Center supports the Office of the Secretary of Defense and, with the National Military Command Center and the World Wide Military Command and Control System (WWMCCS), provide the means for the National Command Authority to organize, mobilize, and monitor the deployment of US forces.

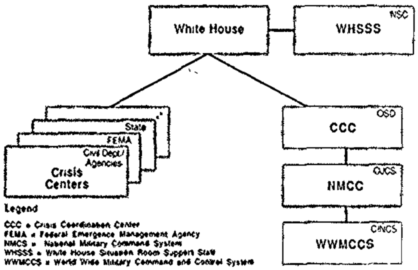


FIGURE 2. CRISIS MANAGEMENT AT THE NATIONAL LEVEL

2. CRISIS AND LOW-INTENSITY CONFLICT

From the military perspective, crisis is defined as "an incident or situation involving a threat to the United States, its territories, and possessions that rapidly develops and creates an condition of such diplomatic, economic, political, or military importance to the US Government that commitment of US Military forces and resources is contemplated to achieve US national objectives." [2] In this context, it is a precursor to Low Intensity Conflict (LIC) which has been characterized by General P.F. Gorman, former CINCUSSOUTHCOM, as "... an interdepartmental activity conducted under the lead of the Department of State and coordinated by the National Security Council." [3] and defined by the US Army as a " ... limited politico-military struggle to achieve political, military, social, economic, or psychological objectives. It is often protracted and ranges from diplomatic, economic, psychosocial pressures through terrorism and insurgency. Low Intensity Conflict is generally confined to a geographic area and is often characterized by constraints in the weaponry, tactics, and level of violence." [4]

In a review of 278 International Crises and 627 Foreign Policy Crises between 1929 and 1979, Brecher, Wilkenfeld, and Moser [5] defined a crisis as being a threat to basic values with a finite time for response and a high probability of involvement in military hostilities. As shown in Figure 3, a crisis can be characterized as an event that exceeds the ability of standard operating procedures to handle or control.

- They require adjustments to normal procedures due to
  - Magnitudes of the issues
  - Time pressures
  - Incomplete information
  - Conflicts in possible responses
  - Irreversibility of decisions

FIGURE 3. CHARACTERISTICS OF NATIONAL CRISIS

MGen J.P. Hyde, et.al. [6] described the frequency of crisis as being about one per year between mid-1950 and mid-1960. During the 1970's, the average increased to three crises per year; and through 1986, the average number of crises had increased to six per year.

Figure 4 contains a listing of 21 well-publicized military crises that the US has been involved in over the past 26 years. (This represents about 30% of the total referred to by MGen Hyde.) As shown in Figure 5, it becomes apparent that the trouble spots tend to be concentrated in three geographic areas: the Caribbean Basin, the Middle East, and Southeast Asia.

- 1961 Caribbean - Cuba (Bay of Pigs)
  Europe - W Germany (Berlin Wall)
- 1962 Caribbean - Cuba (Missile Crisis)
- 1964 Central America - Panama (Canal Zone Demonstrations)
- 1965 Caribbean - Dominican Republic (US Intervention)
- 1967 Middle East - Arab/Israeli War (USS Liberty Attacked)
- 1968 Far East - North Korea (USS Pueblo Captured)
- 1971 South Asia - Indian/Pakistani War, (US Carrier TF)
- 1973 Middle East - October War (US Alert)
- 1975 South Asia - South Vietnam (US Evacuation)
  - Cambodia (US Merchant Ship Mayaguez Captured)
- 1976 Far East - Korea (US Officers Murdered) Operation PAUL BUNYAN
- 1978 South America Guyana (Jonestown Massacre)
- 1979/1980 Middle East - Iran (Revolution US Hostages) Operation EAGLE CLAW
- 1982/1983 Middle East Lebanon. (US Intervention Bombing Incidents)
- 1983 Far East - USSR (KAL Flight 007 Destroyed)
  Caribbean - Grenada (US Intervention) Operation URGENT FURY
- 1985 Middle East - Lebanon. (TWA Flight 847 - Hostages)
- 1986 North Africa Libya (US Air Strike) Operation ELDORADO CANYON
- 1987/1988 Middle East Persian Gulf (Tanker Convoys, USS Stark Attacked)

FIGURE 4. SELECTED CRISES

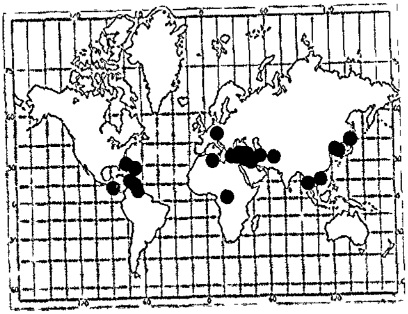


FIGURE 5. GLOBAL SITUATION

Furthermore, the locations support an analysis by Motley [7] where he defines five characteristics of Low Intensity Conflict:

- o They occur in the Third World;
- o They involve the pursuit of limited objectives;
- o They do not fit within traditional military conventional campaigns,
- o They may involve Soviet proxies;
- o They do not require mobilization of additional resources beyond those otherwise maintained for peacetime purposes.

As illustrated in Figure 5, an analysis of twenty-one crises directly involving the United States over the past twenty-six years reveals that only two (Berlin Wall and Cuban Missile Crisis) also involved the USSR.

As designers of information management systems and members of the national command, control, communications, and computer infrastructure, we are all too familiar with the impact of incomplete information, conflicts between possible interpretations of available information, and the requirement to have an answer ready "yesterday" that will successfully withstand all future 'second guessing' or 'arm chair quarterbacking'. The automated systems we have developed to support the planning, deployment, and management of forces in a conventional war may not be able to support the crises that are increasingly a part of the global picture. Today's command center staff officers and their counterparts at the White House Situation Room and the Department of State are faced with increased frequency and complexity of crisis actions, terrorist activity, and low intensity conflicts. Figure 6 illustrates the functions performed by both military and civilian analysts during a crisis situation.

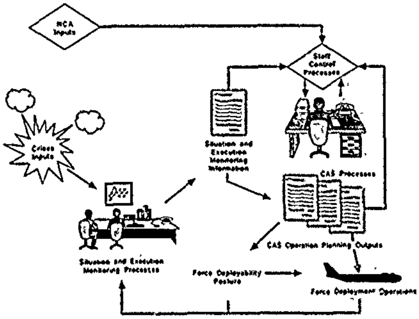


FIGURE 6. CRISIS ACTION SYSTEM

The systems they depend upon to help them cope with the growing threat have not been designed to support the current nature of crisis action processing. Current budget restriction and priorities combine to delay the implementation of integrated systems that would improve their ability to react rapidly to emerging crisis situations.

3. CRISIS ACTION SYSTEM

The President makes the decision regarding which option will be selected in response to a given crisis. The option may be economic, diplomatic, or military, or some combination of the three. One of the tasks of the Joint Chiefs of Staff is to review possible military options. The preferred option is then conveyed by the Chairman of the JCS (CJCS) to the President for his consideration.

The JCS are supported in this effort by the Organization of the Joint Chiefs of Staff (OJCS) which operates the National Military Command Center (NMCC). Via the NMCC, the JCS are kept appraised of our military status (forces, platforms, weapons, and current situation) by commanders in the field.

When a military option to resolve a crisis is selected by the President, that option, in the form of an Execute Order, is communicated from CJCS via the NMCC to the Unified and Specified commands.

The most significant element within the military planning process is time. As described in the Joint Operation Planning System (JOPS) there are two methods of military planning: deliberate planning and time-sensitive planning. Deliberate planning is the process used when time permits development and coordination of plans with numerous organizations. Time-sensitive planning is conducted when time does not permit such development and coordination; as described in Reference 2, the Crisis Action System (CAS) is the process that defines the time-sensitive planning that is performed by the JCS, the Services, the CINCs, and other defense organizations in order to develop response actions during time-constrained (i.e., crisis) situations.

The Crisis Action System (CAS) focuses on the military aspects of a crisis from its beginning, to commitment of forces, to the return to normal operations. CAS is supported in its operation by personnel, procedures, hardware, and software. For CAS to function as it was intended in the NMCC, all of these components must operate effectively.

As shown in Figure 7, CAS consists of up to six phases. Depending on the nature of the crisis, certain phases may be performed in parallel or be bypassed entirely.

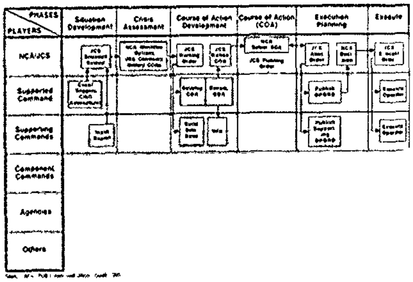


FIGURE 7. CRISIS ACTION PHASES, PLAYERS

When an event occurs that has possible national security implications, a crisis response team is formed. The team monitors inputs from various field sources and assists the JCS in considering military Courses of Action (COAs). They develop alternatives apart from those COAs being prepared by the supported commander, and review and analyze the commander's COAs. The CMCS then presents the COAs in order of priority to the NCA. The team reviews the supported commander's Operation Order (OPORD) for consistency with the JCS Alert Order (if issued) and the current situation, and prepares and disseminates the JCS Execute Order.

As the System Engineer for the NMCC, DCA has programmed significant enhancements to the NMCC in the areas of processing and display, emergency action preparation and dissemination, communications, and survivability/endurance. As described in Section 4 below, the Joint Directors of Laboratories (JDL) are sponsoring the play of distributed simulations and wargames among Service laboratories and DCA. DCA participation in JDL-sponsored warfare simulation is seen to be an excellent way to identify candidate areas for improving the CAS function in the NMCC.

The first order of business in conducting this improvement effort is to define the CAS baseline. All candidate improvements will then be evaluated in terms of the baseline performance. Since crises historically involve information which is incomplete, the CAS will be studied in terms of:

1. The impact of data input quality (obsolescent, low integrity)

2. The impact of data input quality (too much, too little, redundant, conflicting)

3. The need to reduce CAS response times

Two areas of CAS appear to be improvement candidates, namely, the use of decision aids and information management. The first area involves the application of current display technology (desktop as well as large screen), as well as the use of knowledge engineering to provide expert system support.

Information management will focus on the ways that CAS relates intelligence evaluations and operations data, and the design of the CAS data-base to improve crisis information storage and retrieval.

