AR-009-627

DSTO-GD-0079

DSTO Workshop on Intelligent
Decision Support Systems
Held 7 July 1995, Pyrmont

G. Gibbon, Editor

19970307 058

DEPARTMENT ◆ OF DEFENCE
DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION

Mar-18

DSTO Workshop on Intelligent Decision Support Systems, Held 7 July, 1995, Pyrmont

G. Gibbon, Editor

Weapons Systems Division
Aeronautical and Maritime Research Laboratory

DSTO-GD-0079

Approved for public release

Contents

1. Introduction

1.1 The Workshop

The DSTO Workshop on Intelligent Decision Support Systems was held at DSTO Pyrmont on 7 July 1995. Its key objective was to get views on research areas which have the potential to most contribute to the enhancement of performance of sensor suites on the Australian Defence Force's major platforms. It has long been recognised that the individual sensors on a platform are a mature technology level, but that the integration of their information and its presentation to the platform commander is not as advanced.

The issue is of extracting greater operational or tactical support from the entire suite in a situation where there are multiple sensors contributing to situation awareness. The technology of interest is the integration and automated support facilities that occur after the main signal processing. The actual detection and processing of signals in a noisy environment is a mature technology, often achieving close to theoretically optimum results. The other end, involving information representation and management is in its infancy, and this is where this area is providing scope for progress.

This is of rapidly growing interest in Defence circles since this has been identified as a common bottleneck in many applications. This is seen as a research area that offers great promise for Defence applications over the next decade, where the outlook is very good for at least evolutionary advancements by developing existing research thrusts.

Decision Support techniques cover these issues comprehensively by addressing issues such as machine representation of these aspects of human knowledge, automated reasoning about incoming data, human factors regarding display and structure of prompts and suggestions, background calculations, etc.

The Workshop obviously had to attract a multidisciplinary participation, because Decision Support Systems derive their power by being fully integrated in the work place of the human operator. The broad mix of the fifty plus participants assembled at Pyrmont -- ADF Officers with experience on platforms, R&D Contractors credited with systems recently commissioned in practice, and researchers from DSTO, Universities and Industry -- proved an extraordinary catalyst for exposing areas of research importance which would progress the integration of sensors on actual platforms such as the P3C surveillance aircraft and the Collins Class submarines. Discussion was very lively and researchers went away invigorated, having seen real application requirements to which they could turn their mind.

The lesson for researchers striving to improve the performance of sensor suites is that there are great gains to be made by concentrating on the interaction between the automation techniques and the idiosyncrasies of the domain problem. The recurring subject of the Workshop was the fact that the behaviour of a sensor system is rarely correctly matched to the workflow requirements of the operator. Even the most advanced systems have inappropriate design -- "design for experts by experts" -- when, operationally, they are used by relatively unskilled officers. The lesson that the day underlined was the need for a systems approach with a strong human factors

component. If this taken with existing technologies, then considerable advances can be made with existing automated reasoning and information representation and display technologies. There are nevertheless clear challenges for improving the inherent effectiveness of these technologies. The basic foundations of these technologies are going to experience only slow evolutionary development, despite having a worldwide research force applied to them (which will doubtless lead to a wealth of hype terms such as "intelligent agents").

2. The Workshop presentations

2.1 Describing the domain

The workshop was launched by two military speakers chosen for their extensive experience with equipment, in situations where the behaviour of their equipment would be improved greatly by adding forms of automation which are now available and feasible in the near term.

Both speakers had strong feelings about the shortcomings of the current equipment. These ranged from the case where the system is simply lacking in basic features that are taken for granted in other outfits, to the case where designers have added features that employ the latest technology, but the feature does not suit the task. The latter case is a design problem, where the nature of the work procedures have not been considered when installing the feature.

CMDR Chris Donald talked about the realities of operational conditions on the unusually wide variety of platform type with which he is familiar - the P3C, the Oberon submarine and S2 trackers. Stressors on the operator and resulting discomfort have unfortunately not been adequately taken into account in sensor design. He emphasised that the "irrelevant" technical details are often inappropriately apparent to operators, such as in naming of function buttons on a sensor panel.

WGCDR Rick Owen gave a fast moving account of the phases of a typical strike mission, indicating where helpful automation may be conceivable. There is plenty of scope in the Mission Planning and Mission Analysis phases as these are done before and after the flight in an office environment with relatively generous time-frames. On the other hand there are many aspects of the workload during the flight where Decision Support would be very welcome, but with the proviso that it must be timely, accurate and reliable.

Following these two presentations, CDR Donald and WGCDR Owen formed a panel which drew a lively questions and discussion session. The dominant theme of the questions was again on the mismatch between requirements and delivered functionality. Researchers in the audience were keen to hear from the experts how actual system functionality could be enhanced. Much of the questioning was at a detailed level which presupposed a lot of knowledge of the sensor fit on the particular platforms. The problem of finding a development process for sensor suites which would eliminate the mismatch was also tackled, with the consensus being that close involvement of people with the appropriate military knowledge was needed in Defence project teams right from the design phase. However, the actual experience levels of the final users must always be kept in mind.

2.2 Research Protypes

Tristan Chiu from BHP IT described exploratory work to resolve a potential mismatch between operator knowledge and that required to optimally operate one of Australia's most important surveillance systems, the Jindalee Over-The-Horizon Radar Network (JORN). A software application has been developed to capture and disseminate the radar tasking knowledge from DSTO scientists who developed the radar. (First appeared in Proc. ANZIIS'94, Brisbane, Nov 1994)

Gil Tidhar from the Australian Artificial Intelligence Institute described work being done with DSTO's Air Operations Division on air mission modelling which again is tryng to gain advantage from a close relationship with the domain. A reasoning system is being prototyped which uses detailed knowledge of the teams and team tactics during specific missions for fighter aircraft. (This paper is published with permission from Gordon and Breach Science Publishers, SA, Australia)

Keith Mason from DSTO described a software prototype to assist in the task of identifying and tracking radar emitters from radar signals intercepted by a passive sensor system. The key feature of his approach is that it utilises knowledge of the performance (and in particular the limitations) of the sensor processing algorithms as well as knowledge of environmental data (such as from geographic information systems) and knowledge of radar emitter characteristics to re-associate reports of radar emitters that have been fragmented by the sensor system.

2.3 Supporting Research

Frada Burstein talked about work at Monash University on knowledge modelling as a part of Intelligent Decision Support Systems. Her emphasis was that the "intelligence" in IDSS must include a knowledge base component which provides some memory aids for the decision maker and which "learns" from the decision maker's experience. The type of memory aids being invetigated were of the form of a case base of past decisions. Her talk was rounded off by a discussion of similarity measures between cases which are essential to the case based reasoning approach.

Paul Compton reported on work done with medical staff developing a knowledge based system without resorting to the usual expensive process of knowledge engineering and refinement prior to use of the system. The approach to developing knowledge bases (KB), (Ripple Down Rules (RDR)) has been developed which allows for incremental development and modification. The essential feature of the approach is that when a human (or a DSS) provides advice for a situation, and someone else (perhaps an expert with superior expertise) disagrees with the advice, the expert will highlight features in the situation which distinguish it from one where the advice may have been appropriate. If the advice provided by an RDR system is considered incorrect by the user, the RDR system presents the user with a list of differences between the present situation and one or more other situations where the advice would have been correct. The user selects the relevant features and indicates the appropriate advice and the system corrects its knowledge accordingly. The situation is itself stored to be used to provide for further lists of differences.

Andrew Blair presented a comparison of five major formal methodologies for designing IDSSs. To compare these five design methodologies he used a framework to gauge what each of these methodologies do. He showed that these methodologies are quite different in their approach for designing IDSSs and also that these methodologies differ greatly in the amount of support which they provide. The framework presented decomposes the IDSS design methodologies according to some predefined phases that support the development process. The resulting comparative study assists IDSS developers in understanding what support can be gained from using each of the five major design methodologies and in choosing the correct one for their project.

2.4 Maturing Systems

Phil Silver talked about improvements which will provided by the current project to refurbish the sensor and data processing capabilities on the Australian P-3 maritime surveillance aircraft. Better displays, will be available both at the sensor stations and the integrated picture for the tacho. The information representation will use colour and icons. The data management system will allow future expansion, and there are greatly improved target detection and track formation processing and management subsystems.

Ian Croser described a system that the Australian company CEA Technologies has recently sold to the US Navy. The Graphical Data Fusion System for the Mobile Inshore Undersea Warfare System Upgrade provides capabilities for Sensor Management, Track Management, Mission Plannning/Analysis, and C3I Support. This is achieved by addressing detail at every level of the operator functions. The GDFS receives data from the underwater acoustics and a number of radars that provide tracking and ESM.

The GDFS system provides a large screen display which combines input from video cameras and underwater magnetic tracking sensors. There are three windows: one image and two map-based. One of the maps is an overview map providing context. The software generates icons for displaying tracks on either of the map windows which the user has the option of suppressing. The innovation of MIUW is the extent to which human factors have been taken taken into account in the design. The operator is offered total flexibility in the form of display at any given time so as to accommodate any particular operational situation.

5

Brad Noakes reported on the US Navy's test bed for evaluating recent developments in the design of a Decision Support System (DSS) for enhancing tactical decision making under difficult conditions. The Decision Making Evaluation Facility for Tactical Teams (DEFTT) Laboratory at NRaD in San Diego is a six station test bed environment that simulates the Anti Air Warfare (AAW) computer work stations of a shipboard combat information centre. There is an Australia/US collaboration on tactical decision research, where a number of RAN officers were used as subjects in the DEFTT Laboratory. Results from these collaborative studies were presented, in addition to recommendations for areas of further collaborative research. Decision support tools developed at NRaD were evaluated for their applicability for the RAN and other ADF Command Support System projects.

2.5 Related Paper (where the authors were unable to attend)

Neil Fulton tabled a paper covering the issue of Situation Awareness in the Air Traffic Control domain. The specific problem of collision avoidance during cruise is analysed in terms of the effectiveness of various aspects of the ATC system. The paper strongly suppports the view that the Human Factors considerations are of prime importance in shaping the effectiveness of the ATC system, even though there is a heavy reliance on sensor performance to provide the basic functionality. This is supported by noting the long history of attempts to reduce semantic distance, a measure of the mental effort required for the pilot to interpret the readings of instruments relative to the requirements of the present situation.

He also provides an analysis of collision avoidance in terms of visual acquisition, where the reliability of the human visual scan pattern can be quantified for different aviation contexts. There is also discussion of the benefits of redundancy, and there is some progress toward quantifying this, as well as a discussion of performance metrics for separation.

2.6 The Panel Session

Roger Hausmann introduced the panel session with questions of the audience on their attitudes and preparedness for "fail-safety". He pointed out that the military presenters are very concerned about safety and robustness of their systems.

After some discussion about the consequences of various types of IDSS failure, Paul Compton pointed out that while there is no such thing as a fail-safe system, there are increasingly robust techniques for building in something to warn that the current situation is looking dangerous, eg, the set of states are within the valid ranges but are the most extreme yet seen. Vic Sobolewski added another dimension to this by observing that there are many complex applications in Defence requiring safety critical software which must be formally verified before it gains acceptance by the users.

Frada Burstein used some examples to illustrate that what we call Decision Support Systems are not providing real decisions, just support. This point seemed to reappear

throughout the session in various guises, and at the end of the session Paul Compton used the example of the chess programs always being outsmarted by the grand masters by sheer deviousness one way or another, illustrating that there will always be situations where we are performing at a higher level than computers, thus needing decision support rather than unsupervised automation.

There were many occasions when the discussion came back to the fact that there should be more emphasis on the Support rather than on the Decision, since we can derive genuine support in a well designed system, but are not always able to trust automated decisions. For example, Frada pointed out that while a lot of the expert knowledge may be available from the system, for improved decision performance there would need to be some machine learning capability to recognise the cases where situations are not appropriate for the system to offer a decision.

The general issue of Human Factors drew a lot of comment about various types of experience where the performance of the system suffered unnecessarily because of neglect of this aspect, despite being an intuitively obvious way to gain productivity with relatively little effort. Gil Tidhar listed many examples from his experience where there was a lot of effort required to ensure software matches user needs, and there were many other testimoniees to this effect. The discussion on the Human Factors issue during the Panel Session would have been more involved had it not been discussed at great length (and heat) during the speakers question periods throughout the day.

The Panel Session discussions were very lively, with surprising absence of dissenting views, with the trend being a reasonably harmonious confirmation of the opinions above. By 16:50 the general topic of IDSSs had been very thoroughly beaten into submission, and airline schedules were starting to prioritise their way into the commentary. At this point the workshop was closed.