4. JDL WARFARE SIMULATION EFFORTS

In March 1982, the Assistant Secretary of Defense for Research and Development (R&D) recommended establishment of a Joint Directorship of DOD Laboratories and R&D Centers. As part of this initiative, the JDL C3 Research and Technology Program was established in 1983. Planned and guided by the JDL Technical Panel for C3 (TPC3), it consists of representatives from DCA, the National Defense University (NDU), the Naval Postgraduate School (NPS), the Air Force's Rome Air Development Center (RADC), the Army Communications and Electronics Command (CECOM), and the Naval Ocean Systems Center (NOSC). With NOSC as Program Manager, the JDL-TPC3 has sponsored the development and acquisition of a common architecture to support C3 system studies at the Service laboratories and research facilities. Commonality is achieved through use of Digital Equipment Corporation (DEC)-based hardware, a NOSC-developed warfare simulation software package (Research, Evaluation, and Systems Analysis (RESA)), and communications connectivity using the Defense Integrated Security Network (DISNET) component of the Defense Data Network (DDN).

The nodes that are currently planning to participate in distributed warfare simulation using RESA are shown in Figure 8.

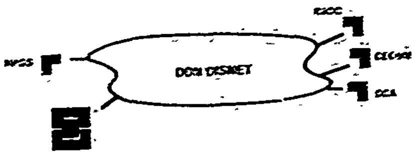


FIGURE 8. THE JDL SIMULATION NETWORK (JDLNET)

The number of nodes may increase in the future. Connectivity will be provided by DDN/DISNET with a data rate of 19.2 kilobits per second. The differences in the representations of the nodes on Figure 8 indicates that one node (here NOSC) must control the central simulation process.

RESA is an outgrowth of the Interim Battle Group Tactical Trainer (IBGTT) effort which was developed by NOSC as a simulation-based training aid for Navy command personnel. RESA provides a two-sided interactive simulation of a large-scale wargame. It includes models of sensors, communications paths, forces, weapons, navigation systems, and weather conditions. Presently incorporating Navy force characteristics and display symbology, the simulation is to be revised in FY89 to include those for the Army and Air Force. Figure 9 provides a summary of present and programmed capabilities of RESA.

The RESA software has been developed so that it can be operated locally as well as distributed. For the local application, the system software is executed on a stand-alone DEC computer system such as a MicroVax-II and all RESA warfare simulation functions would be performed on that system.

- Current capability
  - Naval data base (e.g., 400 ships/subs, 50 A/C, 140 radars, 60 weapons)
  - Timeframe: Emphasis - 1985; Partial 1995, 2010
  - Multiple sub-models (e.g., Kinematics, sensors/systems, wide-area surveillance, weapons/damage).
  - Implemented for local use (e.g., Dahlgren, White Oak, NPS, CECOM, NPC (FRG), CFC (ROK), CENTCOM, NOSC, EADC)
- Programmed capability
  - Army, Air Force data base (e.g., terrain, Corps to Company symbology, missiles)
  - Distributed wargaming in CY 89 (e.g., NOSC, DCA/MITRE, EADC, CECOM, NPS)

FIGURE 9. RESA: JDLNET CRISIS ACTION GENERATORS

For distributed operations, the computer system at one node will actually execute the simulation. Because of the computing power required to handle a scenario involving many force elements and work stations, the DCA node will not be able to act as the central processor for distributed operations. It is, however, able to support three workstations in local mode operations.

Regardless of where this processor is located, user commands from all nodes participating in the simulation will be transmitted over DDN/DISNET to the central processor. At that location (shown in Figure 8 to be at NOSC) the simulation database would be maintained and simulation status updated. Appropriate changes to the database resulting from user commands would be processed and transmitted back to the participating nodes for display processing and local database updating. As a result, although the operation appears to each participant as a distributed processing system, it is actually a centralized processing system with remote display terminals.

Since it is intended that the warfare simulation will use a classified database of force elements, sensors, and weapons, each node requires a communications security unit (TSEC/KG-84A) for encoding/decoding transmissions over DDN/DISNET.

5. IMPLEMENTATION OF THE DCA NODE

The DCA node on DISNET has been established in MITRE's Command Center Engineering Laboratory (CCEL) in McLean, Va. The hardware and software configuration being implemented for the DCA DISNET node is shown in Table 1. This configuration will provide the capability of the node to use RESA to simulate a warfare environment in the same way that it can be done at NOSC, EADC, CECOM, and NPS.

HARDWARE	DESCRIPTION
DEC MicroVAX II	32-bit Computer
DEC LN03 Plus	Laser Color Printer
DEC VT220 (6)	Monochrome Data Terminal
DEC VT220 (3)	Terminal Keyboard
DEC GPX II 230 Monitor	Graphic Display System
DEC ACP 5250	X.25 Controller Board
Tektronix 4208 (3)	Color Graphics Terminal
Tektronix 4956	Ink Jet Plotter
Tektronix 4957 (3)	Tablet
[illegible]	Fixed Plant Adapter, Ref. One Channel
TSEC/KG-84A	Communications Security Unit
Modem	Communications Modulator/Demodulator Unit

SOFTWARE	VERSION
MicroVMS	4.6
[illegible]	4.6
DEC Windowing Software (VWS)	3.2
DEC FORTRAN	4.7
Network Solutions Open Link (TCP/IP)	1.4
DEC Driver for ACP 5250	

TABLE 1. HARDWARE AND SOFTWARE CONFIGURATION

The DCA node currently consists of three work stations, one serving as controller and the other two representing National Military Command System (NMCS) personnel. Each workstation includes a color graphics terminal, three monochrome terminals and a user interface terminal. In the DCA applications, RESA will be used as an event driver, generating a crisis environment within which NMCS personnel would function. The controller workstation would be used to control Blue and Red forces as well as to inject other events (i.e., intelligence inputs, decisions by the National Command Authorities (NCA), and actions taken by other nations); the controller would thus orchestrate the events which stimulate the other workstation operators (the crisis response team)to take action. Figure 10 is a representation of the DCA node, indicating the capabilities that are currently available in the CCEL to support the CAS enhancement investigative process.

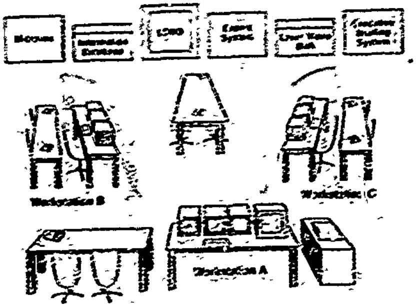


FIGURE 10. DCA NODE TEST REPRESENTATION

Figure 11 shows the layout of the CCEL and the use of the 2500 square-foot facility. When operators and monitors are being trained and scenario events are being developed, the work stations will be located in the area labeled Command Center Testbed Development.

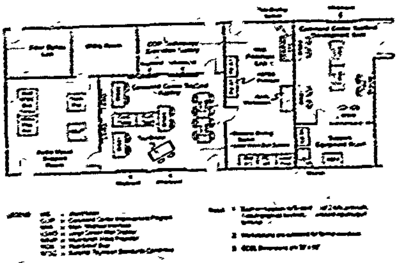


FIGURE 11. COMMAND CENTER ENGINEERING LABORATORY (CCEL)

When exercises are being conducted, the three work stations will be relocated into the Command Center Testbed Facility area. As indicated, the CCEL has in-place decision aids which will be used during DCA simulations activities to prototype candidate improvements to CAS. These decision aids are described below in terms of their proposed use in CAS.

- Use of the Large Screen Wall Display (LSWD) and associated camera/projection systems (video copy camera, flatbed scanner, etc.) to support teleconferencing and briefing. This would facilitate distribution of graphical and textual material existing in hardcopy form or displayed on a terminal to a much wider audience for discussion/decision-making purposes during conferences and briefings.

- Use of executive-type briefing systems to support the crisis team in briefing the DDO as well as briefings to the NCA and JCS. This would expedite the preparation of time- sensitive briefings in appropriate formats and use of color. Results could use split screens, storyboard-type layouts, mixing of text and graphics, etc. Changes recommended during the briefing could be made on-the-fly pursuant to carrying the briefing forward to the next level of review.

- Use of message text formatting to reduce composition time as well as transmission time. (For example, during crisis situations, only selected data are of significance; even though are text formatted, there is a need to impose stricter formatting to improve the storage and retrieval process).

- Use of table-top computer systems to perform back-side processing of input data in a variety of ways to support the needs of different levels of decision-makers. At one end of the spectrum, staff personnel might only be interested in tracking war-fighting capabilities of selected forces; at the other end, high-level planners need a natural language interface to access databases in everyday English.

- Use of laser video disk systems to provide high quality maps and other graphics (personalities, force elements) for display purposes to CAS action officers or to planners. This information could then be directly input into executive briefing systems to assist in the decision-making process.

- Use of methods to fuse the data carried in WIS (WWMCCS Information System) and other crisis management systems (e.g., White House, Crisis Coordination Center, and the intelligence community). WIS is the modernization program for WWMCCS ADP that includes updates to the CAS.

6. CAS GAMING AT THE DCA NODE

Like the other JDLNET nodes, the DCA node will operate in both a local and a distributed or "netted" basis. During both operations, the DCA node will function as echelon-above-CINC, namely, as the NCA, JCS, and NMCC staff.

During distributed operations in FY89, when JDLNET is expected to be operational, present plans are to exercise a series of European, Central American, and Korean area scenarios. Although the scenarios will focus on echelons-below-CINC, that is, conventional warfare, the scenarios will allow DCA, using military personnel with appropriate NMCC operational skills, to execute the NMCC role. The scenarios will be modified to emphasize CAS participation when the COA is being determined and before actions are taken by the supported and supporting commanders (represented by other nodes) to execute the OPORD. In addition, the scenario will allow the NMCC to issue additional alerts and plans as a result of changes to the situation.

The basic scenarios will be defined by NOSC, RADC, and/or CECOM. These scenarios will provide general information (i.e., the crisis situations, the tactical situations, force lists, workstation assignments at each node, control procedures, and analysis objectives). DCA will specify those changes to the basic scenarios that will allow the DCA node to represent the crisis response team, JCS, and NCA. Participation will include requesting information from, and transmitting orders to, the supported and supporting commands specified in the basic scenarios as well as conducting investigations into the use of CAS decision aids and information management techniques that are of special interest to DCA.

Injection of DCA node messages for transmission to other nodes will be pre-coordinated in the warfare simulation plans as will the responses that may be required to be transmitted to the DCA node by the other nodes. When messages are required to be transmitted according to the scenario script, the planning portions of the messages will be provided; the personnel involved will then input the date/time groups, perform minor editing, and add a section of the message like the mission statement or decision. At the DCA node, the emphasis will be on the procedures crisis response team personnel invoke during scenario execution and the ways to enhance the procedures. The function of the controller position will be to ensure that no actions are taken that will obviate continued play of the scenario as planned.

Typical messages to be sent/received by the DCA node during distributed warfare simulation exercises are as follows:

a. Send "Requests for CINC Assessments" to supported command

b. Send "Requests for Input Reports" to supporting commands

c. Receive "CINC Assessments" and "Input Reports"

d. Send "Warning Order" to supported and supporting commands

e. Receive "Recommended COA" from supported command in the form of "Commander's Estimate"

f. Send "Alert Order"/receive "OPORD" from supported command

g. Send "Execute Order" to supported and supporting commands

7. TIME SEQUENCING OF CRISIS EVENTS

Planned JDLNET warfare scenarios will be tactical in nature in order to satisfy the needs of the Service laboratories. Initial scenario events will cause tensions between the US and the other countries involved, to escalate. That is, they will define the nature of the crisis and are thus the natural objects of the Crisis Action System (CAS). The DCA node will be the primary player at this time. The DCA node will play a back-up role to the other nodes after the Execution Order has been issued and the actual warfare simulation is in progress.

Typical events that would occur during two days of scenario time are presented in Figure 12. During Day 1, the DCA node would be fully manned and the other nodes would monitor scenario play on a watch-team basis. During Day 2, the roles would reverse; the other nodes would be fully manned and the DCA node would employ a watch team.

DAY 1		HOUR ONE	HOUR TWO	HOUR THREE	HOUR FOUR
DCA NODE PRIME	CHECKOUT	• CINC ASSESS. (P) RECEIVED • INTEL REPORTS (P) RECEIVED • NCA CONSIDERS OPTIONS, JCS CONSIDERS COA	• JCS ISSUES WARNING ORDER • SUPPORTED (P) COMMANDER FORWARDS ESTIMATE • JCS BRIEFS NCA ON COA & NCA DECIDES	• JCS ISSUES ALERT ORDER • SUPPORTED COMMANDER PUBLISHES OPORD • SECDEF (P) DIRECTS JCS TO ISSUE EXECUTE ORDER	TERMINATE
DAY 2		**HOUR ONE**	**HOUR TWO**	**HOUR THREE**	**HOUR FOUR**
SERVICE NODES PRIME	CHECKOUT	• SUPPORTED COMMANDER ISSUES EXECUTE ORDER • INTEL UPDATES (P) ISSUED BY JCS • FRIENDLY/ENEMY ELEMENTS CONTROLLED BY BLUE/ORANGE COMMANDERS	• JCS REQUESTS UPDATED COMMANDERS ESTIMATE • UPDATED ESTIMATE PREPARED/TRANSMITTED →	• JCS DIRECTS OPORD CHANGES • SUPPORTED COMMANDER ISSUES REVISION →	TERMINATE

(P) - PRESCRIPTED

FIGURE 12. DCA NODE ACTIVITIES FOR A DISTRIBUTED SCENARIO

8. SUMMARY

To recap the way in which DCA will participate in distributed warfare simulation over JDLNET, the following actions will be accomplished:

a. RESA will be used to set the crisis event stage by describing the distribution and capabilities of platforms, sensors, and weapons

b. Time-sensitive and/or lengthy message traffic will be prescripted, and injected manually be control personnel into the simulation at the appropriate times

c. Measures of Effectiveness (MOEs) that are unique to DCA interests will be developed for the CAS process. It is not expected that RESA will be modified for this purpose

d. Decision aids and information management aids will be implemented at the DCA node so that their use is transparent to the Service laboratory nodes.

Table 2 presents a milestone schedule for DCA activities over the next two fiscal years.

RESA augmentation refers to the adapting of scenario play to handle time- sensitive and/or lengthy messages. RESA's present limit of eight lines of narrative per message will be modified by a manual work-around technique. It is anticipated that significant efforts will be required initially to train observers, participants, and test direction personnel as well as to prepare test plans and develop scripts and supporting scenario applications software relevant to DCA's needs.

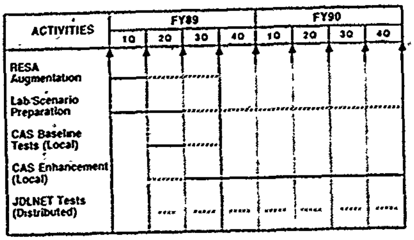


TABLE 2. MILESTONE SCHEDULE

As shown, DCA tests on a local basis will be conducted monthly. Because of the amount of pre-and post-test coordination involved with distributed tests, it is expected that DCA node participation would be limited to once per quarter.

Assuming that JDLNET connectivity, funding, and resources are provided as planned, we look forward to presenting an update on our activities at next year's JDL Research Symposium.

References

1. Annual Report To The Congress, FY88

2. Joint Operation Planning System (JOPS), Vol. IV (Crisis Action System), 1985

3. Testimony before Senate Armed Service Committee, January 1987.

4. Joint Low-Intensity Conflict Project Final Report, Executive Summary, USA Training and Doctrine Command, 1 August 1986.

5. Michael Brecher, Jonathan Wilkenfeld, Sheila Moser, "Crises in the Twentieth Century," Pergamon Press, 1987.

6. MGen J. P. Hyde, Col. J. B. Warren Jr., Cmdr C. E. Kesson II, "C3 Planning in Crisis Response", Signal, November 1986

7. J. B. Motley, "Low Intensity Conflict: Global Challenges", Presentation to The MITRE Corporation, 21 June 1988

The Knowledge-Based Battle Management Testbed Project

Captain Jeff Grimshaw

Air Force Systems Command
Rome Air Development Center
Griffiss AFB, NY

1.0 Abstract.

The development of expert systems designed to help plan and execute tactical strike missions has led to the need for some type of testing environment. To this end, a testbed is being developed to exercise battle management expert systems individually and in concert with other systems. In addition to providing a framework for evaluating these expert systems, the testbed will establish database and message passing protocols so that future expert systems can concentrate on reasoning over the data and not on the input/output issues.

2.0 Introduction.

Expert system technology has matured to the point that systems can be built to assist the battlefield commander. These systems use geographic data, offensive and defensive force strengths, and operational policy to assist the user in making plans and decisions. Most battlefield commander decision aids have been designed for independent use. No mechanisms are provided for sharing or requesting information between multiple systems. Furthermore, the development of each new system has required the building (or rebuilding) of input/output facilities, both between the system and user and between the system and any required databases.

Another limitation has been the static method of testing new systems. Typically, an expert system is presented a well planned, stationary problem. A more realistic evaluation would be performed by presenting the system with a dynamic problem, requiring decision processes while controllable (e.g., human input, alternate expert systems, etc.) and non-controllable (e.g., equipment malfunction, enemy operations, etc.) actions are continually modifying the problem domain.

To address these issues, the Knowledge-Based Battle Management (KB-BATMAN) Testbed Project was initiated [7]. As its name implies, the purpose of this project was to develop a testbed for evaluating various battle management technologies. The testbed was partitioned into four major components:

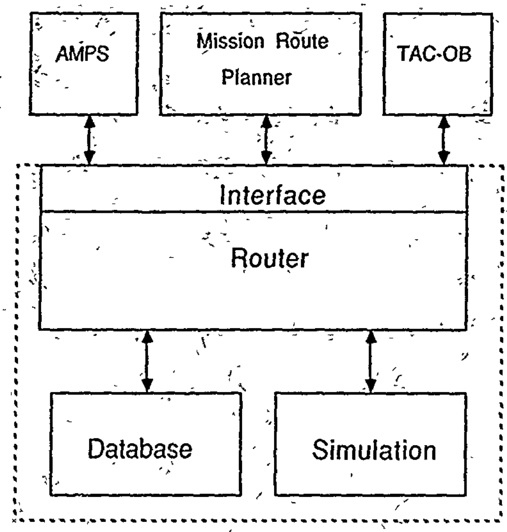


Figure 1: KB-BATMAN Testbed with Battle Management Decision Aids

1. a database containing problem domain data,

2. an interface between expert systems and the testbed,

3. a router for alerting systems to changes in the database, and

4. an object-oriented simulation to provide a dynamic environment.

These four components will be addressed in subsequent sections. An overview of the testbed is shown in Figure 1. The Database, Interface, and Router components make up the Framework. The Framework suffices in providing battle management expert systems with a standardized access to Database information as well as a mechanism for communicating between expert systems. During an operational exercise, changes to the Database would be made based upon post-mission debriefings, intelligence reports, forward air controller reports, etc. The Simulation was developed for non-exercise testing. This component was included to provide a realistically changing environment, extending the basic Framework into a Testbed.

The Framework provides for asynchronous operation of expert systems which may be distributed over heterogeneous hardware and software. This architecture is similar to the Blackboard artificial intelligence approach [5]. The system operates in an opportunistic, dataflow manner whereby problems or partial solutions are posted to the database for further refinement. The problem handling capabilities of the various systems are maintained, implicitly, by the router. As information useful to the systems becomes available, the router makes appropriate notifications. The intent is to have very low coupling between the expert systems. The resulting architecture is modular, allowing for the easy addition or removal of decision aids.

The initial thrust of this research has been to develop a testbed oriented towards tactical air operations. However, the overall system has been developed generically so that in the future, ground force operations may be included. This will be important since air and land missions will need to be planned and executed in concert.

3.0 Battle Management Decision Aids.

Battle management decision aids are being developed to assist commanders in various aspects of an air-land war in Central Europe. These expert systems include A Multi-Level Planning System (AMPS) (developed by MITRE-Bedford), TAC-OB (developed by MITRE-Washington), and aircraft route planners.

The AMPS project is a follow-on from the KNOBS Replanning System (KRS) developed at MITRE [2]. These systems were designed to plan tactical air strikes based on friendly resources and prioritized enemy targets.

The purpose of TAC-OB is to provide the mission planning expert systems with a prioritized target list. TAC-OB accomplishes this in a three step process. Initially, the sensor reports are retrieved from the Database using the Router and sent to TAC-OB. Secondly, these reports are reformatted and compared to current enemy order of battle and targets. This results in an overall assessment of threat status. Lastly, this threat status is prioritized based upon capability and vulnerability, and then sent to the Database.

Mission route planners are being developed for use at the Squadron level for selecting ingress to and egress from the target. Due to the complexity of the task, route planners are usually semi-automatic, working with the pilot [1]. Depending upon the time and resources available, future route planning systems may need to be more automated, tailoring their results to the preferences of the individual pilots.