3. The Program

0900 - 0910	Opening Address: Dr Roger Creaser, Chief of Maritime Operations Division
0910 - 0950	"Situation Awareness - One Operator's Perspective", CMDR Chris Donald, RAN
0950 - 1030	"Enhancing Mission Success through Applications of Intelligent Decision Support", WGCDR Rick Owen, RAAF
1030 - 1045	Morning Tea
1045 - 1110	"Incremental Development of Decision Support Systems", Paul Compton, UNSW
1110 - 1135	"The Australia/US collaboration on tactical decision research: NRaD, the RAN and DSTO", Brad Noakes, CSSG, ITD, DSTO
1135 - 1200	"Methodologies for Building IDSSs", Andrew Blair, UTS
1200 - 1225	"Modelling Teams and Team Tactics in Whole Air Mission Modelling", Gil Tidhar, Aust. AI Inst., Clinton Heinze & Mario Selvestrel, AOD, DSTO
1225 - 1345	Lunch
1345 - 1400	"A Knowledge-Based Strategy for the Re-association of Fragmented Sensor Reports", Keith Mason, EWD, DSTO
1400 - 1415	"An Intelligent Radar Tasking System", Tristan Chiu, BHP IT, Doug Kewley & Chris Crouch, HFRD, DSTO, Laurie Lock Lee, BHP IT
1415 - 1440	"Knowledge Modelling for Intelligent Decision Support", Frada Burstein & Helen Smith, Monash
1440 - 1505	"Software Tools for the AP-3C Operator", Phil Silver, AOD, DSTO
1505 - 1530	"Brief Analysis of Multi-Sited Multi-Sensor Inshore Surveilllance System for US Navy", Ian Croser, CEA Technologies
1530 - 1550	Afternoon Tea
1550 - 1625	Panel discussions introduced by Roger Hausmann, Geneva Computing
1625 - 1645	Washup, Simon Goss, AOD, DSTO

4. The Papers

Situation Awareness - One Operator's Perspective

CMDR Chris Donald, RAN

ASSTASS PD Campbell Park
CP 3-4-13 Canberra
06 266 2370

SITUATION AWARENESS

ONE OPERATOR'S PERSPECTIVE

MY NAME IS CHRIS DONALD, DAFFY TO MY FRIENDS AND SOMETHING WORSE TO MY ENEMIES. THE AIM OF THIS PRESENTATION IS TO OPEN AT BEST SOME DISCUSSION POINTS AND AT WORSE SOME FESTERING SORES ON THE SUBJECT OF INTEGRATION AND AUTOMATED SUPPORT IN A MULTISENSOR ENVIRONMENT. TO EXPLAIN WHERE I'M COMING FROM I SUPPOSE I SHOULD TELL YOU WHERE I CAME FROM. I SPENT MY BABY YEARS AS A GRUNT LUGGING AN M-60 AROUND THE J. IN THAT ENVIRONMENT I HAD FOUR BASIC SENSORS -SIGHT, HEARING, SMELL AND GUTFEEL - VERY PERSONAL SENSORS BUT AS ANY ARMY BLOKE WILL TELL YOU LAND WARFARE CAN BE A VERY PERSONAL THING. GETTING SICK OF LUGGING MY HOUSE AROUND ON MY BACK I JOINED THE NAVY AND BECAME AN OBSERVER - SPENDING 8 FUN FILLED YEARS IN S2 TRACKERS AS A SENSOR OPERATOR, THEN TWO YEARS WITH RAAF AS A P3B NAV/SENSO AND P3C SENSO. FOLLOWING THAT I WORKED THE ACOUSTIC INTELLIGENCE PATCH IN AJAAC - HAVING THE FUN OF RIDING THE ODD O-BOAT. SINCE 1987 I HAVE BEEN IN THE SPONGE WORKING FOR THE SUBMARINERS IN DSMPW AND IN THE NEW SUBMARINE PROJECT, LATELY, I HAVE THE FUN OF BEING THE DIRECTOR OF ACOUSTIC SURVEILLANCE SYSTEMS AND TACTICAL ARRAY SONAR SYSTEMS - A JOB THAT PRETTYWELL COVERS THE WHOLE SPECTRUM - BOTTOM MOUNTED ARRAYS THROUGH TO SONOBUOYS. I CONSIDER MYSELF VERY LUCKY TO HAVE BEEN EXPOSED TO SUCH A WIDE RANGE OF PLATFORM TYPES (MY BOOTS TO THE COLLINS) -BUT SOMETIMES ONE'S LUCK RUNS OUT - THUS I WAS FOUND BY GREG GIBBON AND COERCED INTO STANDING HERE - NEVER BUY A USED CAR FROM GREG!

FIRSTLY, I MAKE NO APOLOGIES FOR CLASSING MYSELF AS A HUNTER/KILLER. MY JOB IS TO CONTRIBUTE TO WHATEVER THE NATIONAL AIM MAY BE AND I SUSPECT THE BEST WAY TO DO THAT IS BY WINNING ENCOUNTERS. I REMEMBER WHEN I WAS KID, GOING IN NEXT DOOR TO WATCH THE RICH PEOPLE'S TELEVISION THERE WAS AN ADVERTISEMENT FOR POWER COACHING COLLEGE THAT WAS THE LATE 50'S EARLY SIXTIES VERSION OF TIM SHAW AND DEMTEL - THE SLOGAN WAS 'KNOWLEDGE IS POWER' - IN MY EXPERIENCE HOW TRUE THAT STATEMENT IS!

I AM GOING TO ORIENT MY DISCUSSION TO PLATFORM BASED SENSOR SYSTEMS AND
STAY WELL AWAY FROM CONFUSION CUBED & INDECISION (C3I). THIS APPROACH IS
BASED ON TWO PREMISES:

MY BACKGROUND IS IN PLATFORM SENSORS, AND

AS GENERAL PATTON ALMOST SAID - THE GENERALS CAN MAKE ALL THE DECISIONS
THEY LIKE, IT'S STILL THE INFANTRYMAN WITH HIS BAYONET THAT ACTUALLY WINS
THE ENGAGEMENT.

THE CALL FOR PARTICIPATION MENTIONED THE TERM - HIGHLY STRESSED HUMAN
OPERATOR. AS A PLATFORM MAN THESE ARE THE NEGATIVE STRESSORS (NOT IN ANY
PARTICULAR ORDER AS THEY VARY FROM PLATFORM TO PLATFORM AND THE SITUATION)
I SEE ACTING ON US WHEN WE ARE TRYING TO FIGHT OUR SYSTEMS AND THESE ARE
THE ONES I THINK WE SHOULD ATTACK:

THINGS OPERATOR'S BITCH ABOUT

* BOREDOM/MONOTONY
* COMMAND PRESSURE
* OPERATOR OVERLOAD
* THREAT OF ENEMY ACTION
* PLATFORM MOTION
* HEAT
* NOISE
* LIGHTING
* COLD
* VIBRATION

* FATIGUE/TIREDNESS
* DISPLAYS/CONTROLS
* WORKSTATION DESIGN
* RISKY PEACETIME OPERATIONS
* MOTION SICKNESS
* ILLNESS
* AIR CONTAMINATION
* NIGHT WATCHKEEPING
* AIR PRESSURE

VUGRAPH	TALKING POINTS

VUGRAPH   **TALKING POINTS**

<u>BOREDOM/MONOTONY</u>

IN MY OPINION THIS NEGATIVE STRESSOR IS A DIRECT
RESULT OF THE WAY IN WHICH SIGNAL PROCESSING HAS
BEEN DEVELOPED OVER THE YEARS. WE THE OPERATORS
ARE SUPPLIED WITH A BOX FULL OF WONDERFULLY CLEVER
ALGORITHMS, EACH WITH THEIR OWN PURPOSE AND, MORE
OFTEN THAN NOT, LITTLE SINS, BOTH OF WHICH THE
OPERATORS KNOW LITTLE ABOUT. BECAUSE OF THIS SENSOR
OPERATION IS A VERY PASSIVE HUMAN ACTIVITY IN THE
MAIN - WAITING FOR SOMETHING TO HAPPEN AND THEN
GOING INTO A TIZZ. THE SENSORS ARE NOT TASK DRIVEN
- FOR EXAMPLE WE ARE GIVEN TRACKERS WITH SOME FUNNY
NAME LIKE MARKOV MODEL ONE AND ALPHA-BETA - WHAT
THE HELL DOES THAT MEAN - WHY NOT CALL THEM LONG
RANGE TRACKER AND SHORT RANGE TRACKER. BECAUSE THE
OPERATORS DO NOT KNOW WHAT THE FEATURES ARE FOR
THEY DO NOT DRIVE THEIR SENSORS - BUT SELECT A HIT
AND HOPE OPTION AND WAIT FOR IT ALL TO HAPPEN.
HAVING NOTHING TO DO (OR THE DOING REQUIRES AN
UNDERSTANDING THAT IS TOO HARD) THE OPERATORS SIT
BACK AND GET BORED.

<u>FATIGUE/TIREDNESS</u> MUCH THE SAME STORY AS FOR BOREDOM

<u>COMMAND PRESSURE</u> THIS IS WORTH A PAPER BY ITSELF. IN ESSENCE THE
OPERATORS DO NOT REALLY UNDERSTAND WHAT COMMAND
WANTS AND THE COMMAND REALLY DOESN'T UNDERSTAND
WHAT THE OPERATORS CAN GIVE THEM - CLASSIC FIELD
FOR POTENTIAL CONFLICT - ONE IN WHICH THE OPERATOR
WILL ALWAYS LOSE CAUSE THE COMMAND HAS MORE
STRIPES. SORTING OUT THE INFORMATION CHAIN IS
CRITICAL TO GETTING EVERYTHING ELSE RIGHT.

<u>DISPLAYS/CONTROLS</u> MUCH THE SAME STORY AS FOR BOREDOM AND COMMAND PRESSURE
RE INFO CHAINS

<u>OPERATOR OVERLOAD</u> A CONSEQUENCE OF COMMAND PRESSURE IN MOST INSTANCES

<u>WORKSTATION DESIGN</u>   MUCH THE SAME STORY AS FOR BOREDOM

<u>THREAT OF ENEMY
ACTION</u>

THE SIN OF THE PEACETIME DEVELOPERS - WE JUSTIFY
SYSTEMS ON THEIR SURVEILLANCE CAPABILITIES ETC
WHICH IS A NICE BENIGN ENVIRONMENT AND WORSE WE
DEVELOP OPERATIONAL PROCEDURES AND CONSEQUENT OMI
ETC ON 'EXPERIENCE' IN THE EXERCISE AREAS - WHEN

THE BALLOON GOES UP WE FIND WE'VE GOT IT WRONG AND HAVE TO DEVELOP EUPHEMISMS FOR FRIENDLY FIRE ETC. THE OPERATORS KNOW THIS AND WHEN THE BALLOON GOES UP AS IN THE GULF DEPLOYMENTS THERE IS A FRENZY TO TEST AND MODIFY PROCEDURES AND THROW 'FUNNIES' ONTO THE PLATFORMS TO GET AROUND THE PEACETIME SYSTEMS' SHORTFALLS.

RISKY PEACETIME
OPS   MUCH THE SAME AS FOR THREAT OF ENEMY ACTION

PLATFORM MOTION   NOTHING MORE DISTRACTING THAN TRYING TO DO SOMETHING WHILE PULLING 'G'. ALSO TRYING TO KEEP ABREAST OF THE ANGLES AS THE PROTAGONISTS DO IMMELMAN TURNS. EVER TRIED TO OPERATE A SENSOR IN SEVERE TURBULENCE? LOTS OF STRESSORS INDUCED BY MOTION.

MOTION SICKNESS   AN OBVIOUS CONSEQUENCE OF PLATFORM MOTION - BUT THE OPERATOR STILL HAS TO DO THE JOB - THE SMALLER THE TAC TEAM THE MORE CRITICAL IT IS THAT HE/SHE STILL CONTRIBUTES - I HAVE BEEN IN AN S2 WHERE BOTH THE TACCO AND THE JULIE OPERATOR WERE SERENADING THE PILOT AND MYSELF BUT STILL GETTING THE JOB DONE -28 SONOBUOYS AND 84 SUS DROPPED IN LESS THAN 1 HOUR TO ANNOY THE LOCAL O-BOAT.

HEAT   AN OBVIOUS STRESSOR WHICH CONTRIBUTES TO OPERATOR OVERLOAD, TIREDNESS ETC. A LESS OBVIOUS EFFECT IS THE 'PISSED-OFFEDNESS' IT CAN INDUCE WHEN SWEATY OPERATORS LEAN OVER PAPER TRACES - WET THEM AND THE MACHINE JAMS. NOTHING LIKE TRYING TO ANNOTATE WET PAPER AS WELL.

ILLNESS   SIMILAR TO MOTION SICKNESS. THERE IS A HIDDEN IMPACT IN BOAT CREWS WHERE THINGS SUCH AS COLDS AND INFLUENZA WHERE THE WHOLE CIC GOES DOWN BUT STILL MUST FIGHT.

<u>NOISE</u>	APART FROM THE OBVIOUS OF INDUCING FATIGUE, NOISE CAN BE A GOOD DISTRACTER BUT IN SUBMARINES AND SHIPS IT IS A FACT OF LIFE WHERE DIFFERENT TASKS ARE ALL GOING ON IN THE ONE SPOT. NOISE CAN ALSO BE A POSITIVE AS IT CAN KEEP PEOPLE IN THE PICTURE THAT SOMETHING IS GOING ON.
<u>AIR CONTAMINATION</u>	THIS CAN BE MORE IMPORTANT THAN MAY BE OBVIOUS. CO_2 LEVELS SLOWING THE THOUGHTS DOWN ETC, NOTHING LIKE DISTRACTING AN AIRBORNE OPERATOR THAN A FUEL SMELL.
<u>LIGHTING</u>	THE MOST OBVIOUS IS GETTING SCREEN AND DECISION AIDS LINED UP WITH THE PLATFORM LIGHTING. A OLDIE IS BLACK LIGHTING IN BOATS - COMMAND PRESSURE COMES INTO PLAY ON THIS.
<u>NIGHT WATCHKEEPING</u>	THE WHAT THE HELL AM I DOING THIS LOT FOR SYNDROME - WITH BIORHYTHMS ETC.
<u>COLD</u>	OPPOSITE TO HEAT - PROBABLY CONTRIBUTES MORE TO FATIGUE THAN HEAT.
<u>AIR PRESSURE</u>	EVER BEEN IN A SNORTING BOAT WATCHING THE BAROMETER PLUNGE ON A GULP OR AN ATTEMPTED FLAMEOUT? VERY GOOD TRAIN OF THOUGHT BREAKER -CONTRIBUTES TO NOISE WHEN THE CAPTAIN SPITS THE DUMMY.
<u>VIBRATION</u>	CONTRIBUTES TO FATIGUE. IN MANY CASES AN ATTENTION DIVERTER AS VARIATIONS TO THE LEVEL OF AMBIENT VIBRATION ARE TAKEN AS AN INDICATION SOMETHING IS AMISS. ASK A P3 OPERATOR HOW INTERESTED HE IS IN SENSOR OPERATIONS WHEN NO 1 DONK STARTS RATTLING THE AIRCRAFT ON START-UP AFTER A PATROL LOITER SHUTDOWN.

ALL OF THESE THINGS CAN ADVERSELY AFFECT:

* VIGILANCE

* VISUAL INFORMATION PROCESSING

* AUDITORY INFORMATION PROCESSING

* REASONING/DECISIONMAKING

* PERCEPTIONS OF WHAT IS GOING ON

* CONFIDENCE/MOTIVATION

THE MUSHROOM THEORY

AS IS INDICATED IN THE CALL FOR PARTICIPATION SIGNAL PROCESSING IS A MATURE
AREA. FROM AN OPERATOR'S PERSPECTIVE THE MORE MATURE THE SIGNAL PROCESSING
THE LESS OPERATIONALLY EFFECTIVE IS THE OPERATOR VERSUS THE MODELLING
PREDICTIONS. IN SIMPLE TERMS WHEN I WAS A BOY THE SENSORS HAD AN ON-OFF
SWITCH AND TO MAKE THEM SING I HAD TO WORK OUT WHAT MY ENEMY'S POTENTIAL
COURSES OF ACTION COULD BE AND THROUGH TACTICAL CUNNING CONTRIVE A PLAN TO
WIN. BUT AS THE SIGNAL PROCESSING INCREASED SO DID THE COMPLEXITY OF THE
BOX I HAD TO DRIVE, AND FAR MORE DISTANT DID IT BECOME FROM BEING A WEAPON
OF WAR TO A SCIENTIFIC CURIOSITY. NOT ONLY DID WE HAVE TO TURN IT ON BUT
WE HAD TO LEARN ALL THE RULES ABOUT WHAT IT COULDN'T DO AND WHAT WOULD
HAPPEN TO US IF WE DIDN'T - INEVITABLY YOU'VE BEEN A VERY BAD BOY AND
YOU'LL HAVE TO GO BACK AND START ME UP AGAIN -JUST WHAT I NEEDED WHEN I WAS
RUNNING IN FOR AN ATTACK OR AVOIDING THE BAD GUY ON MY TAIL. OF COURSE ONE
COULD BLAME THE COMBAT SYSTEM SAFETY CRITICAL SOFTWARE PEOPLE FOR THAT ONE
- BUT THAT DIDN'T BECOME A TERM UNTIL THE LATE 80'S.