4.0 Database.

The Database system was originally developed by MITRE (Bedford) to support AMPS. The purpose of AMPS is to demonstrate the use of knowledge-based techniques in the planning of tactical aircraft strikes in Central Europe. Information needed by this system, and retained in the Database, includes:

A. US Tactical Air Command base locations.

B. Locations of specific fighter squadrons.

C. Aircraft type and number at each squadron.

D. Aircraft and weapon capabilities.

E. Enemy airbases.

F. Enemy aircraft deployments.

G. Enemy surveillance radar locations.

H. Enemy surface-to-air missile (SAM) site locations.

In addition to resource and threat data, the Database provides a "scratch-pad" for expert systems and a repository for their results. For instance, the "scratch-pad" would provide a place for the planner to post tentative plans while it checks for constraint violations. The Database is also a repository for expert system results such as planned missions, flight routes, and prioritized target lists.

While MITRE-Bedford continues the development of AMPS, MITRE-Washington has proceeded on the development of the Framework. Part of the Framework development involves the evolution of the AMPS database into the independent Database. This is necessary since the Database will need to support diverse types of expert systems, not just AMPS.

The information in the Database, although unclassified, is of sufficient resolution to exercise AMPS. The information needed by the Simulation is read in from the Database and assumed to be correct. This provides the model for "ground truth," the "real" state of the environment. Changes to the Database (e.g., aircraft availability, target status, etc.) will be accomplished by the Simulation. While the Simulation maintains ground truth, uncertainty may be introduced to the data posted to the Database. For instance, target status based upon post-mission debriefings may not be as accurate as that provided by reconnaissance missions.

The Database system consists of ASCII datafiles that may be accessed relationally. The specific model used was implemented as a custom built, single-access, relational database management system. This system supports the standard functions including addition, deletion, and projection of data.

5.0 Interface.

The purpose of the Interface is to translate between individual knowledge-based system protocols and the common message protocol used by the Framework. The Interface provides two way communication between expert systems and the Framework, as well as a method for communicating requests/information between inter-dependent knowledge-based systems. Development of the Interface is being conducted by MITRE-Washington and is currently in an early stage.

6.0 Router.

The purpose of the Router is to handle requests between each expert system and the Database or other expert systems. In addition, the Simulation will access the Database through the Router, both for receiving mission Air Tasking Orders (ATOs) and for sending post-mission results (e.g., Bomb Damage Assessments (BDAs), aircraft losses, etc.).

The Router "orchestrates" the operations of the Framework, providing an intelligent method of controlling system operation.

7.0 Simulation.

The purpose of the Simulation is to simulate the execution of planned tactical missions against defended targets and return the appropriate results. Tactical missions include air strikes, refueling missions, and reconnaissance missions. Future extensions of the Simulation will support ground operations such as tank, artillery, and troop deployments.

7.1 Simulation design.

An object-oriented approach was selected for the Simulation. This approach has several advantages over a conventional, procedural approach. First and foremost, simulation naturally lends itself to the notion of objects and operations. Each object, whether it be an aircraft, base, or town, has certain attributes which could be inherited from superclass objects. For instance, a specific F-15 has the same class attributes as all F-15s, as well as those of all aircraft, and all moving objects, etc. Defining these attributes once at the appropriate level makes system development much clearer and easier. The second reason for using an object-oriented approach is that messages may be used to communicate between objects. Since objects in a simulation are continually "talking" between themselves, it is important that the technique used to simulate the objects provides an explicit message passing mechanism. Thirdly, since objects change (e.g., move, die, etc.) based upon message passing, simulations take on a dataflow characteristic similar to that of the real world. Aircrews receive orders, plan routes, fly missions, occasionally get shot down, effect targets, and regroup at the home base. As this script evolves, planned and unplanned changes in the simulation, (e.g., saturation of air-defense systems or mechanical problems), are handled appropriately by behaviors without explicit control by a "main program." (A behavior is procedural code tied to an object or object class.)

7.2 Simulation Implementation

The Simulation was implemented in the Enhanced ROSS in Common Lisp (ERIC) object-oriented programming language developed by 1Lt Mike Hilton at the Rome Air Development Center (RADC) [4]. ERIC has two major improvements over ROSS: an enhanced clock facility and a Common Lisp implementation. The clock facility provides a mechanism for controlling objects temporally. The clock itself is an ERIC object and views time as discrete points. Thus, activities that are to be performed in the future get posted "on" the clock based upon a real numbered time. These activities may be set to occur at a fixed time or as an offset to the current time. The time of the clock may be set and then incremented using the "tick" command. Tick advances the clock a user-defined amount of time. After time has advanced, all messages posted on the clock queue for execution before the current time are removed from the queue and broadcast to the objects. Objects with behaviors corresponding to these messages then perform the required operations. Since managing an event queue can require a lot of CPU time, mechanisms for expediting this function are important.

Managing the event queue consists primarily of inserting new events at the proper location on the queue and retrieving the next event for execution. While retrieving the next event is trivial, searching the queue for insertion may be much more time consuming. In addition, it can be expensive to remove events that should not occur. To keep the event queue short, and thereby reduce queue processing, ERIC uses multiple queues. Thus, each object contains its own event queue while the ERIC clock maintains the next event for each object. As an event occurs, ERIC takes the next scheduled event on the object's event queue and posts it, if such an event exists. This provides two benefits. First, events can be inserted and/or deleted from short object event queues speeding up the process. Second, when an object "dies," the system need only delete the object's first reference on the system event queue and then delete that particular object's event queue. This also speeds up the simulation.

As its name implies, ERIC was written in Common Lisp. Specifically, Symbolics Common Lisp was used. The selection of Common Lisp was to make the simulation language more portable between various platforms. Competing with this desire to standardize was the concern over execution speed. Simulations are by nature computationally intensive programs. A simulation of flights of aircraft flying against air defenses requires continual analysis of who can "see" whom and what actions or responses are required. As the number of objects in the simulation increases, the process can become bogged down. To help alleviate this problem, ERIC was modified to take advantage of the Flavors capabilities of the Symbolics Lisp environment. While this approach has speeded up the simulation process, portability to non-Symbolics Lisp machines has been affected. One solution that would provide the best of both worlds would be to re-implement ERIC using the Common LISP Object System (CLOS), once the standard for this object-oriented system has been agreed upon. This should provide the speed of Flavors with the portability of Common LISP.

7.3 Map Display System

Early in this project it became clear that an object-oriented cartographic system would be useful, allowing weapon systems to query geographic entities for information. For instance, Army convoys need road and city data when traversing cross-country. If a road is impassable, some mechanism must be provided so that real or simulated commanders can develop alternate plans. Such a system has been developed [3]. The resulting cartographic database is read in during system initialization as multiple ERIC objects. Thus, commanders can check road status as easily as checking the status of a mechanized battalion.

The loaded cartographic ERIC objects also provide ground-truth for the KB-BATMAN Testbed. Certain types of data within the Database will always be accurate, (for example locations of bridges), however variable data may be inaccurate (such as the status of bridges). For example, consider the airstrike that destroyed its target but failed to return to base. Assuming that the aircrews had not been debriefed in-flight, only the simulation would know the target had been destroyed. The Database would reflect inaccurate data. In order to update the Database, the real or simulated commander might have to order a reconnaissance mission. Operating the Database with incomplete and/or incorrect data will further test the robustness of the battle management expert systems.

The resulting system generates an interactive, variable resolution map that can be incorporated into applications programs requiring cartographic support. Highlights of the system include: feature abstraction (ability to turn features on and off), unconstrained pan and zoom, and mouse sensitive cartographic features. The system runs on a Symbolics Lisp Machine equipped with a color graphics display [3].

The cartographic features are actually ERIC classes and objects. The classes include: airstrips, country borders, bridges, dams, heliports, lakes, obstructions, powerlines, railroads, roads, road intersections, towns, and waterways. Object locations may be retrieved in units applicable to latitude/longitude, Universal Transverse Mercator (UTM), or Military Grid Reference System (MGRS). The data for the cartographic features were originally taken from Joint Operations Graphics (JOG) charts. These charts provided unclassified data of sufficient precision (approximately one kilometer) for the Simulation.

8.0 Status

Portions of the Testbed have begun to take shape. With the maturation of AMPS has come the specification and development of the Database. These contractual efforts have progressed concurrently with in-house RADC work on the Simulation and Map Display System.

The "first-cut" of the Simulation was completed earlier this year. This version can send airstrike missions as ordered in ATOs and coursely simulate enemy air defenses including fixed and mobile SAM sites. Upon completion of the missions, "aircrews" provide PDA reports as well as aircraft attrition numbers.

Currently, the Router and Interface components are not implemented. Now that the Simulation is "flying" and other decision aids are being developed, these components will need to be created.

9.0 Future Work.

While MITRE-Washington continues to work on the Router and Interface development, in-house RADC work will center on extending the capabilities of the Simulation. This will include the ability to simulate reconnaissance and refueling missions, as well as further refinement to the air-defense network. An additional area of research will concern the encapsulation of offensive and defensive tactics doctrine into the Simulation. While recent work has centered on getting the system up and running, future efforts will need to concentrate on the Simulation's fidelity, including the area of doctrine.

Additional in-house RADC work will address user interface issues. Specifically, a requirements analysis will be performed next year to identify the best ways of communicating with the user. Applicable technologies include various natural language, speech generation and recognition, and graphics techniques.

Future contractual efforts will also investigate land operations, with emphasis on how these impact air operations. As an example, if enemy ground attacks have overrun friendly airbases or friendly army forces have destroyed high valued enemy targets, what effect will this have on the mission planning process of AMPS or the target prioritization of TAC-OB? The ground war would also be of interest to the mission route planning system since flying over enemy held territory is not advisable. This expansion to the Simulation should provide a more realistic environment for testing the decision aids.

There will also be follow-on work to develop a more robust database. The new system will be a relational database using ORACLE. The purpose of this effort is provide the Testbed with a larger, more realistic database which will provide an environment for addressing large scale and real-time database issues.

10.0 Conclusions.

Development of the KB-BATMAN Testbed will provide a foundation for testing battle management aids. This testing will help improve the accuracy and robustness of decision aids designed for either operational or research uses. In addition to exercising individual aids, the Testbed will provide a mechanism for investigating synergistic effects of multiple expert systems reasoning over the data in the Database.

This project has also provided an opportunity to use knowledge engineering techniques to the simulation problem domain. Simulations readily lend themselves to an object-oriented implementation. The eventual use of tactical domain knowledge in the simulation will help provide a realistic testbed for stressing decision aids.