MORE IMPORTANT WAS THE FACT THAT EVERY MENTAL RESOURCE THAT WAS DIVERTED TO
ENSURING THE PROCESSOR'S WELLBEING WAS BEING DIVERTED FROM THE OPERATIONAL
TASK AT HAND. IN EFFECT TRAINING TIME WAS DIVERTED FROM ENSURING OPERATORS
HAD GOOD KNOWLEDGE OF THE ENEMY TO MAKING SURE WE DID NOT BREAK ONE OF THE
ENGINEER'S PRECIOUS PROCESSORS.

DO NOT BE DELUDED THAT THIS HAS GONE AWAY - EVEN WITH TODAY'S WONDERFUL
WINDOWS ENVIRONMENT IT IS STILL A PIECE OF PIE TO LOSE THE WINDOW YOU WANT
WHEN THE MAIL IS INCOMING.

IN TERMS OF UNDERSTANDING THE ENEMY WE TRULY HAVE BECOME MUSHROOMS - WE ARE
IN THE DARK ON WHAT HIS NATURE IS AND THE SCIENTIFIC COMMUNITY BELIEVE WE
CAN BE SATIATED BY BEING FED ON GIGABIT.

NOT HAVING THE TIME TO LEARN ABOUT THE ENEMY TOTALLY UNDERMINES OUR
CONFIDENCE AND MAY BE THE BIGGEST STRESSOR OF THEM ALL.

WHEN I TALK ABOUT KNOWING THE ENEMY IT IS NOT SO MUCH THE TRADITIONAL HIGH
LEVEL STUFF ONE GETS FROM DIO, RATHER IT IS MORE GENERIC - FIGHTERS
IRRESPECTIVE OF PARTICULAR TACTICS TEND TO CARVE UP THE AIRWAVES RATHER
DIFFERENTLY THAN MARITIME PATROL AIRCRAFT, BALLISTIC MISSILE FIRING
SUBMARINES TEND TO BEHAVE DIFFERENTLY THAN ATTACK BOATS, TORPEDO FIRING
SUBMARINES BEHAVE DIFFERENTLY TO MISSILE FIRING BOATS. IT IS UNDERSTANDING
THESE BEHAVIOURS WHICH IS INTEGRAL TO OUR CONFIDENCE IN A FIGHT - KNOWING
THE TACTICS IS A BONUS.

BUT OUR SENSORS HAVE FORCED PEOPLE INTO SEEING TARGETS AS SQUIGGLES ON A
SCREEN AND THE UNDERLYING UNDERSTANDING OF WHAT IS CAUSING THE SQUIGGLES TO
LOOK THE WAY THEY DO HAVE BEEN ERODED OVER THE YEARS - THE MORE
TECHNOLOGICALLY WONDERFUL THE SIGNAL PROCESSING THE QUICKER THE EROSION
PROCESS.

IN MULTI-SENSOR FUSION THE OPERATORS ARE TRYING TO GROUP THE SQUIGGLES FROM
THE DIFFERING SENSORS BUT BECAUSE THEY CANNOT IMAGINE THE NATURE OF THE
BODY EMITTING THE DETECTION STIMULI THEY REALLY HAVE A PROBLEM PULLING THE
PICTURE TOGETHER.

OUR OWN LACK OF TACTICAL DEVELOPMENT OVER THE YEARS STANDS AS MUTE
TESTIMONY THAT SOMETHING IS DIVERTING OUR THOUGHT TRAINS.

AS NSA SAID WHEN I WAS DRAINING DOWN ON HIM IN EARLY JUNE - THESE PEOPLE
SOUND TECHNICALLY COMPETENT BUT THEY CAN'T GRASP THE BIGGER PICTURE'
TOO TRUE AND IT TOOK A PHD TO SAY IT FOR ME - THE BASTARD!!

SO WHAT DO I THINK WE SHOULD BE DOING:

THIS GAME IS DIFFERENT TO ANYTHING DSTO HAS EVER DONE BEFORE - GONE ARE THE
DAYS OF THE INNOVATIVE INDIGENOUS GIZMO WHERE SOME ADF SPONSOR ASKED FOR A
BETTER MOUSETRAP AND THEN WHEN TO HIS NEXT POSTING WITHOUT EVER REALISING
DSTO WERE BUILDING THE BEST GODDAMN BEAVER TRAP EVER IMAGINED. THIS IS A
GAME WHERE THE FIRE MUST BE HELD TO THE SPONSOR - WE ARE TALKING
INFORMATION MANAGEMENT AND THE ONLY PEOPLE THAT CAN FILL YOU IN ON
INFORMATION REQUIREMENTS ARE THE OPERATORS. AND I CAN ASSURE YOU THIS WILL
BE AN ARM WRESTLE - FIRST YOU HAVE TO OVERCOME THE CULTURAL CLASH. ON ONE
SIDE THERE IS US THE OPERATORS -THE TOE CRUSHERS. ON THE OTHER THE

SCIENTISTS - THE COSINE CRUSHERS. AND HANGING AROUND WATCHING US BOTH WITH
TREPIDATION ARE THE ENGINEERS.

FIRST CULTURE CLASH:
IN MY EXPERIENCE THE SCIENTISTS THINK THE OPERATORS KNOW EVERYTHING ABOUT
OPERATIONS AND THE OPERATORS THINK THE SCIENTISTS KNOW EVERYTHING ABOUT
SCIENCE - AND NOTHING IS FURTHER FROM THE TRUTH.

SECOND CULTURE CLASH:
OPERATORS LIVE IN A BRUTAL ENVIRONMENT, THEY OFTEN SEE WHO THEY KILL AND
THEY SEE THEIR OWN CASUALTIES. THIS DRIVES THEIR OUTLOOK. THIS MAY NOT BE
THE CASE WITH THE AVERAGE SCIENTIST.

THIRD CULTURE CLASH:
FOR SCIENTISTS THE HOW SOMETHING DONE MAY BE THEIR VERY RAISON D'ETRE.
OPERATORS WORRY ABOUT WHAT IT DOES AND COULDN'T CARE LESS ABOUT HOW ELEGANT
A SOLUTION MAY BE.

AND SO ON.......

ONCE YOU JUMP THIS HURDLE AND YOU CAN GET THE OPERATORS TALKING IT IS TIME
TO LIVE BY SOME RULES (PINCHED FROM A US OPERATOR/PHD (DR BRADFORD A.
BECKEN):

1.   NEVER BASE A SYSTEM DESIGN UPON SOME HYPOTHETICAL SCENARIO AS
        TO HOW THE SYSTEM WILL BE EMPLOYED WHEN IN FLEET USE.

2.   MAKE CERTAIN THAT A SYSTEM DESIGN IS NOT TUNED TO ONE
        PARTICULAR OPERATING ENVIRONMENT.

3.   WITHOUT PROTOTYPES TO EVALUATE, NO MATTER HOW IMPERFECT, THE
        OPERATING FORCES HAVE DIFFICULTY IN DEFINING THEIR OPERATIONAL
        REQUIREMENTS.

4.   WHEN INTRODUCING A NEW CAPABILITY TO THE FLEET, KEEP IT SIMPLE.
        IT IS BETTER TO SOLVE A PROBLEM IN SMALL, SEQUENTIAL STEPS
        RATHER THAN IN A SINGLE GIANT LEAP.

5.   A SUCCESSFUL PROGRAM REQUIRES A CLEARLY DEFINED PROGRAM SPONSOR
        WITH A BROADLY RECOGNISED NEED AND REASONABLE CONTINUITY IN
        PROJECT MANAGEMENT.

HIS OPINION MIRRORS MINE AFTER HAVING THE TRUE FUN OF BUILDING A NUMBER OF
SYSTEMS FOR THE RAN SUBMARINE COMMUNITY - AND WE ACCIDENTLY DID EVERYTHING
HE SUGGESTS.

SO WHERE ARE YOUR BIG DANGERS;
(vugraph)

POINT 2 THE TIMEFRAME IN WHICH IT TAKES TO PRODUCE PROTOTYPES (ESPECIALLY
SINCE DSTO ISN'T ALLOWED TO DO THAT ANYMORE) WE HAVE TO GO TO INDUSTRY AND
THIS TAKES TIME AND CUTS ACROSS THE POSTING CYCLES OF YOUR ORIGINAL
PALADIN(S).

POINT 3 THE ADF'S UNWILLINGNESS TO FREE UP OPERATORS FOR FULLTIME SERVICE
IN THE DSTO WHERE WE COULD REALLY WHIP SOME OPERATIONAL UNDERSTANDING INTO
THE SCIENTISTS
ASSUMING YOU'VE FOUND A PALADIN WILLING TO TAKE THE BALL UP THROUGH THE
FORWARDS WHERE CAN WE MAKE GAINS?

**OPERATORS WHO ARE WILLING/AVAILABLE TO DO THE HARD YARDS IN THE INITIAL
SYSTEM DEFINITION PROCESS. A FEW REASONS:**

* EROSION OF USEFUL KNOWLEDGE,

* THE TIMEFRAME IN WHICH IT TAKES TO PRODUCE PROTOTYPES,

* THE ADF'S UNWILLINGNESS TO FREE UP OPERATORS

Enhancing Mission Success through Applications of Intelligent Decision Support

WGCDR Rick Owen, RAAF

DRR-AF
Russell Offices, C-4-49

Abstract

Outside, it's pitch black, no moon and a heavy overcast sky has completely obliterated the meagre night illumination. Hmm, so much for the NVGs. It is not the sort of night you would like to be out driving you car, but here you are at 60 metres above the ground travelling at close to 1,000 kph. You're thinking to yourself, 'the most intelligent decision I could have made was to stay at home', but it's a job.

Flying a night strike mission in a RAAF F-111C. Intelligent Decision systems have a part in enhancing the chances of mission success. This technology can be applied to all phases of the strike mission, starting at mission planning and finishing with improvements in mission analysis and crew debriefing procedures.

The following phases/tasks are typical in a strike mission:

    a.   Mission Planning
    b.   Navigation
    c.   Self Protection
    d.   Target Acquisition and Tracking
    e.   Weapon Delivery, Tactic and Recovery
    f.   Mission Analysis

Mission Planning and Mission Analysis make use of off-board systems that feed or are fed from on-board data and traditionally operate in isolation from the combat environment. Outputs and inputs are not necessarily real-time, nor do they require real-time solutions. Navigation systems are generally mature technology and include Inertial Navigation Systems, Terrain Referenced Navigation and Global Positioning Systems. The application of these systems to solve velocity, position and attitude resolution is well established. The use of navigational data as part of an information fusion system to control electronic emissions, for example, will require further investigation.

The presentation will examine three critical phases of a typical strike mission with a view to the application of technology solutions to improve situational awareness, tactical decision making and hence mission performance. These phases are:

    a.   Self Protection
    b.   Target Acquisition and Tracking
    c.   Weapon Delivery, Tactic and Recovery.

INTELLIGENT DECISION SUPPORT SYSTEMS
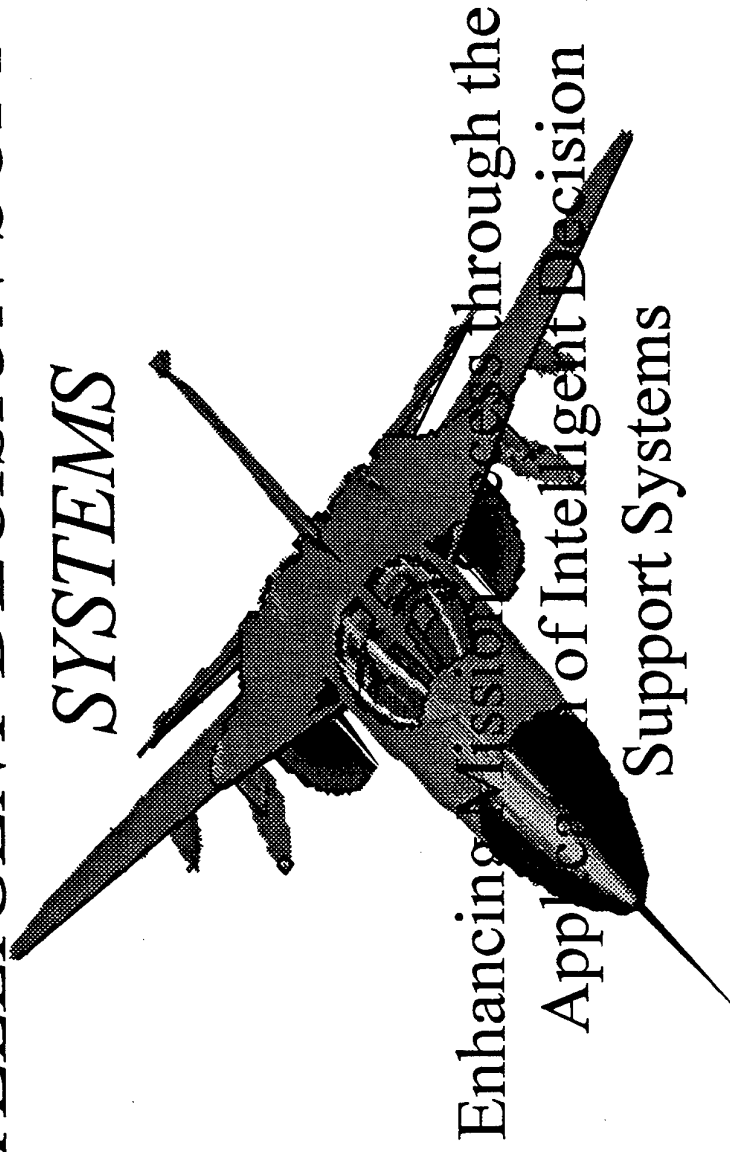
Enhancing Mission Success through the Application of Intelligent Decision Support Systems

Presentation by:
WGCDR Rick Owen

Mission Planning & Analysis

- Systems are off-board.
  * Not Real-time solutions.
- Data recorded on-board.
- Mission Planning for optimum route.
  * Threats, Terrain, Target, Timing & Weapon.
- Mission Analysis for critical analysis of plan v actual mission.
- Feedback into mission planning System.

21

5
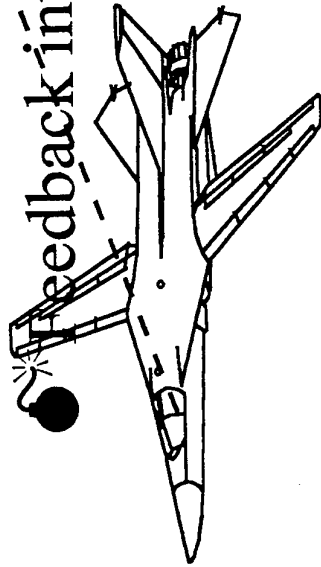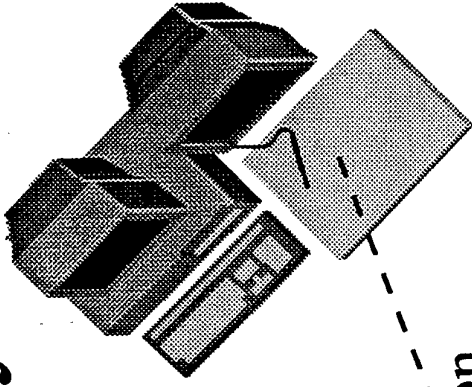
Navigation Systems

✈ Well established technology.

✈ Military Applications unique.

➤ Self-contained.

➤ Extreme Accuracy.

➤ Jam Resistant.

✈ Mixture of INS, TRN, GPS.

Mission Phases

Mission Planning

Enroute Navigation

Self-protection

Weapons Release,
Tactic & Recovery

Target Acquisition
& Tracking

Mission Analysis
& Debrief

AIM
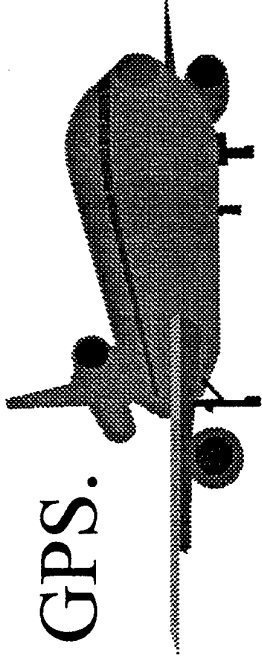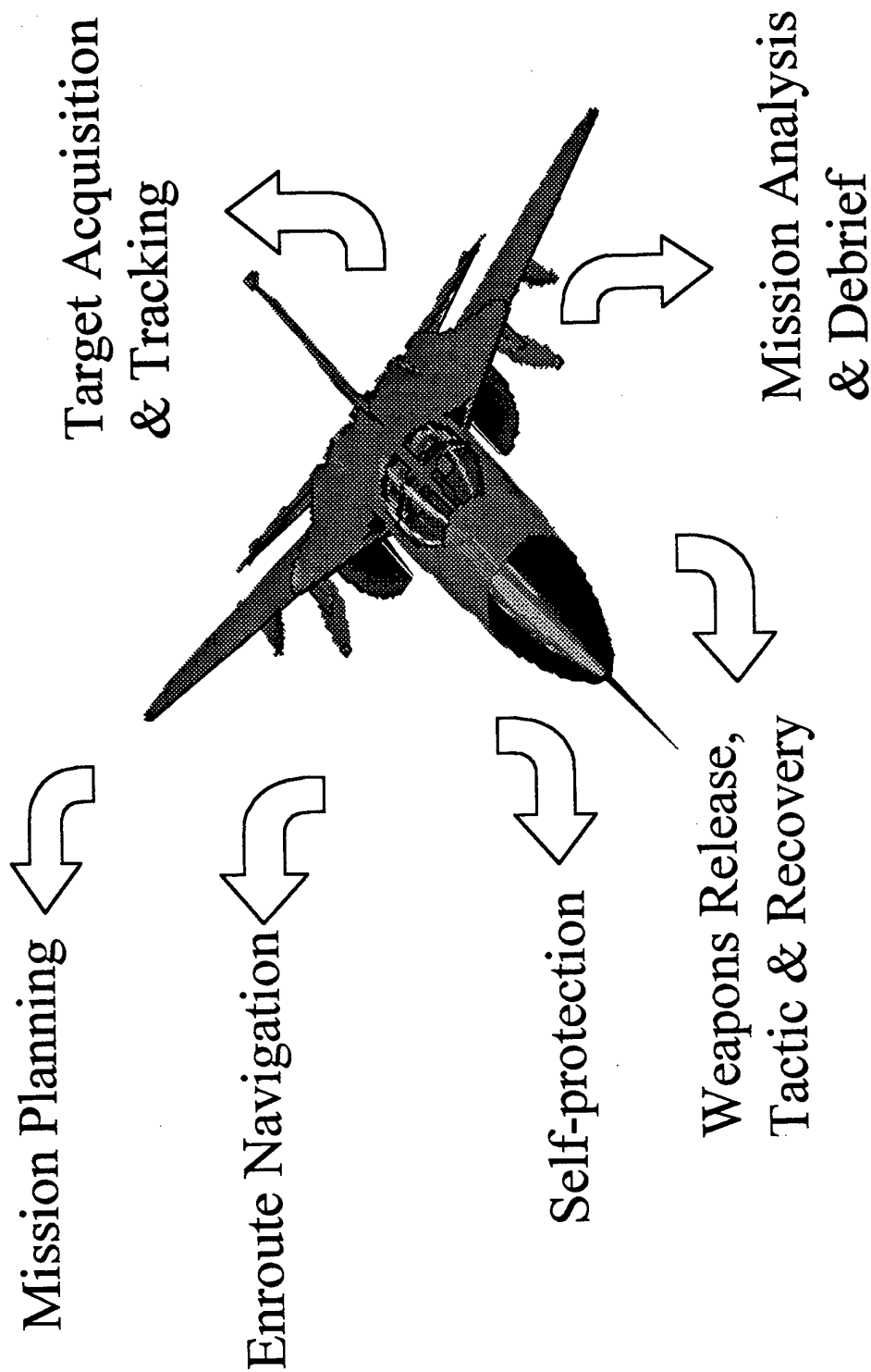
o provide an operator view of the areas intelligent decision systems could be employed to enhance mission effectiveness.

Secondary, validity of intelligent systems for real-time tactical decision making.

Mission Phases

Mission Planning

Enroute Navigation

Target Acquisition & Tracking

Self-protection

Weapons Release, Tactic & Recovery

Mission Analysis & Debrief

6

Multi-Sensors?

Infra-red Pave Tack
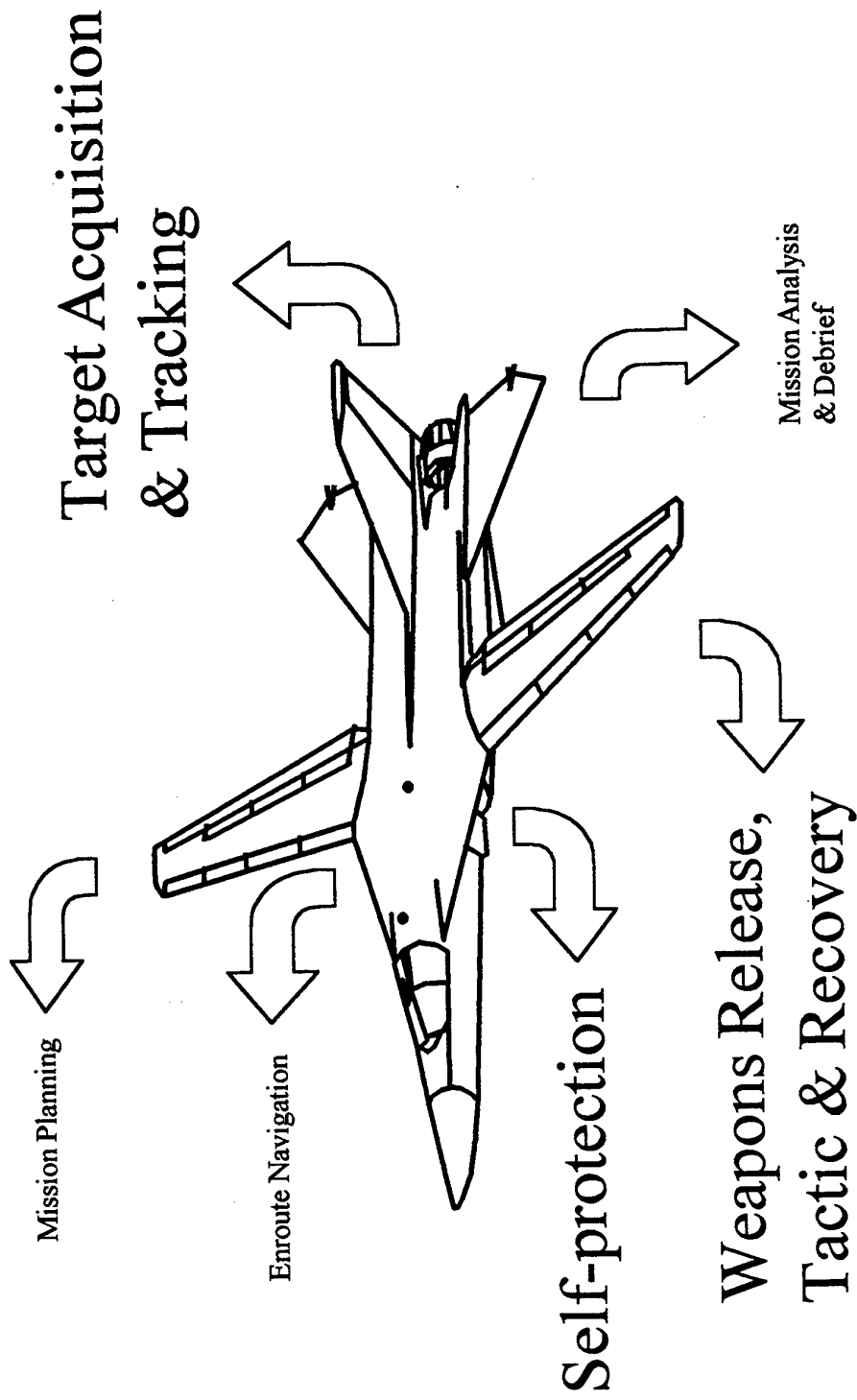
EW Threat Receivers

Navigation & Weapon Data

Radar

Self-Protection

- ☑ Prior Knowledge
- ☑ Threat Spectrum Coverage
- ☑ Timely Detection
- ☑ Correct Response
- ☑ Platform Detectability (Stealth)
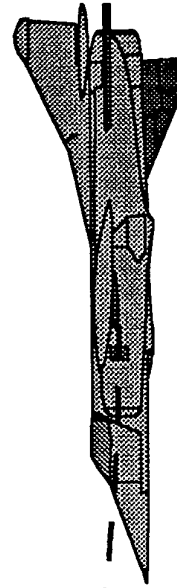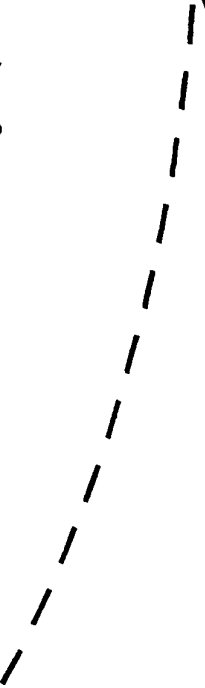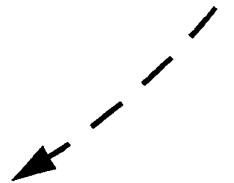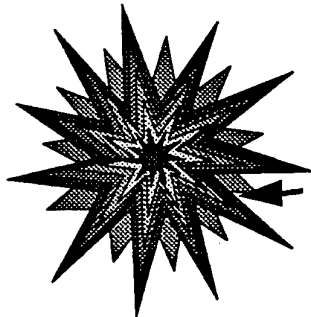
Self-Protection

- ☑ Prior Knowledge
- ☑ Threat Spectrum Coverage
- ☑ Timely Detection
- ☑ Correct Response
- ☑ Platform Detectability (Stealth)
- ☑ S**t Gunning
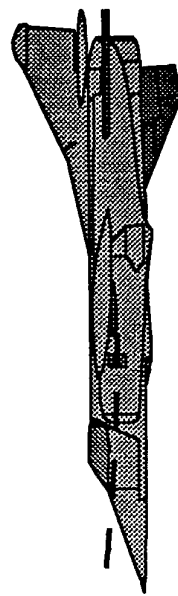
Self Protection (cont.)
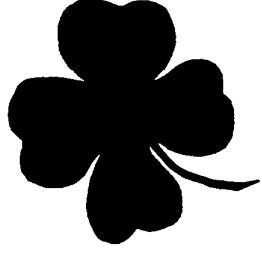
- Automatic vs Manual
- Reliability
- Operator Trust
- Tactical Response
- Display
  - ➔ Visual
  - ➔ Audio
  - ➔ Overlay

29
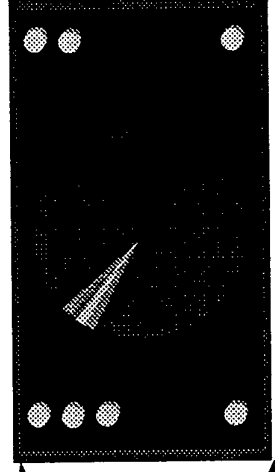
Target Acquisition & Tracking

ACQUISITION
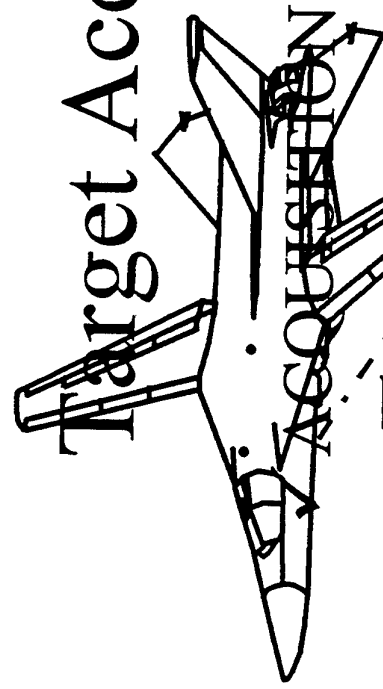
- ☑ Navigation Accuracy
- ☑ Sensor Fusion - Target Identification
- ☑ Display