BIBLIOGRAPHY

1. Bradshaw, Jeffrey S., 2Lt, "A Pilot's Planning Aid for Route Selection and Threat Analysis in a Tactical Environment," Air Force Institute of Technology Thesis, AD-A178871, December 1986.
2. Dawson, Bruce C., et al, "Knowledge-Based replanning System," RADC Final Technical Report, RADC-TR-87-60, May 1987.
3. Hilton, Michael L., 1Lt, "Design and Implementation of the Map Display System," Unpublished RADC report, May 1988.
4. Hilton, Michael L., 1Lt, "ERIC: An Object-Oriented Simulation Language," Rome Air Development Center In-House Report Number RADC-TR-87-103, July 1987.
5. Nii, H. Penny, "Blackboard Systems: The Blackboard Model of Problem Solving and the Evolution of Blackboard Architectures," The AI Magazine, Summer 1986.
6. Nugent, Richard O., and Richard W. Tucker, "TAC-1: A Knowledge-Based Air Force Tactical Battle Management Testbed," RADC Interim Report, RADC-TR-88-10, January 1988.

MISSION ORIENTED COMMAND AND CONTROL PLANNING BY THE MARITIME MAJOR NATO COMMANDERS

D.B. GRISWOLD
MITRE C3I DIVISION
McLEAN, VIRGINIA 22102

ABSTRACT

There is a direct relationship between military operations and Command and Control (C2). In recognition of that fact, NATO has adopted a mission-oriented concept for C2 system planning. The Maritime Major NATO Commanders are using a top-down mission-oriented concept which compares C2 system capability objectives with C2 system capabilities to determine if the C2 system is supporting operations to the degree required. C2 system capability objectives are derived from analysis of the operational mission. C2 system capabilities are assessed from analysis of the C2 system elements. Packages of C2 system improvements are developed as needed to resolve identified problems. An overall improvement plan is developed consistent with a NATO C2 system goal architecture. This paper addresses the C2 planning concept as used by the maritime commands.

INTRODUCTION

Within the North Atlantic Treaty Organization (NATO) there exists a command and control (C2) infrastructure which is owned, operated, and maintained by NATO. This infrastructure encompasses what might be called the theater level of command. The theater level of command is comprised of those commands which are internationally staffed. Typically, these commands range from the Major NATO Commanders (MNCs) down to, but not including, the level of command staffed entirely by national personnel (e.g. Corps, Wings, and Task Groups).

Requirements for improving the military C2 infrastructure are specified bi-annually by the three Major NATO Commanders: Supreme Allied Commander Europe (SACEUR), Supreme Allied Commander Atlantic (SACLANT), and Allied Commander in Chief Channel (CINCHAN) in the Tri-MNC Command and Control Plan (Tri-MNC C2 Plan). In 1986, NATO's Defence Planning Committee decided that proposed improvements to NATO's C2 systems should be justified in terms of the operational missions which those systems support. In response to NATO direction, the MNCs have adopted the concept of mission-oriented C2 planning for preparation of the next edition of the Tri-MNC C2 Plan.

This paper presents the major aspects of the mission-oriented approach being used by the maritime MNCs, SACLANT and CINCHAN. Emphasis is on process and framework rather than results.

THE MISSION ORIENTED PLANNING PROCESS

A six step planning process for plan development was agreed to by the MNCs in late 1987. The steps in the process are:

- -Operational Framework Development
- -C2 Requirements Definition
- -Current C2 System Assessment
- -Solution Options Definition
- -Improvement Plan Development
- -C2 Plan Coordination/Impact Assessment

Each of these steps is discussed below.

OPERATIONAL FRAMEWORK DEVELOPMENT

The basic framework for addressing C2 improvements is derived from an analysis of the NATO mission. The NATO Conceptual Military Frameworks (CMFs) define mission components and functions and establish their relationships. The CMFs are a long range forecast of requirements for military systems and their supporting elements; some statements of maritime C2 needs are found in the Maritime Conceptual Military Framework (MCMF). An early decision was made to constrain the framework elements of the mission-oriented approach used in the Tri-MNC C2 plan so that the mission, key mission components, and military functions would be those defined by the CMFs.

The operational framework consists of strategic, key mission component, and military function elements. The operational framework addresses these operational elements within the context of four levels, or types, of conflict: tension and crisis, conventional conflict, limited nuclear conflict, and general nuclear war. Although many of the significant events are actually threshhold events between levels of conflict, each level of conflict is defined uniquely in such a way that significant events are associated with a given level of conflict so that the unique conflict stresses placed on any system may be highlighted and all the stresses of conflict are addressed. The operational framework can be illustrated as a series of matrices. The operational framework reflects the mission requirements for all the MNCs. The foundation of the framework is the strategic objectives matrix.

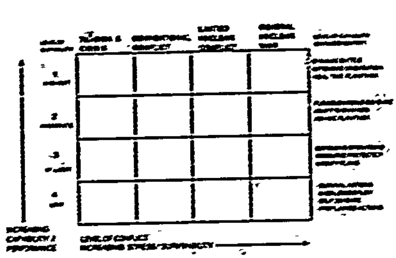


Figure 1 Strategic Matrix

The strategic objectives matrix is essentially a four by four relationship that portrays the basic command objectives in reference to levels of conflict and levels of operational performance as illustrated in Figure 1. The two dimensions are a reflection of the capability to invest in increased performance of forces and supporting systems in the same conflict environment or to invest in survivability of forces and support as the stress (conflict environment) changes or to invest in some combination of performance and survivability that best suits the command. It is desirable for C2 system analysis that the different operational blocks reflect significant breakpoints in C2 system investment. The highest level of operational performance or capability objective represents the desired military objectives as stated in various operational plans. However, desired capabilities may not always be achieved for force structure, financial, or technological reasons. Therefore, capability objectives which are subsets of the desired are derived to provide a yardstick by which achievement can be gaged. It is desirable that the lower order capability objectives be alternative concepts of operation. The current philosophy portrays the lowest level as one of preparation and self-defense; the next highest level as one of predominately defensive operations from relatively fixed positions; the next highest level as one of highly mobile, flexible defensive operations; and the highest level of capability (desired goal) as one of flexible, dynamic operations characterized by offensive actions. Any set of capability objectives in one level is incorporated automatically in the next higher level of capability objectives. The horizontal dimensions of the strategic matrix represents the change in conflict stress defined by level of conflict parameters. The strategic objectives matrix portrays a possible range of independent yet related actions and is designed to illustrate the impact of those actions in terms of force and system investments.

Operational actions executed to achieve strategic objectives are defined by Key Mission Components (KMCs). The conceptual military frameworks define nine key mission components. The maritime commands are interested, primarily, in five of those key mission components: military contribution to crisis management, sea control, maritime power projection, control and protection of Allied shipping, and flexible response beyond conventional defense. Attain and maintain a favorable air situation is of direct interest to some European land-based elements of the maritime MKCs.

The key mission components characterize the warfighting aspects throughout the spectrum of conflict. Performance capability objectives are developed for each key mission component. The levels of capability generally reflect the same philosophy as that of the strategic objectives matrix in the vertical dimension. Level of conflict aspects are not portrayed individually for each key mission component, because those are determined by the strategic level of conflict with which the key mission component is associated. The key mission components are applicable to all levels of conflict.

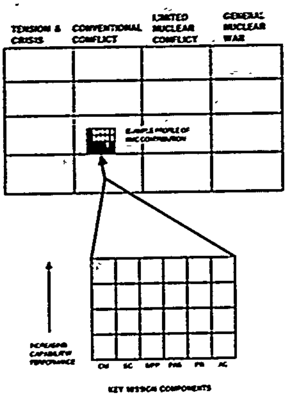


Figure 2 Key Mission Components

Figure 2 portrays the key mission component capability objectives matrix and illustrates an example relationship to a given set of strategic objectives. The individual contribution of any key mission component will be different for the different levels of performance and level of conflict. Thus, for each set of strategic objectives there is a key mission component profile. At the same time the mission components are not unique unto themselves, but have considerable interdependence in a conflict environment. For example, the capability to accomplish, successfully, the protection and control of shipping would be very difficult without some execution of a sea control capability. Related to key mission components are the military functions.

Military functions comprise those essential activities which contribute to or directly support any given key mission component. They are also defined by the conceptual military frameworks. For example, antisubmarine warfare is defined as a military function and is an integral part of the sea control key mission component. Military function capability objectives are developed in the same way as objectives are developed for key mission components. Military function capability objectives profiles exist within key mission component structures similar to the way mission component profiles exist within the strategic objectives matrix. Military functions are analyzed as integral parts of the key mission components rather than individually for this first assessment effort of the maritime MNCs.

The strategic objectives, key mission component objectives, and military function objectives form the backbone of the operational framework. Relationships between key mission components and military functions are established in the conceptual military frameworks; however, the relationships between strategic, key mission component and military functions are further amplified for the maritime effort. Military functions are defined as integral operational, supporting operational, or general supporting. Key functional aspects are included in expanded definitions for key mission components. The maritime commands tend to consider the prosecution of wartime tasks in terms of campaigns and campaign areas rather than specific mission components. However, campaigns tend to be associated with one or more of the NATO specified key mission components. Thus, an initial analysis based upon key mission components would seem to be satisfactory for addressing the key issues associated with C2 for maritime commands. Some expansion of the key mission component objectives is necessary to determine the nature of C2 objectives associated with operational objectives.

COMMAND AND CONTROL REQUIREMENTS DEFINITION

C2 objectives or requirements are derived from operational objectives by analyzing the total context of the decision process as illustrated in Figure 3. While capability objectives may be derived for any set of operational factors (strategic, mission, function, or task), they are derived only for the key mission components for the first maritime effort. The derivation of C2 objectives is accomplished by evaluating key factors of force effectiveness, the battle environment, key decisions made at various command levels, and the C2 activities associated with the decision process. Key C2 implications are derived from an evaluation of these parameters. Eventually mission specific C2 system objectives are established by evaluating the entire decision process in the context of operations supported and C2 implications previously derived.

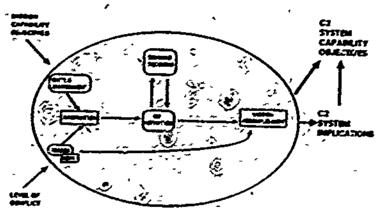


Figure 3 Deriving Command and Control System Objectives

The key force factors (sometimes referred to as key measures of force effectiveness) address those operational activities (or functions) which have the most important contribution to any of the operational levels of capability for any given key mission component. The key force factors are expansions of the operational level of capability descriptions. These factors may include the key functions which best describe the operational requirement. These key force factors are quantified to the degree possible. For example, "conduct antisubmarine warfare to a distance of 600 nautical miles from sea or shore bases" might be a key factor in achieving a given level of the sea control mission component.