✓ **TRACKING**

- ☑ Sensor Fusion - Aimpoint/s Identification
- ☑ Auto-Track, Multi Target Track
- ☑ Display
- ☑ Ergonomics
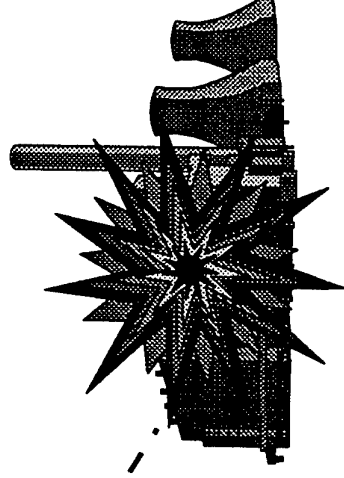
Weapon Delivery, Tactic and Recovery Slide 1 of 3

☑ Weapon Delivery

✓ PGM/Dumb

✓ Weapon Envelope Cue

✓ Release Conditions Confirmation

✓ Target Tracking/Aimpoint Selection

✓ Abort Criteria

31

Weapon Delivery, Tactic and Recovery Slide 2 of 3

☑ Tactic

☒ Freedom of Manoeuvre

  ✓ Maintain Target Tracking

  ✓ Threat Avoidance

  ✓ Ground Avoidance

☒ Safe/Optimum Flight Path Indication (HITS)

13

Weapon Delivery, Tactic and Recovery Slide 3 of 3

☑ Recovery

☒ Similar to previously discussed points

  ✓ Ground Avoidance

  ✓ Threat avoidance

  ✓ HITS

  ✓ BDA

  ✓ Multi-aircraft RV points

33

14

Intelligent Decision Systems

- Reliable/Accurate
- Timely
- Adaptable to Mission
  - Types (A to A, A to G)
  - Phases within a mission
- Priority Assessment
- Output Easy to Understand
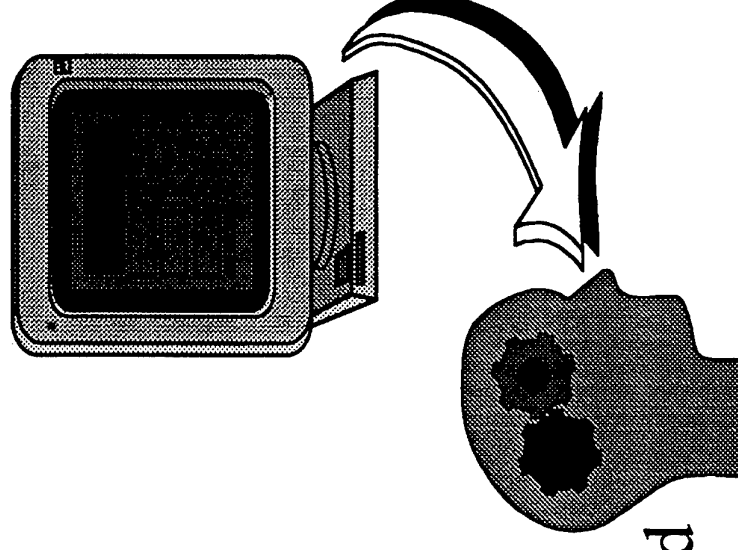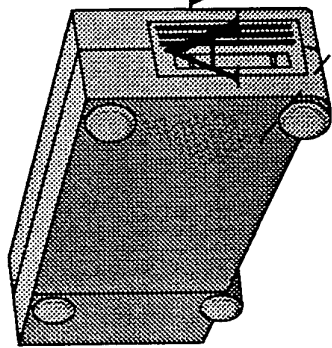
Avionics Implications
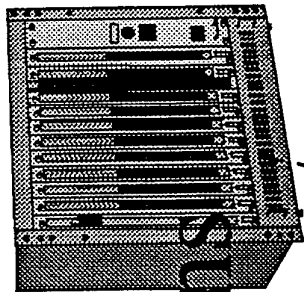
☒ Avionics Architecture

☒ Computer Processing Capability

☒ Display Systems

☒ On-board Storage Capacity

35

16
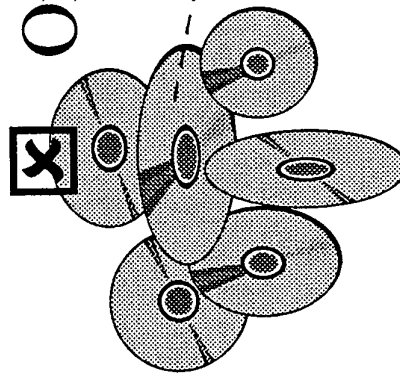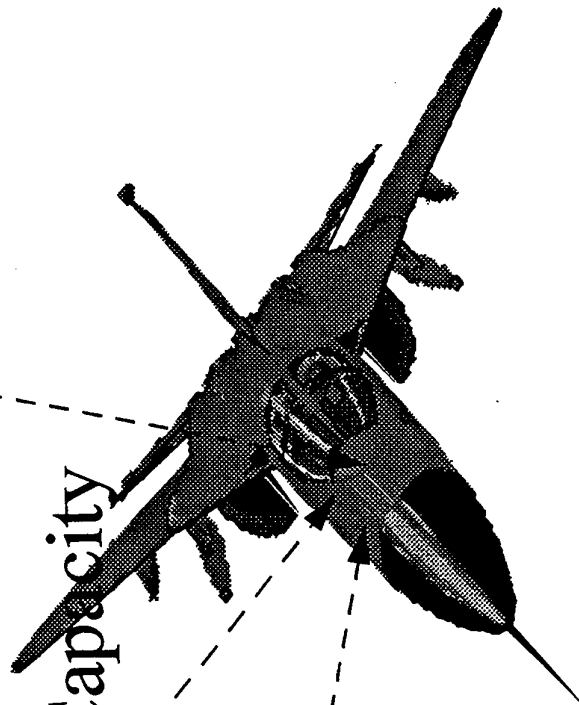
Airborne Decisions

☒ 90% of tactical decisions may be automated

☒ Remaining 10%, I'm not sure

Incremental Development of Decision Support Systems

Paul Compton

School of Computer Science and Engineering
University of New South Wales, Sydney 2052
compton@cse.unsw.edu.au

Abstract

A fundamental problem in developing and maintaining a knowledge based system (KBS) is that not only is it difficult to obtain and incorporate current expertise into the KBS, but the expertise in the domain may itself be evolving as people gain experience with new types of data from new sensors and new environments. An approach to developing knowledge bases (KB), (Ripple Down Rules (RDR)) has been developed which allows for such incremental development and modification. The essential feature of the approach is that when a human (or decision support system (DSS)) provides advice for a situation, and someone else (perhaps an expert with superior expertise) disagrees with the advice, the expert will highlight features in the situation which distinguish it from one where the advice may have been appropriate. If the advice provided by an RDR system is considered incorrect by the user, the RDR system presents the user with a list of differences between the present situation and one or more other situations where the advice would have been correct. The user selects the relevant features and indicates the appropriate advice and the system corrects its knowledge accordingly. The situation is itself stored to be used to provide for further lists of differences. This simple approach has been used to be build both real world and experimental systems, including the large medical system, PEIRS, and extended in various ways with the following results:

Large systems can be built without a knowledge engineer, purely from domain expertise.

Knowledge acquisition (KA) is very simple and the time for each acquisition is largely independent of the KB size. KA can usually take place within the normal workflow.

Despite the refinement based approach the resulting KB is relatively compact and the KA efficient.

Initial feature abstraction from data can be relatively simple as local refinement can readily improve weak abstractions.

The methodology assumes a user sufficiently competent to recognise bad advice. However, the history of the corrections made provides a way for the system to assess and report on its own reliability. In one experiment it correctly provided a warning for all situation where its advice was wrong reducing the checking requirement by 60%.

The system provides a strong case-based explanation for its advice. Extensions to the KA allow causal models matched to the KB to also be developed incrementally and provide an alternate source of explanation.

The following research areas are less developed:

For all the systems developed so far the data has been available on- line. The approach should be suitable for interactive domains but this has not been tested.

A machine learning method has been developed producing the same representation which produces very compact KBs. This needs to be integrated with the manual KA method.

It should be possible to allow for refinement of feature extraction knowledge without corrupting higher level knowledge about such features which is also being refined in the same system

Finally, the approach is being extended from classification to construction tasks.

The final and somewhat speculative goal of this work is an approach to building KBS whereby no prior domain or problems solving analysis is required, but one can start incrementally building a system by patching errors. The system then reflects on the evolving structure of its knowledge and re-organises itself to optimise its performance for the task it detects it is being trained for.

The Australia/US collaboration on tactical decision research: NRaD, the RAN and DSTO

Brad Noakes

DSTO C3 Research Centre
Fernhill, Canberra
Brad.Noakes@dsto.defence.gov.au

Abstract

The Australia/US collaboration on tactical decision research was conducted at the Decision Making Evaluation Facility for Tactical Teams (DEFTT) Laboratory at NRaD in San Diego in May 1995. A number of RAN officers were used as subjects in the DEFTT Laboratory, as part of the ongoing TActical Decision Making Under Stress (TADMUS) program to explore recent developments in decision theory and human computer interaction technology, and apply new decision making models to the design of a Decision Support System (DSS) for enhancing tactical decision making under highly complex conditions. The DEFTT Laboratory is a six station test bed environment that simulates the Anti Air Warfare (AAW) computer work stations of a shipboard combat information centre.

Eight RAN officers (ranging from LEUT to CMDR) were run through the TADMUS DEFTT as four two person teams, comprising a Commanding Officer (CO) and a Tactical Action Officer (TAO). The TADMUS program has completed its baseline studies using the AAW scenarios in the DEFTT. From the Baseline studies they had observed a number of problems in AAW using the current naval systems. In the current phase of TADMUS experiments a DSS that has been developed to aid the CO and TAO will be analysed to determine if it can decrease the incidence of problems that were observed in the baseline study. The RAN group were the first subjects to use the new DSS.

In addition to the collaboration with the TADMUS research program, the same eight RAN officers were also involved in the Collaborative Situation Assessment for C4I (CSA for C4I) project. This project evaluates the use of collaborative technologies in aiding Command and Control Warfare (C2W) teams. This experiment involved two four person C2W teams running through two scenarios constructed around a Non-combatant Evacuation Operation (NEO) at the battlegroup level. The four member team acted in the roles of the C2W Commander, the Intelligence officer, the Cryptologist and the Electronic Warfare officer.

Results from these collaborative studies are presented, in addition to recommendations for areas of further collaborative research. Decision support tools developed at NRaD were evaluated for their applicability for the RAN and other ADF Command Support System projects.

A Comparative Study of Formal Methodologies for Designing IDSSs

Andrew Blair, John Debenham, Jenny Edwards

Key Centre for Advanced Computing Studies,
School of Computing Sciences,
University of Technology, Sydney,
P.O. Box 123, Broadway, N.S.W. 2007, Australia
Tel: (02) 330-1842, Fax: (02) 330-1807
Email: andrew, debenham, jenny@socs.uts.edu.au

Abstract

Intelligent Decision Support Systems (IDSSs) are tools for helping decision making where uncertainty or incomplete information exists and where decisions involving risk must be made using human judgement and preferences. Over the past decade, IDSSs have been built for law enforcement, telecommunications, stock investment, manufacturing and transportation. Despite the large amount of research work in building IDSSs in recent years, little attention has been devoted to investigating the methodologies which are being used in designing IDSSs and understanding the extent to which these methodologies support the design of IDSSs.

In this paper, we compare five major formal methodologies for designing IDSSs. These five design methodologies combine and extend design techniques associated with Knowledge Based Systems and Decision Support Systems to provide support for constructing IDSSs. To compare these five design methodologies we use a framework which we have developed to gauge what each of these methodologies do. Our comparison shows that these methodologies are quite different in their approach for designing IDSSs and also that these methodologies differ greatly in the amount of support which they provide.

This comparative study is presented so as to assist IDSS developers in understanding what support can be gained from using each of the five major design methodologies and in choosing the correct one for their project. Furthermore, our study can be used by IDSS developers to compare the way that they work with the way proposed by each of the five major design methodologies.

Keywords: Intelligent Decision Support Systems, Knowledge Based Decision Support Systems, Decision Support Systems, Knowledge Based Systems.

1. INTRODUCTION

The major benefit that IDSSs have over traditional Decision Support Systems (DSSs) is that they assist decision makers to gain insight into the difficult decision at hand rather than just solving the problem or producing a somehow "correct" model of the decision [Holtzman, 1989]. IDSSs provide this support by combining and extending techniques associated with Knowledge Based Systems (KBSs) and DSSs. IDSSs are defined in detail by Gottinger et al. [1992], Holtzman [1989] and Buckner et al. [1991] and are known also as "Intelligent Decision Systems" [Holtzman, 1989], "Knowledge Based Decision Support Systems" [Dalal et al., 1992], "Expert Support Systems" [Van Weelderen, 1991] and "Expert-Based Systems" [Goul et al., 1987].

During the past decade, a large number of IDSSs have been implemented, and have illustrated that IDSSs significantly improve business processes within the organisation [Blair et al., 1995]. Despite the large number of IDSSs which have been built, few formal design methodologies have been published for the development of IDSSs when compared to the number of formal design methodologies for KBSs and DSSs; a methodology is a formal methodology if at any stage a new developer can take over and can continue in the same manner as the original developer, otherwise it is an informal methodology. We recently conducted an exploratory survey to identify what methodologies an International group of IDSS developers are using to build their IDSSs [Blair et al., 1995]. The exploratory survey identified that over half the 65 IDSSs surveyed were built using informal design methodologies and there is a need for formal design methodologies for IDSSs.

After an extensive search of international computer journals and conference proceedings in the past ten years, we could find only five major formal design methodologies for IDSSs; there are a number of methodologies which claim to combine DSS and KBS techniques but do not claim to support the development of IDSSs; see for example Saxena [1991]. Van Weelderen [1991, p. 25] attempted a similar search for methodologies in 1991 but reported that he could not find any contributions which focused on IDSS design. The five major formal design methodologies we identified are: 1) the Expert-Based Systems Methodology [Goul et al., 1987], 2) the "MEDESS" Expert Support System methodology [Van Weelderen, 1991], 3) the Visual Interactive Modeling Approach [Angehrn et al., 1990], 4) the Intelligent Decision System methodology [Holtzman, 1989] and 5) the Text Analysis Approach which supports the knowledge acquisition phase [McGovern et al., 1991].

In this paper we review and compare those five major formal design methodologies above. The paper begins (section 2) by outlining our framework which we use to compare the five major design methodologies. We then outline (section 3) the five major design methodologies and outline (section 4) the results of our comparison.

2. A FRAMEWORK FOR COMPARISON

Software design methodologies differ substantially both in the development phases that they address and in the way in which these phases are dealt with. To compare IDSS design methodologies, we describe a framework which we have developed to gauge what a given methodology does; thus we are able to compare the methodologies. In this section, we describe how the framework has been derived and then describe it.

To date, there have been a number of frameworks proposed for comparing KBS methodologies and DSSs methodologies, but none for comparing IDSS design methodologies. The framework which we employ has been constructed by combining the frameworks from Knowledge Acquisition [Fensel et al., 1994a], KBS Conceptual Modelling [Fensel et al., 1994b], Knowledge Engineering in its entirety [Clark, 1992], DSSs [Saxena et al., 1989] [Sol, 1990], and using the criteria for building IDSSs as described in Holtzman [1989]; note we also employ criteria for performing software reuse from [Biggerstaff et al., 1989] and criteria for user interface design from [Angehrn et al., 1990].

Our proposed framework has been constructed by determining which aspects of the above frameworks are relevant to the analysis of IDSS design methodologies. Our framework has been designed to accommodate the development phases typically associated with both

KBSs and DSSs [Saxena et al., 1989]; this then provides a benchmark for identifying the differences between the methodologies and the extent to which those methodologies support the development process. The development phases are: "Knowledge Acquisition / Requirements Analysis", "Conceptual Design", "External Design", "Internal Design", "Reuse Analysis" and "User Interface Design".

Our framework for comparing IDSS design methodologies consists of a sequence of "groups". Each group consists of a sequence of questions which pertain to one of the development phases of IDSSs mentioned above. Each question addresses a significant criterion in the design of effective IDSSs; these questions are sufficiently fine-grained to draw out the differences between the methodologies. The questions in each group are listed in Appendix A; note, we referenced where each question has been taken from the frameworks mentioned above. Below we define the development phase associated with each group in our framework.

• **Group KA: Knowledge Acquisition / Requirements Analysis.** In our framework, we have combined the Knowledge Acquisition (KA) and Requirements Analysis (RA) phases in line with the large amount of recent research work which has argued that there is a significant overlap between these two phases [Sharp, 1994] [Bryd et al., 1992]. In general, the *KA/RA phase* defines the problem which the IDSS will address and what the IDSS needs to do.

• **Group CD: Conceptual Design.** The *conceptual design* phase begins with the requirements specification from the KA/RA phase and constructs a complete model of the system called a "conceptual model". The *conceptual model* is a description of how the requirements will be achieved - independent of how the software system will be structured and implemented [Sol, 1990]; for a similar definition see [Debenham, 1995].

• **Group ED: External Design.** The *external design* phase begins with the conceptual model and constructs the "external model". The *external model* is a functional and implementation independent description of the IDSS [Debenham, 1995].

• **Group ID: Internal Design.** The *internal design* phase begins with the external model and constructs the internal model which is a a functional and implementation dependent description of the IDSS [Debenham, 1995].

• **Group RA: Software Reuse Analysis.** *Software Reuse Analysis* is defined as the process of using existing software components in the construction of software systems with the goal of reducing development costs [Biggerstaff et al., 1989]. In our framework we have separated "software reuse analysis" from the other development phases, because it is an important aspect of IDSS construction [Blair et al., 1995]. It is difficult to investigate software reuse when it distributed across all the development phases.

• **Group UI: User Interface Design.** *User interface design* is the process of specifying the interaction between the decision makers and the IDSS. Our framework incorporates user interface design because the user interface design is an important part of building an effective IDSS [Angehrn et al., 1990].

3. METHODOLOGIES FOR IDSS DESIGN

In this section we review the five major formal design methodologies which have been proposed for designing IDSSs.

3.1 Expert-Based Systems Methodology

The Expert-Based Systems (EBS) design methodology [Goul et al., 1987] is based largely on the ROMC paradigm which focuses on "Representations", "Operations", "Memory aids" and "Control mechanisms". The *representation* defines how the IDSS is structured and is presented to the user. The *memory aids* are tools which allow the user to record personal insights and observations. They allow the user to clarify definitions, and include such things as scratch pads, dictionary, remark and session traces. The *operations* are the intrinsic features that cannot be changed by the user, and the *controls* are the selections a user can make during system use. The major steps of this design methodology are defined below:

1) The domain expert selects an environment for the system to simulate that is familiar to the ultimate users. This environment gives rise to representations such as table of contents with chapter headings, a dictionary, and a scratchpad.
2) The developer decomposes, with help from the domain expert, the IDSS into chapters and sub-chapters (ie. units) and determines how they will be represented.
3) The developer asks the domain expert to describe four classes of operations which are: a) questions and possible answers to be posed by the system in a given chapter, b) suggestion by the system to explore other chapters, c) advice to be presented to the user, d) overall conclusions that should be presented to the user.
4) The memory aids which the IDSS will provide to the user are defined.
5) The "orientation" of the advice to be included in the system is determined by the developer and domain expert. The *orientation* is a category of advice which is offered.
6) The reasoning in each of the chapters is integrated by the developer.
7) A prototype of the system is implemented by the developer. The domain expert reviews this version.
8) The domain expert's comments are used during the construction of the system.
9) The "integrator module" which integrates the reasoning between the chapters of the system is implemented.
10) A test is conducted to see whether the system improves the quality of decision making.

3.2 Intelligent Decision Systems Methodology

The second major IDSS design methodology is the Intelligent Decision Systems (IDS) methodology by Holtzman [1989]. This methodology consists of two distinct methods: the "deterministic attention-focusing method" and the "probabilistic decision method"; performed in this order. The *deterministic attention-focusing method* constructs a deterministic decision model of the decision making problem and removes those variables in the model which are not sensitive to variations in their value. The *probabilistic decision method* is the process of enriching the decision model with probability measures and with a utility function. The phases

of these two methods are outlined below. The "phases" are performed iteratively to refine the decision model - ie. to improve the accuracy and completeness of the model.

The attention-focusing method consists of three phases: "basis development", "deterministic sensitivity analysis" and "deterministic basis appraisal" - performed in this order. *Basis development* is the construction of the *decision basis* which is a formal model of a decision problem in its most comprehensive form. *Deterministic sensitivity analysis* is the process of reducing the size of the decision basis by eliminating unimportant variables (ie. non sensitive variables). *Deterministic basis appraisal* is the process of reviewing the decision basis to determine that it is complete, accurate and that all unimportant variables have been removed.

The probabilistic decision method has three phases: "probabilistic and risk encoding", "decision-theoretic computations" and "recommendation and basis appraisal". The *probabilistic and risk encoding phase* consists of probabilistic and risk-attitude assessment and the evaluation of probabilistic and risk sensitivities. The *decision-theoretic computation phase* produces an optimal policy for the decision making problem. The *basis appraisal phase* reviews the decision basis for the purpose of investigating the effect of unexplored alternatives.

3.3 MEDESS Methodology

Another major IDSS design methodology is the MEDESS Methodology [Van Weelderen, 1991]. This methodology is based on the DSS development framework proposed by Sol [1990], which distinguishes between a "way of thinking", a "way of modelling", a "way of working" and a "way of control". The *way of thinking* describes three perspectives from which a problem situation is considered. These three perspectives are the micro-, the meso- and the macro-perspective, and are concerned with how the performance of a decision-maker, organisation, and cooperating organisations can be improved (respectively). The *way of modelling* defines the specific way the DSS is modelled. The *way of working* defines the approach which a developer employs to produce a complete and accurate description of the DSS. The *way of control* addresses the balance between the efficiency and the effectiveness of a DSS design process.

MEDESS requires that in the "way of modelling" perspective the developer address four problems: "systelogical", "infological", "datalogical" and "technological" problems. The *systelogical problem* describes the difficult problems a decision maker solves and the performance of the domain expert in solving the problem. The *infological problem* defines which information is processed by a decision maker to solve a problem. The *datalogical problem* describes how information is processed and grouped, without taking into account which technology is used to achieve this. The *technological problem* specifies how information is processed, thereby taking into account which technology is applied to achieve this.

MEDESS is divided into two major phases, the "understanding" and the "design" phase. In the *understanding phase* the current problem situation is conceptualised and specified, and in the *design phase* the new problem situation is conceptualised and specified. The problem situation of both phases are defined by analysing the micro, meso and macro-perspectives and modelling iteratively the systelogical, infological, datalogical and technological problems of each perspective.

3.4 Visual Interactive Approach

The fourth major design methodology is the Visual Interactive Modeling (VIM) approach proposed by Angehrn et al. [1990]. The developers of this methodology believe that the main goal of IDSSs is to provide decision makers with tools for interactively exploring, designing, and analysing decision situations. The VIM methodology is based on two design principles "usability prior to functionality" and "active cooperation". The first principle requires that the user interface of the IDSS be designed before the functions are analysed and specified; note the other four design methodologies in our study specify the functions of an IDSS before designing the user interface. The second principle requires the IDSS be an advisor and facilitator to the decision makers.

The methodology is divided into two major phases. The first phase of the methodology is: 1) to analyse the decision context of the IDSS, 2) to determine an appropriate language and form of communication between the IDSS and decision makers, and 3) to identify notions, concepts, and operations which are familiar to the decision maker and that correspond to his/her knowledge and experience. The second major phase is to identify the fundamental constructs that support the visual interactive environment of the IDSS.

3.5 Text Analysis Approach

The Text Analysis Approach (TAA) [McGovern et al., 1991] is an IDSS design methodology for performing the KA/RA phase of IDSSs. The developers of TAA believe that conventional KBS methodologies for the KA/RA phase do not support the entire KA/RA phase of IDSSs and methodologies need to be constructed for IDSSs. In TAA, text analysis with Influence Diagrams are used as a basis for knowledge acquisition. The process of eliciting knowledge is accomplished by expressing the problem as text, and then analysing the text to extract the variables or more abstract descriptions of actions, events and outcomes. The relationships are then identified and reduced to influences and informational links. The relationships and variables are then used to construct an Influence Diagram. After the "first cut" Influence Diagram has been developed the next phase is to examine each of the nodes in the diagram to check for clarity and to expand them if necessary.

4. RESULTS OF THE COMPARISON

The results of the comparison are shown in Table 1. Each row of the table represents a question in our proposed framework, and each of the five IDSS design methodologies are represented as a column in the table.

Five responses are used to indicate how well a methodology answers a question. They are "Yes", "No", "Few", "Existing" and a bullet (ie. "•"); *Yes* means that the methodology does precisely what the question asks, *No* means that it does not support what the question asks, *Few* means that it does some of what the question asks, *Existing* means that its does what the question asks by using other existing design methodologies, and a bullet means that the question is not applicable to the methodology.

As shown by the table, the five design methodologies provide little support for the entire IDSS development process. Most of the support these methodologies provide is for the KA/RA phase. Little support is provided for the conceptual, external, internal and reuse analysis phases.

ID	Description	EBS	IDS	MEDESS	VIS	TAA
K1.	Investigate class of decisions	Few	Yes	Few	No	No
K2.	Knowledge elicitation	Existing	Yes	Existing	No	Existing
K3.	Multiple knowledge sources	No	No	No	No	No
K4.	Decision maker and Expert	Yes	Yes	No	Yes	No
K5.	Circumstances and preferences	Yes	Yes	No	Yes	No
K6.	Uniform analysis approach	No	No	No	No	No
K7.	Decomposition and abstraction	Few	Few	Few	Few	Few
K8.	Four Decision Making Phases	No	Yes	No	No	No
K9.	Graphical Models	Yes	Yes	Yes	Yes	Yes
K10.	Languages for describing	No	Existing	No	Existing	Existing
K11.	Validation and verification	Yes	Few	No	No	No
C1.	Easy transition	Yes	Yes	No	Yes	•
C2.	Decomposition and abstraction	Few	Few	Few	Few	•
C3.	Normalisation	No	Few	No	No	•
C4.	Uniform approach for modelling	No	No	No	No	•
C5.	Graphical Model	Yes	Yes	Yes	Yes	•
C6.	Rich language for knowledge	No	No	No	No	•
C7.	Knowledge independent from use	No	No	No	No	•
C8.	Support for risks and uncertainties	No	Yes	No	No	•
E1.	Easy transition	Yes	Yes	Yes	Yes	•
E2.	Selection of algorithms	No	No	No	No	•
E3.	Decision analytical techniques	No	Few	No	No	•
E4.	External model	No	No	No	No	•
E5.	Functional operations	No	No	No	No	•
I1.	Internal Model	No	No	No	No	•
I2.	Ease of transition	No	No	No	No	•
I3.	Defining operational constraints	No	No	No	No	•
I4.	Deciding what should be deduced	No	No	No	No	•
I5.	Support risk and uncertainty	No	Yes	No	No	•
I6.	Decision analytical techniques	No	Few	No	No	•
I7.	Deciding language to implement	No	No	No	No	•
R1.	Searching for components	No	No	No	No	No
R2.	Understanding components	No	No	No	No	No
R3.	Interconnection of components	No	No	No	No	No
R4.	Modifying components	No	No	No	No	No
R5.	Evaluating component feasibility	No	No	No	No	No
R6.	Support all development phases	No	No	No	No	No
U1.	Usability before functionality	No	No	No	Yes	No
U2.	Representation schema	No	No	No	Yes	No
U3.	Defining collaboration	No	No	Yes	Yes	No
U4.	Decomposition and abstraction	Few	No	No	Few	No
U5.	Manipulation of interface	No	No	No	Yes	No
U6.	Rules for visual language	Few	No	No	Yes	No
U7.	Ease of Transition	Yes	No	No	No	No

Table 1. Comparison of IDSS Design Methodologies Using Our Framework

The IDS methodology appears to provide the most support in comparison with the other four methodologies. In particular, it provides the most support for performing the KA/RA phase. For instance, it uses a language for describing requirements and encourages the developer to analyse the four decision making phases which an IDSS can support.

The TAA methodology provides the least support compared to the other methodologies. Although this methodology is for performing the KA/RA phase only, it is the worst methodology for performing this phase. This methodology is to simplistic and does not provide many guidelines on how to break down the complexity of gathering the knowledge for a difficult decision.