Battle environment addresses the nature of the battle which influences C2 and is associated with two strategic elements: level of performance factors and level of conflict factors. At the highest level of defined operational performance, the battle environment tends to be defined as dynamic in nature with both defensive and offensive aspects where the offensive aspects are emphasized. A high degree of integration of effort is required between commands, both vertically and laterally, at the highest level of operational capability. Consequently, planning and force execution tend to approach real time. On the other hand, at the lowest defined level of capability, operations tend to be defined as autonomous with limited lateral coordination. The lowest level of capability would be exercised from plans previously developed in peacetime. The battle environment also addresses the stresses imposed by the selected level of conflict: tension, conventional, or limited nuclear. Key characteristics of the battle environment are stated for each operational and conflict level.

Types of command decisions tend to be a function of level of command. "Command" refers to all commands from the senior theater commander to the force or unit commander. Two levels of command are identified for use in the current analysis: theater and tactical. The theater level of command includes the Major NATO Commanders (MNCs), the Major Subordinate Commanders (MSCs), the Principal Subordinate Commanders (PSCs), and other specified NATO international commanders. The tactical level of command includes the commanders directly associated with tactical forces which are normally national even though they may come under control of NATO commands. Decisions made by the theater levels of command tend to be strategic in nature and deal with resource allocation and long-range actions. Decisions at the tactical levels of command tend to be more closely associated with direction and control of forces on a day-to-day basis. The nature of the decision is likely to influence the type of information required and the timeframe for evaluation and planning. Only the theater level of command will be specifically addressed in the Tri-MNC C2 Plan. However, it is necessary to evaluate the decision process at all levels of command because the decisions and associated C2 activity are highly interrelated across the entire chain of command from theater commander to unit commander.

Four sets of C2 activity are defined: assess, plan, direct/control, monitor. The assess activity contributes to analysis of friendly and enemy capability, evaluation of alternate courses of action, and comprehension of the battle picture. Plan activity includes the generation and selection of options; planning may be formal and written or informal. Direct/control activity comprises those actions directly associated with tasking forces and maintaining effective continued force activity. The monitor activity includes those actions taken to gain information about friendly activities, enemy activities, the environment, and mission progress. The characteristics of the particular activity (e.g. real-time option generation and evaluation) are determined by the level of operational capability supported. Ideally, activity descriptions would be developed for each command; for the current effort, generic descriptions of key activities are developed for the defined theater and tactical command levels.

Evaluation of the foregoing key factors, to include the specified operations, reveals certain key C2 implications for each level of operational capability. Key C2 implications are derived for three levels of conflict: tension and crisis, conventional conflict, and limited nuclear war. These key implications tend to address those factors which are critical or essential from a C2 perspective to the successful accomplishment of operations. Implications are dervied for source data (primarily enemy intelligence), information processing, information exchange between commands, and facilities to support command and planning. Implications for procedures are also evident but are not explicitly stated because the effort is oriented toward physical support elements. Key C2 implications developed in this fashion should agree with those developed by other studies or those intuitively obvious to the command. The involvement of operational and technical staff members in this process is essential to obtain the correct operational perspectives and definition and to ensure the implications derived are, in fact, real values. Derivation of C2 system implications is a precursor to the derivation of C2 system capability objectives.

Evaluation of the total operational context and associated C2 factors is used to develop C2 system capability objectives. Specific C2 system capability objectives are developed for each key mission component and for each level of operational capability within the mission component. The first derivation of objectives results in a set of first order C2 system capability objectives which are stated in operational terms. For example, "Command and control information system (CCIS) to generate and evaluate employment options in real time" may be one system objective. "Very high capacity communications to support large volumes of secure data and video transmission" may be a communications system objective within the set of C2 system capability objectives. System capability objectives are developed and stated for the four system physical elements: war headquarters, communications, command and control information systems (CCIS), and sensors/source data (for warning installation). A caveat is needed here. Sensors are primarily a national responsibility. For the most part, only sensors associated with air defense operations are acquired as NATO equipment. However, fusion systems will be procured under NATO infrastructure funding. The term "source" is used to portray the vital need for sensor data even though such data must be a national input into the NATO intelligence system. Any assessment of capability will necessarily address the capability to provide adequate data; however, the solutions to resolve deficiencies will address only those system elements which can be acquired by NATO. The C2 capability objectives provide an operational basis for assessing the ability of the C2 system to support specific operational objectives and missions.

COMMAND AND CONTROL SYSTEM CAPABILITY ASSESSMENT

The C2 system capability objectives provide an operationally oriented basis for assessing the capability of the C2 system. More detailed C2 system capability objectives are derived from the first order operational statements by using a defined standard set of system attributes. For example, connectively, capacity, and reliability are defined performance attributes for physical systems. The degree of performance associated with any given attribute changes as the level of operational capability changes. Survivability attributes such as electronic survivability are also defined. In relation to the operational structure, performance attributes tend to be vertically related, while survivability attributes tend to be horizontally related. The use of standard attributes, operational and system, allows consistent evaluation of capability.

The C2 system physical elements must be cataloged and described. Each command reviews those system elements that support the command and describes the capability of those elements in terms of contributing physical systems (e.g. transmission media, switches, ADP equipment, bunkers) and the attributes of the individual physical systems (e.g. reliability, survivability). A C2 system physical element descriptive catalog is being developed and will be maintained and issued as a centrally controlled data base. The physical systems descriptions provide a means for comparison to the desired goals stated in the capability objectives.

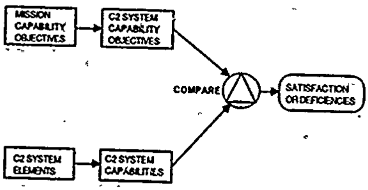


Figure 4 C2 Capability Assessment

The capability of the C2 system to support a given level of mission capability objective is assessed by comparing the capabilities of the physical elements with the capabilities needed as stated in the capability objectives. Cataloged capabilities descriptions of the physical systems tend to reveal the level of operational capability a particular system can support. Some comparisons will be multifaceted. For example, the capability of communications elements must be assessed by evaluating individual components (e.g. transmission media, switching, services) and the aggregate of those components (e.g. networks). The assessment by comparison process is not a "cookbook" type evaluation. Considerable military operational and technical judgment must be used in determining the capability of physical elements to support a given capability level and the aggregate capability of the physical elements to provide the necessary and sufficient support required. Assessment is performed independently for each level of conflict so that stress or survivability factors are readily evident.

The comparison of C2 system capabilities with C2 system capability objectives indicates the degree to which the C2 system can support operational objectives and the deficiencies in capability to support any given level of operational capability. The deficiencies are portrayed in both operational and technical terms. Based upon command priorities, improvements to C2 system capabilities can be developed to provide the degree of C2 system support required by the strategy and force structure.

SYSTEM IMPROVEMENTS / SOLUTION OPTIONS DEVELOPMENT

Improvements to the C2 system will be developed by constructing improvement packages consistent with an overall architectural goal. The C2 system architecture is a description of the major structure or design features of a group of interrelated and interconnected systems, subsystems, equipment, devices, components or other elements. The NATO Communications and Information Systems Agency (NACISA) is responsible for development of the overall NATO C2 system architecture. C2 system improvement packages developed by the commands must be consistent with the overall NATO system architecture. The concept of developing packages is new in the NATO planning environment. Packages are sets of logically related system projects designed to achieve specific mission, functional, or regional goals.

The construction of C2 system improvement packages can be accomplished in a series of logically related steps. Initially, system requirements are established by analysis of operational needs. Operational objectives are established by the command and the current capability is evaluated to determine ability to support operational objectives. The identification of deficiencies is basically the mission-oriented assessment previously discussed. A vital factor must be considered in the development of improvements to the C2 system; this factor should be considered even during the assessment. The command must account for the capability to satisfy operational objectives with the force structure, both current and proposed. In an environment of scarce resources, it is essential that the C2 system be consistent with the force structure; that is, it should provide the capability to support operational objectives equal to the force capability to accomplish operational objectives, but not more. Once deficiencies to support the forecast force structure are determined, the development of system solutions may progress.

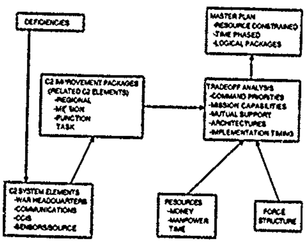


Figure 5 Developing C2 System Improvements

C2 system physical element improvement projects are developed to solve specific problems. The design of new or additional C2 system physical element capability considers the potential of various means of satisfying requirements. Consideration is given to expansion or modification of existing or programmed system elements. New system design is suggested where required. Consideration is also given to the use of common-user components or, alternately, dedicated user components. Package development also entails an examination of multiple mission requirements and solutions. For example, the capability developed to support one set of key mission component objectives may also be applicable to another key mission component set of objectives with some marginal increase or addition in performance. Based upon command perspective and emphasis, physical element improvements will be grouped in logically related packages. It is likely that some packages will be NATO-wide common-user sets.

The initial improvement package development will be unconstrained by budget considerations and will address the capability needed to provide the required C2 support to the forecast force structure. Subsequent tradeoff analysis will be conducted based upon forecast resources and review of the most cost-effective means of providing capability resulting in a resource constrained set of improvement packages. The development of a resource constrained improvement plan will depend upon both resource constraints and command mission emphasis.

IMPROVEMENT PLAN DEVELOPMENT, COORDINATION AND, IMPACT ASSESSMENT

The development of improvement plans will be multi-faceted. Each of the MNC subordinate regional commands will develop regionally oriented C2 system improvement plans which will be included in the final Tri-MNC C2 Plan. MNCs will coordinate the efforts of their subordinates. Coordination between MNCs will occur throughout the planning process. The MNCs will develop a joint C2 system improvement plan which encompasses all the requirements for the military portion of the C2 system once the regional plans have been submitted. Continuous coordination between MNCs, their subordinates, and designated NATO agencies will be essential for developing both the common user and dedicated user aspects of the plans.

The assessment of the impact of C2 system improvements on mission objectives is integral to the development of a resource constrained improvement plan. Operational objectives are reviewed and priorities established for improvement to specific mission components during plan development. Resources are applied to satisfy priority requirements and packages tailored to both the need and available resource. Packages are developed with specific change to support operational capability in mind. Thus, the operational impact of the improvement plan is immediately evident.

Efforts by the NATO commander will culminate in the publication of the Tri-MNC C2 Plan. This C2 plan will state the military command and control requirements; current deficiencies in capability to meet requirements; time-phased, resource constrained proposals to implement improvements; and the capability achieved to support operational goals by implementation of those proposals.

CONCLUSIONS

The mission-oriented approach to C2 planning is being introduced into the NATO C2 planning activity. The maritime Major NATO Commands have chosen to institute a structured approach which compares C2 system capability objectives with achieved C2 system capabilities to determine the ability of the C2 system to support any given level of operational capability. Planning is directed toward evaluation and improvement of capability to support specified key mission components. The current planning effort is directed toward improving the C2 system for the theater level of commands; however, tactical requirements and the interfaces between theater and tactical C2 systems are considered to provide an accurate view of total NATO C2 capability to support the NATO missions. Improvements to the C2 system for the theater level of command will be reflected in the Tri-MNC C2 Plan. A mechanism for recommending improvements to the C2 systems supporting the tactical level of command is yet to be developed. A salient feature of the particular maritime approach is the ability to conduct successive iterations at greater levels of detail once an evaluation identifies key issues. In summary, the most important aspect of the mission-oriented approach by the maritime MNCs is the ability to relate C2 system requirements and capabilities to specific mission capabilities on an always consistent basis.

1988 SYMPOSIUM ATTENDEES

Harold B Aldrich
US Army - Ft Leavenworth
231 Third Avenue
Leavenworth, KS 66048
Tel: 913-684-3413

Stamos K Andreadakis
MIT
Room 35-417/LIDS
Cambridge, MA 02139
Tel: 617-253-2346

Michael Athans
MIT
Room 35-406/LIDS
Cambridge, MA 02139
Tel: 617-253-6173

Susan L Bales
Office of the Chief of Naval Operations
CNO Executive Panel
4401 Ford Avenue
lexandria, VA 22302-0268

Edward C Brady
he MITRE Corporation
525 Colshire Drive
McLean, Virginia 22102
el: 703-883-6201

Peter Breitkopf
MC Corporation
entral Engineering Laboratories
205 Coleman Avenue
anta Clara, CA 95052
el: 408-289-2662

onald E Brown
nstitute for Parallel Computation
chool of Eng and Applied Sciences
hornton Hall
niversity of Virginia
harlottesville, VA 22901
el: 804-924-5393

Rex V Brown
Decision Science Consortium, Inc.
1895 Preston White Drive
Suite 300
Reston, VA 22091
Tel: 703-620-0660

Alicia Browner
Lockheed Austin Division
6800 Burleson Road
O/T4-20 B/30E
Austin, TX 78744
Tel: 512-448-5159

Lois M Brunner
Naval Post Graduate School
Monterey, CA 93943-5000
Tel: 408-646-2618

LCDR Rex A Buddenberg
Commandant, U.S. Coast Guard
(G-TES-1)
Washington, DC 20593
Tel: 202-267-1206

Adam Caromicoli
ALPAHTECH, Inc
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01730
Tel: 617-273-3388

Jan Chandler
Naval Ocean Systems Center
Code 412, 271 Catalina Blvd.
San Diego, CA 92152
Tel: 619-553-3998

Shi-Chung Chang
National Taiwan University
Dept of Electrical Engineering
Taipei, Taiwan
Tel: 886-2-3415251

1988 SYMPOSIUM ATTENDEES

Gerald A Clapp
Naval Ocean Systems Center
Code 805, 271 Catalina Blvd.
San Diego, CA 92152
Tel: 619-553-1165

COL Gary Q Coe
The Pentagon
OJCS/J6
Room 1D841
Washington, DC 20301-5000
Tel: 202-697-8091

Stuart L Crawford
Advanced Decision Systems
1500 Plymouth Street
Mountain View, CA 94043-1230
Tel: 415-960-7300

John P Crecine
The Georgia Institute of Technology
Office of the President
225 North Avenue, N.W.
Atlanta, Georgia 30332
Tel: 404-894-5051

Ronald L Crepeau
Naval Ocean Systems Center
Code 412, 271 Catalina Blvd.
San Diego, CA 92152-5000
Tel: 619-533-3990

Allan H Curry
FORSCOM, C4-J6
Atlanta, GA 30330
Tel: 404-752-4114

William J Dejka
ISA
10260 Vista de la Cruz
La Mesa, CA 92041

Marc D Diamond
FMC Corporation
Advanced Systems Center
1300 South Second Street
P.O. Box 59043
Minneapolis, MN 55459-0043
Tel: 612-337-3229

Robert Duchatellier
The MITRE CORPORATION
145 Wyckoff Road
Eatontown, NJ 07724
Tel: 201-544-1414

Marguerite M Dymond
AMSSA (Air Warfare Div.)
Director AMSAA
AMXSY-AIC (Dymond)
Aberdeen Proving Grounds
Maryland 21005-5071
Tel: 301-278-6356

John T Egan
U.S. Naval Research Laboratory
NRL, Code 5703.10
Tactical Electronic Warfare Div.
4555 Overlook Avenue, SW
Washington, DC 20375-5000
Tel: 202-767-0762

Eliott Entin
ALPHATECH, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01803-4901
Tel: 617-273-3388

Dr. Anthony Ephremides
University of Maryland
Electrial Engineering Dept.
Engineering Research Bldg. (093)
College Park, MD 20742
Tel: 301-454-6871

Boyd C Fair
SRI International
Information Sciences and Technolog
333 Ravenswood Avenue
Menlo Park, CA 94025
Tel: 415-859-3136

1988 SYMPOSIUM ATTENDEES

CPT Anderson L Hamilton
U.S. Navy
P.O. Box 15898
Arlington, VA 22215
Tel: 703-971-1470

Lewis F Hanes
Westinghouse R&D Center
1310 Beulah Road
Pittsburgh, PA 15235
Tel: 412-256-2682

Bradley J Harnish
RADC/DCLD
Griffiss AFB, NY 13441-5700
Tel: 315-330-7751

Henry A Heidary
Hughes Aircraft Company
5901 E. Cowboy Circle
Anaheim, CA 92807
Tel: 213-616-6972

Eric Heilman
Ballistic Research Laboratory
Room 212, Bldg.394
Aberdeen Proving Ground
Maryland 21005-5066
Tel: 301-561-4595

Richard A Helfman
Ballistic Research Laboratory
SLC BR-SE-C
Aberdeen Proving Ground
Maryland 21005-5066
Tel: 301-278-6657

Paul J Hiniker
Defense Communications Agency
C4S/A720
Arlington Hall Station
Arlington, VA 22212-5410
Tel: 703-692-6241

Gerald Hopple
George Mason University/IIS
ISSE/SIIE
Fairfax, VA 22030
Tel: 703-323-3725

Patrick A Hosein
MIT/LIDS
Rm. 35-409
777 Massachusetts Avenue
Cambridge, MA 02139
Tel: 617-253-5992

R. Douglas Howson
AT&T Bell Laboratories
1 Whippany Road, 14D-425
Whippany, NJ 07981
Tel: 201-386-4579

Lester Ingber
Naval Postgraduate School
Physics Dept., Code 61IL
Monterey, CA 93943
Tel: 408-646-2687

Ronald M James
ALPHATECH, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01821
Tel: 617-667-7863

Victoria Y Jin
MIT
Room 35-415/LIDS
Cambridge, MA 02139
Tel: 617-253-2832

Leland Joe
Rand Corporation
Systems Sciences Department
P.O. Box 2138, 1700 Main St.
Santa Monica, CA 90406-2138
Tel: 213-393-0411

1988 SYMPOSIUM ATTENDEES

Bela Feher
Naval Ocean Systems Center
Code 446
San Diego, CA 92152-5000
Tel: 619-553-9226

Philip Feld
Defense Systems Inc.
7903 Westpark Drive
McLean, VA 22102-4294
Tel: 703-883-1063

Benjamin Fisher
Hughes Aircraft Co.
P.O. Box 3310
Bldg 618/MS L305
Fullerton, CA 92634
Tel: 714-998-6665

Romulus Fratila
SPAWAR
National Center 1, Rm. 11 W 18
Code 313B
Washington, DC 20363-5100
Tel: 202-692-6414

Sheldon Gardner
Naval Research Laboratory
Code 8345-25
4555 Overlook Avenue, S.W.
Washington, DC 20375-5000
Tel: 202-767-3846

CDR Paul E Girard
Naval Ocean Systems Center
ASW Systems Division
Code 62
San Diego, CA 92152-5000
Tel: 619-553-2113

Irwin R Goodman
Naval Ocean Systems Center
Code 421, Seaside Bldg. 600
Command and Control Dept.
San Diego, CA 92152-5000
Tel: 619-553-4014

William E Gorham
The MITRE Corporation
7525 Colshire Drive
McLean, Virginia 22102-3481
Tel: 703-883-7413

Charles Graff
U.S. Army CECOM
AMSEL-RD-C3-AC-B
Ft. Monmouth, NJ 07703
Tel: 201-532-0164

Inara Gravitis
SAIC
1710 Goodridge Drive
McLean, VA 22102
Tel: 703-734-4096

Robert W Grayson
The MITRE Corporation
7525 Colshire Drive
MS Z341
McLean, VA 22102-3481
Tel: 703-883-7441

David B Griswold
The MITRE Corporation
7525 Colshire Drive
MS Z270
McLean, VA 22102
Tel: 703-883-7345

James E Haile
IFFN JTF
Kirtland Air Force Base
New Mexico 87123
Tel: 505-844-5994

Stanley M Halpin
ARI Field Unit
PO Box 3407
Ft. Leavenworth, KS 66027-0347
Tel: 913-684-4933

1988 SYMPOSIUM ATTENDEES

Stuart Johnson
National Defense University
Director, INSS/SCDC
Ft. Lesley J. McNair
Bldg. 59
Washington, DC 20016
Tel: 202-475-1094

Demetrios Kazakos
University of Virginia
Electrical Engineering Department
Thornton Hall
Charlottesville, VA 22901
Tel: 804-924-6081

Richard T Kelly
Pacific Science & Eng Group, Inc
5090 Shoreham Place
Suite 206
San Diego, CA 92122-3963
Tel: 619-535-1661

David Kleinman
University of Connecticut
Dept. of Electrical & Systems Eng.
U-157, Room 312
260 Glenbrook Road
Storrs, CT 06268
Tel: 203-486-3066