5. CONCLUSIONS

In this paper we have compared five major formal methodologies for designing IDSSs. Our comparison has identified that these methodologies are vastly different in their approach for designing IDSSs and that they provide little support for the conceptual, external, internal and reuse analysis phases. We are currently constructing an IDSS design methodology to address the shortcomings associated with these methodologies. Early versions of our methodology are described in [Blair, 1994a] [Blair, 1994b][Blair et al., 1994].

6. REFERENCES

Angehrn A. A. and Luthi H. J. (1990) Intelligent Decision Support Systems: a Visual Interactive Approach, Interfaces, 20, 6, Nov-Dec., 17-28.

Blair, A. (1994a) Inter-connection of Software Components in Intelligent Decision Support Systems, The Australian Computer Journal, February, 11-19.

Blair, A. (1994b) A Design Methodology for Knowledge-Based Decision Support Systems, International Conference on Expert Systems for Development Specifications, Thailand, 27-31 March, 18-23.

Blair, A., Debenham, J. K. and Edwards, J. (1994) Requirements Analysis for Intelligent Decision Support Systems, submitted to the Australian and New Zealand Conference on Intelligent Information Systems , Brisbane, 30 November - 2 December, 1994.

Blair A., Debenham J. K, and Edwards J. (1995) A Comparative Study of Methodologies for Building IDSSs, submitted to Journal of Decision Support Systems.

Clark, J. A. (1992) A Framework for Comparing Knowledge Based System Design Methodologies, AI'92, Hobart.

Biggerstaff, T. J. and Perlis, A. J. (1989) Software Reusability: Concepts and Models, ACM Press, New York, Volume 1.

Bonarini, A. and Maniezzo, V. (1991) Integrating Expert Systems and Decision Support Systems: Principles and Practice, Knowledge-Based Systems, 4, 3, 172-176.

Bryd, T. A., Cossick, K. L. and Zmud, R. W. (1992) A Synthesis of Research on Requirements Analysis and Knowledge Acquisition Techniques, MIS Quarterly, March, 117-138.

Buckner, G. and Shah, V. (1991) Intelligent Decision Support Systems, The Journal of Computer Information Systems, 31, 2, 61-66.

Dalal, N. P. and Yadav, S. B. (1992) The Design of a Knowledge-Based Decision Support System to Support the Information Analyst in Determining Requirements, Decision Sciences, 23, 6, November, 1373.

Debenham, J.K. (1995) A Unified Approach to Requirements Specification and System Analysis in the Design of Knowledge-Based Systems, in proceedings of the 7th International Conference on Software Engineering and Knowledge Engineering (SKE'95), Maryland, USA, June 22-24, 144-146.

Fensel, D. and Poeck, K. (1994a) A Comparison of Two Approaches to Model-based Knowledge Acquisition, KADS-Meeting 94, March 24-25, Sankt Augustin, Germany.

Fensel, D. and Van Harmelen, F. (1994b) A Comparison of Languages which Operationalize and Formalise KADS Models of Expertise, The Knowledge Engineering Review, 9, 2, 105-146.

Gottinger, H. W. and Weimann, P. (1992) Intelligent Decision Support Systems, Journal of Decision Support Systems, 8, 4, 317-332.

Goul, M. and Tonge, F. (1987) Project IPMA: An Application of Decision Support System Design Principles to Building Expert-Based Systems, Decision Sciences, 18, 448-467.

Holtzman, S. (1989) Intelligent Decision Systems, Addison-Wesley, New York.

McGovern, J., Samson, D. and A. Wirth (1991) Knowledge Acquisition for Intelligent Decision Systems, Decision Support Systems, 7, 3, , August, 253-261.

Saxena, K. B. C. and Siu, B. K. T. (1989) DSS Development Methodologies: A Comparative Review, Tech. Report, TR-89-05, Department of Computing Studies, Hong Kong Polytechnic, Hong Kong, September 13.

Saxena, K. B. C. (1991) Decision Support Engineering: A DSS Development Methodology, IEEE, 98-109.

Sharp, H. (1994) Knowledge Acquisition and Requirements Analysis ?, The 2nd International Conference on Expert Systems for Development, Bangkok, Thailand, March 28-31.

Sol, H. G. (1990) Information systems to Support Decision Processes, in H. G. Sol, and J. Vecesenyi, (eds), Environments for Supporting Decision Processes Amsterdam, North-Holland: PNHC, 115-127.

Van Weelderen, J. A. (1991) MEDESS: A Methodology for Designing Expert Support Systems, PhD Dissertation, Technical University Delft.

APPENDIX A
FRAMEWORK FOR COMPARING IDSS DESIGN METHODOLOGIES

Group KA: Knowledge Acquisition / Requirements Analysis

K1. Does the analysis process require the developer to investigate the "class of decisions" which the IDSS will support ? [Holtzman, 1989]

K2. Is there a specific procedure for eliciting knowledge from the domain expert ? [Clark, 1992]

K3. Is there a specific procedure for the inclusion of knowledge from knowledge sources of various types (human experts, documentation, experimentation, observation and induction) ? [Clark, 1992]

K4. Do the decision maker and human expert interact closely with the requirements analysis process ? [Holtzman, 1989]

K5. Does the analysis process model the circumstances and preferences which each decision maker might input into the IDSS ? - does it model the different views of the decision makers ? [Holtzman, 1989] [Sol, 1990] [Saxena et al., 1989].

K6. Is a uniform approach used to gather the inter-disciplinary requirements of the IDSS ?

K7. Are there techniques for abstracting and decomposing the inter-disciplinary requirements ? [Holtzman, 1989]

K8. Does the analysis process encourage the analyst to examine how the IDSS will support the decision makers through the four decision making phases ? [Holtzman, 1989]

K9. Are there models for representing graphically the inter-disciplinary requirements ? [Holtzman, 1989] [Sol, 1990]

K10. Are there languages for describing the inter-disciplinary requirements ? [Fensel et al., 1994a]

K11. Are there techniques for validating and verifying the requirements ? [Fensel et al., 1994]

Group CD: Conceptual Design

C1. Is there an easy transition from the RA/KA phase to the conceptual design phase ?

C2. Are there methods for abstracting and decomposing the conceptual design of the IDSS ? [Fensel et al., 1994b]

C3. Are there techniques for removing redundancy from the conceptual model ? - ie. are there techniques for normalising the conceptual design ? [Clark, 1992]

C4. Is the conceptual design modelled using a uniform approach ?

C5. Is there a graphical model to represent the conceptual design ? [Fensel et al., 1994b]

C6. Is the design language rich in the sense that it allows the expression of different kinds of knowledge by different language primitives ? In particular, how does the language represent concepts, properties, values, relations and structures ? [Fensel et al., 1994b]

C7. Can the knowledge domain be expressed independently from its use ? Is it free from control knowledge ? [Clark, 1992]

C8. Is there support for modelling the decision maker's risks and uncertainties ? [Holtman, 1989]

Group ED: External Design

E1. Is there an easy transition from the conceptual design phase to the external design phase ?

E2. Is there support for deciding which types of algorithm are best for implementing the IDSS ?

E3. Is there any support for modelling the decision analytical techniques which are provided by the IDSS ?

E4. Is there an external model and is there a representation scheme that supports it ? [Clark, 1992]

E5. Can the functional operations of the IDSS be expressed in the external model (that is, functional update types and functional query types) ? [Clark, 1992]

Group ID: Internal Design

I1. Is there an attempt to derive and represent the internal model of the IDSS ? [Clark, 1992]

I2. Is there an easy transition from the external design phase to the internal design phase ?

I3. Is there support for defining the operational constraints of the IDSS ? [Clark, 1992]

I4. Does the methodology help the designer with determining what data/information should be stored and what data/information should be deduced ? - with the objective of constructing an IDSS that has optimal system performance. [Clark, 1992]

I5. Is there support for implementing the risks and uncertainty of the decision makers ? [Holtzman, 1989]

I6. Is there support for specifying how the decision analytical techniques will be implemented? [Holtzman, 1989]

I7. Is there support for determining which language is the most suitable for implementing the inter-disciplinary methods of the IDSS ? [Holtzman, 1989]

Group RA: Reuse Analysis

The following questions have been derived from [Biggerstaff et al., 1989].

R1. Are there techniques for searching the organisation for potential software components ? - ie. are there techniques which help the developer to reverse engineer ?

R2. Are there techniques which help the developer with understanding the functionality associated with existing software components ?

R3. Are there techniques for assisting the developers with the interconnection of existing and hand coded components within the IDSS ?

R4. Are there techniques which assist the developers with modifying existing software components so that they can be reused in the IDSS ?

R5. Are there techniques for evaluating the feasibility of reusing existing software components in IDSSs ?

R6. Do the reuse techniques support all phases of the development process ? - eg, requirements analysis phase, conceptual design phase, etc.

Group UI: User Interface Design

U1. Is usability considered before functionality ? [Angehrn et al., 1990]

U2. Is there a user interface model and is there a representation scheme that supports it ? [Angehrn et al., 1990]

U3. Is there support for defining how the decisions makers will cooperate with the IDSS ? [Holtzman, 1989]

U4. Does the methodology offer any solutions that break down the complexity of designing the user interface ?- eg, the methodology might suggest an incremental design of the user interface ? [Angehrn et al., 1990]

U5. Does the methodology help the analyst with designing user interfaces which allow the decision makers to manipulate the decision-making model of the IDSS ? [Angehrn et al. 1990]

U6. Are there well-established rules for designing the visual language of the IDSS ? [Angehrn et al., 1990]

U7. Is there an easy transition between this design phase and the other design phases ?

MODELLING TEAMS AND TEAM TACTICS IN WHOLE AIR MISSION MODELLING*

Gil Tidhar
Australian Artificial Intelligence Institute
171 LaTrobe Street
Melbourne, Australia
gil@aaii.oz.au

Mario Selvestrel
Air Operations Division, DSTO
506 Lorimer Street
Fishermens Bend, Australia
Mario.Selvestrel@dsto.defence.gov.au

Clinton Heinze
Air Operations Division, DSTO
506 Lorimer Street
Fishermens Bend, Australia
Clinton.Heinze@dsto.defence.gov.au

ABSTRACT

The problem of whole air mission modelling is part of a larger problem which is the problem of simulating possible war-like scenarios in the air, sea, and on land. In such modelling systems one is required to model the behaviour of various actors and the resources that are available to them. One aspect of this problem is the modelling of a group of actors as a team and then modelling the coordinated behaviour of such a team to achieve a joint goal.

In the domain of air mission modelling the actors are pilots that control aircraft and their behaviour is referred to as tactics. In this paper we present the approach we adopted in modelling teams and team tactics as part of the development of the Smart Whole AiR Mission Model (SWARMM) for the DSTO, Air Operations Division.

MODELLING TEAMS AND TEAM TACTICS IN WHOLE AIR MISSION MODELLING*

Gil Tidhar
Australian Artificial Intelligence Institute
171 LaTrobe Street
Melbourne, Australia
gil@aaii.oz.au

Mario Selvestrel[†]
Air Operations Division, DSTO
506 Lorimer Street
Fishermens Bend, Australia
Mario.Selvestrel@dsto.defence.gov.au

Clinton Heinze
Air Operations Division, DSTO
506 Lorimer Street
Fishermens Bend, Australia
Clinton.Heinze@dsto.defence.gov.au

ABSTRACT

The problem of whole air mission modelling is part of a larger problem which is the problem of simulating possible war-like scenarios in the air, sea, and on land. In such modelling systems one is required to model the behaviour of various actors and the resources that are available to them. One aspect of this problem is the modelling of a group of actors as a team and then modelling the coordinated behaviour of such a team to achieve a joint goal.

In the domain of air mission modelling the actors are pilots that control aircraft and their behaviour is referred to as tactics. In this paper we present the approach we adopted in modelling teams and team tactics as part of the development of the Smart Whole AiR Mission Model (SWARMM) for the DSTO, Air Operations Division.

1 INTRODUCTION

Modelling the behaviour of teams is a problem that concerns many analysts who are attempting to model the behaviours of groups of humans and also concerns researchers in Distributed Artificial Intelligence (DAI) who are attempting to model the behaviour of groups of artificial agents. Such attempts include building modelling systems for business processes [Fox93] and power grid monitoring [Jen93]. The problem of modelling the activity of teams of artificial agents [GK93, KLR+92] is a combination of two sub-problems: the first is the modelling of the team itself [Tid93, Wer90] and the second is the modelling of the team activity [Fox81, JM92, KLR+92].

The problem of modelling teams and team behavior includes a wide range of sub-problems. Such sub-problems include the problem of representing teams with a variety of organizational structures and providing a representation of team behavior that allows the implementation of a variety of communication, coordination, and control mechanisms. This problem is further enhanced when considering real-time systems that are embedded in a dynamic environment.

Here we describe the approach taken to the modelling of teams and team behaviour as part of the modelling of whole air missions. This work is part of the development of the Smart Whole AiR Mission Model (SWARMM) [ASH+94] system. The purpose of the SWARMM system is to simulate the physics of whole air missions, simulate the pilot reasoning involved in such missions, and to interface with advanced visualization software to enable better understanding of whole air missions.

Whole air mission modelling comprises a set of activities related to the flying of a combat mission by a group of aircraft. The type of aircraft selected for a particular mission will depend first and foremost upon the type of mission that is to be flown. The type of tasks relevant to whole air mission modelling include:

Air Defence - The use of aircraft to defend an area against an attack from the air.

Attack - The use of aircraft to attack a ground target.

Sweep - The use of fighter aircraft to clear a path for an incoming strike force.

Escort - An attachment of fighter aircraft which closely accompany the attack aircraft to defend them against

attack by enemy aircraft.

We refer to the behaviour of the pilots as *tactics* and to the behaviour of a group of pilots as *team tactics*. Teams are represented as a set of sub-teams that have adopted structures that describe their inter-team relationships. These relationships include an allocation of organizational and functional responsibilities. The team tactics are modelled as plans that are executed by the individual members of the team. The choice of the team tactics depends on the task to be achieved, the particular situation, the structure adopted, and the availability of communication facilities.

The tactics of individual pilots involve carrying out a sequence of sub-tasks or manoeuvres. Each of these tasks can further be divided into smaller sub-tasks. For example, if the pilot decides to go into attack, the bearing to the nearest target has to be determind; having determined this the course has to be changed towards it. Changing the course might involve a sequence of manoeuvres. Thus, the formalism must be able to capture sequential, parallel, iterative, and non-deterministic actions.

At any instant the pilot might be able to employ a number of different tactics but may be executing only one of them. Also, while executing a particular tactic the pilot might have already decided (i.e., intended) to execute other tactics some time in the future. Thus, the representational formalism needs to distinguish between the pilot having, executing, and intending a particular set of tactics [GL86, RMSM92].