Cynthia Kohn
1563 Feather Avenue
Thousand Oaks, CA 91360
Tel: 805-495-2737

Patricia Langendorf
RADC/IRDW
Bldg 240
Griffiss AFB, NY 13441-5700
Tel: 315-330-3126

Ronald W Larsen
Naval Ocean Systems Center
Artificial Intelligence Technology
Branch, Code 444
San Diego, CA 92152
Tel: 619-553-4142

Prof. John Lehoczky
Carnegie Mellon University
Department of Statistics
Baker Hall 232
Pittsburgh, PA 15213
Tel: 412-268-8725

Dr. Alexander H Levis
MIT
Room 35-410/LIDS
Cambridge, MA 02139
Tel: 617-253-7262

Jack M Ligon
Commandant (G-TT) USCG
Washington, DC 20593
Tel: 202-267-1261

Kenneth M Lindberg
The MITRE CORPORATION
145 Wyckoff Road
Eatontown, NJ 07724
Tel: 201-544-1414

Jean MacMillan
ALPHATECH, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01803
Tel: 617-273-3388

Kenneth R Madsen
ALPHATECH, Inc.
2 Burlington Executive Center
11 Middlesex Turnpike
Burlington, MA 01803-4901
Tel: 617-273-3388

Gerald S Malecki
Office of Naval Research
Code 1142PS
800 N. Quincy Street
Arlington, VA 22217
Tel: 202-696-4741

1988 SYMPOSIUM ATTENDEES

Philippe J Martin
GIAT (France)
Division T, AMX-APX
13 Route de la Miniere
78013 Versailles FRANCE
Tel: 1-3949-3046

Glenn J Marum
AT&T Bell Laboratories
1 Whippany Road
Whippany, NJ 07981
Tel: 201-386-4763

Israel Mayk
US Army CECOM, Commander
Center for C3 Systems
AMSEL-RD-C3-IR-1
Fort Monmouth, NJ 07703-5000
Tel: 201-544-4996

Michael R McFarren
AAMRL/HED
Wright-Patterson AFB
Ohio 45433
Tel: 513-255-8822

Stephen R McReynolds
Applied Technology Associates, Inc.
400 Gordon Drive, Ste. 504
Lionville, PA 19353
Tel: 215-524-1590

Byron B Moore
Defense Communications Agency
Advanced Technology Office
1440 Isaac Newton Square (N)
Reston, VA 22090
Tel: 703-437-2506

CMDR Newton Morgan, JR
US Space Command (Stop 35)
Systems Integration Staff (J4/6S)
Peterson AFB, CO 80914-5003
Tel: 719-554-5971

Thomas Moriarty
United Technologies Systems Analysis
Suite 850, 2000 Corporate Ridge
McLean, VA 22102
Tel: 703-749-3853

Laura C Morrison
Lockheed Austin Division
Buiding 30F North
6800 Burleson Road, T5-40
Austin, TX 78744-1016
Tel: 512-448-5126

Carl N Mutchler
The MITRE Corporation
MS-W932
1820 Dolley Madison Blvd.
McLean, VA 22102
Tel: 703-883-5527

James B Nelson
Lockheed Missles & Space Co., Inc.
O/TO-10, B/30E
6800 Burleson Road
Austin, TX 78744
Tel: 512-448-9488

J. Nievergelt
University of North Carolina
at Chapel Hill
Computer Science Department
CB #3175, Sitterson Hall
Chapel Hill, NC 27599-3175
Tel: 919-962-1800

Michael O'Connor
Decision Science Consortium, Inc.
1895 Preston White Drive
Suite 300
Reston, Virginia 22091
Tel: 703-620-0660

Jason D Papastavrou
MIT
Room 35-407/LIDS
Cambridge, MA 02139
Tel: 617-253-3631

1988 SYMPOSIUM ATTENDEES

C. Joseph Pasquariello
Defense Communications Agency
Code A130
Washington, D.C. 20305
Tel: 202-692-4761

Kshemendra N Paul
Techno-Sciences Inc.
7833 Walker Drive, Suite 620
Greenbelt, MD 20770
Tel: 301-345-0375

Steven L Pendergast
Hughes Aircraft Company
Ground Systems Group
P.O. Box 3310
Building FU 604, MS B111
Fullerton, CA 92634
Tel: 714-732-2579

Didier M Perdu
MIT
Room 35-417/LIDS
Cambridge, MA 02139
Tel: 617-253-2346

Joe Provenzano
JPL/CALTECH
MS-510-264
4800 Oak Grove Drive
Pasadena, CA 91109
Tel: 818-397-9991

CPT Dale V Raebel
OASD (C3I), DASD (C3), MA&E
Rm 2C252, THE PENTAGON
Washington, DC 20301-3040
Tel: 202-697-4540

Al Roberts
The MITRE Corporation
1820 Dolley Madison Blvd.
McLean, VA 22102
Tel: 703-883-5691

Joseph Rockmore
Advanced Decision Systems
1500 Plymouth Street
Mountain View, Ca 94043-1230
Tel: 415-960-7300

R. Jay Roland
Roland & Associates Corporation
500 Sloat Avenue
Monterey, CA 93940
Tel: 408-373-2025

Julie Rosen
SAIC
1710 Goodridge Drive
McLean, VA 22102
Tel: 703-556-7354

Izhak U Rubin
UCLA
Dept of Electrical Engineering
School of Engr. & Applied Science
6731 Boelter Hall
Los Angeles, CA 90024
Tel: 213-825-2327

Larry Sadler
Naval Ocean Systems Center
7937 Hemingway Avenue
San Diego, CA 92120
Tel: 619-553-3969

Michael D Salomone
Carnegie-Mellon University
College of Humanities
& Social Sciences
Pittsburgh, PA 15213-3980
Tel: 412-268-2830

Daniel Serfaty
ALPHATECH, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01803-4907
Tel: 617-273-3388

1988 SYMPOSIUM ATTENDEES

LTCOL Walter W Sevon. Jr.
HQ US Southern Command
PSC Box 315
APO Miami, FL 34007
Tel: 407-282-4452

Richard R Shively
AT&T Bell Laboratories
Advanced Technology Systems
Laboratory
Whippany Road, Room 14A219
Whippany, NJ 07981
Tel: 201-386-4715

CPT Richard J Simchik
HQ, US Army Signal Center
Attn: ATZH-CDC
Fort Gordon, GA 30905
Tel: 404-791-2800

S. Christian Simonson
Lawrence Livermore National Lab.
Conflict Simulation Center
P.O. Box 808, L-315
Livermore, CA 94550
Tel: 415-422-6569

Dr Harold W Sorenson
HQ USAF/CCN
Room 4E 320
The Pentagon
Washington, DC 20330-5040

Michael Sovereign
SHAPE Technical Center
Oude Waalsdorpersweg 61
P.O. Box 174, 2051 C.D. The Hague
The Netherlands
Tel: 31-70-142371

Greg Stachnick
Advanced Decision Systems
1500 Plymouth Street
Mountain View, Ca 94043-1230
Tel: 415-960-7300

Stuart H Starr
Mitre Corporation
Director of Plans
7525 Colshire Drive
McLean, Va 22102-3481
Tel: 703-883-5494

Alan D Sussman
SAIC
MS 33
10260 Campus Point Drive
San Diego, Ca 92121
Tel: 619-546-6459

Ricki Sweet
Sweet Associates, Ltd.
2001 Jefferson Davis Hwy.
Suite 1010
Arlington, VA 22202
Tel: 703-979-1455

David D Sworder
University of California,
San Diego
Department of AMES, MS R011
LaJolla, CA 92093
Tel: 619-534-4498

Robert R Tenney
ALPHATECH, Inc.
2 Burlington Executive Center
111 Middlesex Turnpike
Burlington, MA 01803
Tel: 617-273-3388

Paul Torelli
Defense Communications Agency
Chief, C3 Planning Division, Bldg A
Arlington Hall Station
Arlington, VA 22212-5410
Tel: 202-692-6241

COL James Toth
National Defense University
Industrial College of the Armed Force
Ft. Lesley J. McNair
Washington, DC 20319
Tel: 202-475-1848

1988 SYMPOSIUM ATTENDEES

CPT Frank Tryon
National Defense University
National War College
Ft. Lesley J. McNair
Washington, DC 20319
Tel: 202-475-1743

Vice Admiral J. O Tuttle
Chief, Modeling & Analysis Division
JCS-J8F
The Pentagon
Washington, DC 20301-5000
Tel: 202-695-5478

Harry Van Trees
MACOM
5004 Clairborne Drive
McLean, VA 22101
Tel: 703-847-5417

Willard S Vaughan
Office of Naval research
Director, Cognitive and Neural
  Sciences Division
800 N. Quincy Street
Arlington, VA 22217
Tel: 202-696-4505

Ronald Virostek
The MITRE Corporation
7525 Colshire Drive
McLean, Virginia 22102
Tel: 703-883-6266

Daniel Wagner
Naval Postgraduate School
Operations Research Department,
212 Root Hall
Monterey, CA 93943-5000
Tel: 408-646-3215

James T Walton
MIT
Room 35-407/LIDS
Cambridge, MA 02139
Tel: 617-253-3631

Yangwei Wang
Iowa State University
201 Coover Hall
Ames, IA 50011
Tel: 515-294-9472

Patrick J Ward
Army Material System Analysis Activity
2214 Apolle Terrace
Havre de Grace, MD 21078
Tel: 301-278-6601

Harold Waters, II
Research Triangle Institute
Herbert Bldg, CDSR
P.O. Box 12194
Research Triangle Park,
North Carolina 27709-2194
Tel: 919-541-6808

David P Watrous
Presearch, Inc.
8500 Executive Park Avenue
Fairfax, VA 22031
Tel: 703-876-6400

Howard W Williams
The MITRE Corporation
7525 Colshire Drive, MS Z305
McLean, VA 22102-3401
Tel: 703-883-6465

Richard P Wishner
Advanced Decision Systems
1500 Plymouth Street
Mountain View, CA 94043-1230
Tel: 415-960-7320

A.E.R. Woodcock
Synectics Corporation
10400 Eaton Place, Suite 200
Fairfax, VA 22030-2208
Tel: 703-385-0190

1988 SYMPOSIUM ATTENDEES

Ronald E Wright
Ferranti Computer Systems, Ltd.
Priestwood, Workingham Road
Bracknell, Berkshire
RG121BA, England
Tel: 0-344-427171

Mary Wuest
205 Yoakum Parkway
Alexandria, VA 22304
Tel: 703-461-7947

AUTHOR INDEX