A combat scenario usually involves more than one aircraft from the same side. Aircraft are organized into tactical units and have specific roles assigned to them. For example, Howlett [How91] shows how a mission to defend an air base might be accomplished by four sector Combat Air Patrols (CAPs) and two back-up CAPs. Any mission involving multiple aircraft is accomplished by adopting team tactics. One must be able to decompose hierarchically the team tactics based on the organization of pilots. Team tactics give rise to notions such as mutual beliefs, joint goals, and joint intentions which are the beliefs, goals, and intentions shared by multiple pilots or teams of pilots.

The dynamic nature of whole air mission scenarios means that aircraft can dynamically reorganize themselves. For example, when two aircraft from different groups are shot down, the remaining aircraft of each group may combine together to form a single group. Dynamic reorganization of aircraft may also force dynamic reassignment of roles. The representational formalism for team tactics must be capable of dynamic reorganization and dynamic reassignment of roles.

The underlying system used for modelling the teams and implementing the team tactics is the Distributed Multi-Agent Reasoning System (dMARS) [Aus94]. Previous work on this problem has included the development of a demonstrator system [RMSM92] that has investigated the ability to model team tactics using the Procedural Reasoning System [GL86]. In that work teams were represented as special agents that did not correspond to any particular aircraft or pilot and that executed the team tactics by instructing the sub-teams to perform particular sub-tasks. This has allowed only for a centrally coordinated type of team behaviour. In this work we are implementing an operational system and have taken a different approach that allows for a distributed control of team behaviours.

Different types of coordinated behavior have also been implemented using the Soar architecture [LJN94, LNR87] although these experiments have focused primarily on the coordination required between two fighter aircraft flying in formation. Furthermore, this work has not considered the coordination required in a large scale air mission and across the command hierarchy.

In Section 2 of this paper we describe the underlying technology used in the development of the SWARMM system and a typical scenario that is modelled. In Section 3 we describe the approach taken in modelling teams and the way teams are implemented and in Section 4 we describe the approach taken in modelling different team tactics and the way these tactics were implemented. We conclude this paper in Section 5 with a short discussion.

2   THE UNDERLYING TECHNOLOGY

The problem of modelling whole air missions has been divided into two components (or sub-problems) each modelled (or solved) using different technology. The first component is modelling the behavior of the physical elements and systems that are involved in whole air missions. These include the aircraft, the radar systems, the weapon systems, and the communication systems. The modelling of the physical systems also includes the modelling of the pilot's body, e.g., eyesight, and the effect of various manoeuvres on the pilot's physical state. To model this component we used a Fortran-based computer model called Piloted Air Combat AUStralia (PACAUS) [Ste93].

The second component is modelling the reasoning processes of the pilot. These processes include the process of determining the current situation (referred to as *Situation Awareness*) and the process of choosing and executing the best tactics for the current situation (referred to as *Tactics Selection*). To model this component we used the Distributed Multi-Agent Reasoning System (dMARS) [Aus94].

In the model adopted we view the combination of the pilot's reasoning and the aircraft's physical systems as a single entity which we will refer to as an *aircraft*. We assume that the pilot's reasoning processes receive input from other components/sensory equipment (referred to as the *sensed world*) and provides instructions to other components/equipment (referred to as *pilot response*).

2.1 PACAUS

The PACAUS system is a time stepped simulation of combat between many aircraft from two opposing sides. The models of the hardware and their tactical use comprise in excess of fifty thousand lines of Fortran code. An advanced graphical post-processing program allows three dimensional visualisation of the combat [Mus93].

Input files allow the user to specify the initial scenario and to define certain flags to control the use of the weapons systems, the radar systems, and the aircraft themselves.

There are several aerodynamic models of combat aircraft. The on-board systems such as the radar, the Radar-Warning-Receiver (RWR), and counter-measure facilities such as chaff, flares, and Electronic Counter-Measures (ECM) are modelled by the system. There are simulations of a variety of weapons including fully active and semi-active missiles, infrared heat-seeking missiles and guns. The PACAUS program has been utilized for several research programs for the Royal Australian Air Force and is currently being expanded in several important areas.

2.2 DISTRIBUTED MULTI-AGENT REASONING SYSTEM (dMARS)

The Distributed Multi-Agent Reasoning System (dMARS) is an agent-oriented distributed real-time system. It provides a representational framework and reasoning mechanisms for implementing agents.

Each agent is composed of a set of beliefs, goals, plans, and intentions. The beliefs of dMARS agents provide information on the state of the environment as perceived by the agent and are represented in a first-order logic. For example, the belief that the range from aircraft WARLOCK to an aircraft BOGGY1 is 40 miles can be represented by the statement (range WARLOCK BOGGY1 40). Variables are denoted by the character $, e.g., $target denotes that variable target.

The goals of dMARS agents are descriptions of desired tasks or behaviours. In the logic used by dMARS, the goal to achieve a certain condition C is written as (! C); to test for the condition is written as (? C); to wait until the condition is true is written (^ C); to assert that the condition is true is written as (=> C); and to retract a condition is written as (-> C).

Plans are declarative procedural specifications that represent knowledge about how to accomplish given goals or react to certain situations. Each plan consists of a body, an invocation condition, and a context condition [GL86]. The set of plans in a dMARS application system also includes meta-level plans, that is, information about the manipulation of the beliefs, goals, and intentions of the dMARS agent itself.

The body of a plan can be viewed as a procedure or a tactic. It is represented as a graph with one distinguished start node and one or more end nodes. The arcs in the graph are labelled with the sub-goals to be achieved in carrying out the plan. The invocation condition describes the events that must occur for the plan to be executed. Usually, these events consist of the acquisition of some new goals (in which case, the plan is invoked in a goal-directed fashion) or some change in system beliefs (resulting in data-directed invocation) and may involve both. The context condition describes contextual information relevant for the execution of the plan.

The intention list contains all those tasks that the system has chosen for execution, either immediately or at some later time. An intention consists of some initial plan, together with all the sub-plans that are being used in attempting to execute successfully that plan. At any given moment, the intention list of an agent may contain a number of such intentions, some of which may be suspended or deferred, some of which may be waiting for certain conditions to hold prior to activation, and some of which may be meta-level intentions. Only one intention can be executed at any given moment and the choice of that intention depends on the perceived state of the world and the priority of that intention.

In some applications, it is necessary to monitor and process many sources of information at the same time, e.g., simulating a number of pilots. To facilitate this, dMARS was designed to allow several agents to run in parallel. Although the perceptual input received by each agent may come from the same physical world, each agent has its own database, goals, and plans, and reasons asynchronously relative to other agents, communicating with them by sending messages.

2.3 A TYPICAL SCENARIO

Let us consider two opposing forces. Red Team is planning a strike mission to destroy a ground target within Blue Team's territory. Red Team assembles a *package* of aircraft operating in several different roles. There is a group of sweepers to clear a path ahead of the strike aircraft, there are escort aircraft to accompany the strike aircraft and of course there are the strike aircraft themselves. These three distinct elements have a single goal: to attack successfully the ground target which has been designated for them. They each have assigned responsibilities within the mission that require communication and interaction within the group.

Blue Team will have aircraft operating in the air-defence role protecting their airspace from the incursion by Red Team. These aircraft will either be launched from an airbase when it becomes apparent that an attack is imminent or, if hostile actions have been occurring for a number of days, the aircraft may be flying patrols over an area where an attack is expected.

The hierarchy of command within the team exists in a flexible dynamic way to allow the team to operate at several levels and to split and reform as the situation dictates [Sha85]. Within the operation of a standard mission there will be aircraft performing different tasks. An escort

aircraft may accompany a strike aircraft as its wingman. A pair of lower performance fighter aircraft might accompany a pair of higher performance fighter aircraft to give the illusion of four high performance fighters.

Each situation may require the use of a different command and control structure. Thus the sub-teams will adopt different command and control roles within the team. The different mission goals adopted by the team may require the sub-teams to adopt functional responsibilities with respect to the conduct of the mission. Thus an aircraft may have both a command and control role (e.g., a leader) and a functional role (e.g., an escort). This two-fold responsibility impacts upon the way in which tactics are chosen and the way in which the mission is conducted.

3 MODELLING TEAMS

The aircraft in whole air missions are identified as *singletons* and have unique names. Aircraft are teamed together into pairs, groups, and packages and again each team has a unique name. As the name *pair* indicates, pairs are teams of two aircraft. Groups are teams made up of pairs and/or singletons. Packages are teams made up of groups and/or pairs and/or singletons. For each such team the various teams of which it is made up are referred to as its *sub-teams*.

Each of the sub-teams is assigned at least one role in the team.[1] The role identifies the sub-team's relationships with other sub-teams and the responsibilities it has towards the various functions of the team. We identify two types of structures that are imposed on the team and that correspond to two types of roles. The first is an organizational structure that defines the Command and Control functions in the team, and which is completely hierarchical. We refer to the roles in this structure as *organizational roles*. The second is a functional structure that defines the functional expertise and responsibilities in achieving the task that the team is set to achieve, and does not incorporate any notion of hierarchy. We refer to the roles in this structure as *functional roles*. This model of a team is similar to the model described by Tidhar [Tid93].

Typically, the teaming and naming of aircraft, as is the role assignment, is done prior to the mission in a briefing session in which the team members are briefed by their commanding officer. Due to the dynamic nature of the domain, teams can be dismantled and re-formed dynamically. As teams change their structure(s) in response to the situation, roles can be dynamically re-assigned.

3.1 ROLES

In the organizational structure we identify two types of roles, a *leader* and a *wingman*. Each team has only one

sub-team as the leader but there can be several sub-teams as a wingman although typically there is also only one wingman. The leader sub-team is responsible to make all the decisions for the team and to instruct or inform the wingman on particular decisions or information. The wingman is responsible for following and obeying the leader and providing it with information that may assist it in making decisions.[2] Each sub-team has beliefs about the other sub-teams and their roles and reacts to changes to the perceived world according to its roles.

Functional roles correspond to the way the team achieves its tasks and the responsibilities of each of the sub-teams towards achieving the task. The choice of functional roles depends on the tasks that the team is set to achieve. The way the assigned role determines the responsibilities of the sub-team assigned to this role is via the team tactics (see Section 4).

Role assignment is part of the process of team formation. The assignment of roles typically depends on the skills and capabilities of the sub-team. These in turn depend on the hardware setup of the various aircraft in the sub-team, e.g., aircraft type, weapon systems, etc., and the tactical knowledge held by the pilots. When a team is assigned a role all its sub-teams are aware of this assignment and can act accordingly.

Example 1

Consider the team WARLOCK-12 that is part of the RED team and which has two singleton sub-teams, WARLOCK-1 and WARLOCK-2, that are the LEADER and WINGMAN respectively. Given that WARLOCK-12 has to fly in formation towards a particular target, the leader, WARLOCK-1, will determine and fly along a course. The wingman, WARLOCK-2, will simply follow WARLOCK-1 while flying in some predetermined disposition. In the case of any observed unique events, WARLOCK-2 will communicate its observations to WARLOCK-1 and wait for a command.

The pair WARLOCK-12 is teamed with the pair JESTER-12 to form the group PYTHON. WARLOCK-12 is the LEADER and JESTER-12 is the WINGMAN. In the context of the team PYTHON, the teams WARLOCK-12 and JESTER-12 are also assigned the functional roles ATTACK and ESCORT respectively. The group PYTHON is teamed with the singleton GRIZZLY to form the package THUNDER. In the context of the team THUNDER the teams PYTHON and GRIZZLY are assigned the functional roles STRIKE and TANKER respectively.

□

Note that THUNDER has only functional roles. This means that the coordination and decision making processes are implicit in the team tactics and that each sub-team is responsible to inform other sub-team of any change to the tac-

---

[1] Each sub-team may be assigned more then one role, but it is the responsibility of the designer to ensure that a sub-team is not assigned two conflicting roles as is the case in real missions.

[2] Note that the leader of a team can be a sub-team and hence that sub-team's leader will make the decisions for the whole team. If that leader is also a team then its leader will make the decision for the whole team, and so on and so forth until the leader is a singleton.

tics. This type of behaviour corresponds to the notions of commitment cf. Cohen and Levesque [CL91] and responsibility cf. Jennings and Mamdani [JM92].

3.2 IMPLEMENTATION

In dMARS the agent's beliefs are implemented as relations (or predicates) in a relational database. Since each subteam in a team has at least one role the team itself can be represented as a relation between the team, the sub-teams, and the role that the sub-team has been assigned in the team. We refer to this relation as `role-in-team`. The only teams that do not have sub-teams are aircraft. Such teams are identified with the predicate `singleton` (e.g., `(singleton WARLOCK-1)`).

Each agent is aware of its name (via the predicate `myname`) and can hence deduce its membership in different teams. Basically, if the agent has been assigned a role in a team then it is a member of that team. If that team has been assigned a role in another team then the agent is also a member of the other team. Not only can the agent deduce its membership in a team, it can also deduce the roles it has been assigned in that team. This knowledge allows the agent to adapt its behaviour as specified in the team tactics.

There are two additional aspects of a team that are required for the purpose of specifying the behaviour. The first is the set of members, that is all the aircraft that are either a sub-team of the team or members of any of the sub-teams. The second is the size of a team, that is the number of members. Both these aspects are implemented as functions that deduce this information from the relations `role-in-team` and `singleton`.

Example 2

Let us describe the beliefs of the `WARLOCK-1` pilot. The team `THUNDER` is comprised of the singletons `WARLOCK-1`, `WARLOCK-2`, and `GRIZZLY` and also the aircraft that are part of the team `JESTER-12`. The team `WARLOCK-12` has only organizational roles and is modelled with the predicates `(role-in-team WARLOCK-12 WARLOCK-1 LEADER)` and `(role-in-team WARLOCK-12 WARLOCK-2 WINGMAN)`.

In the context of the team `PYTHON` the sub-teams are assigned both organizational and functional roles and the team is modelled with the predicates:

```
(role-in-team PYTHON WARLOCK-12 LEADER)
(role-in-team PYTHON WARLOCK-12 ATTACK)
(role-in-team PYTHON JESTER-12 WINGMAN)
(role-in-team PYTHON JESTER-12 ESCORT)
```

The sub-teams of the team `THUNDER` are assigned only functional roles and the team is modelled with the predicates `(role-in-team THUNDER PYTHON STRIKE)` and `(role-in-team THUNDER GRIZZLY TANKER)`.

□

4 MODELLING TEAM TACTICS

Team tactics are used to ensure that the team works together as a coherent entity. There are two main types of methods to control and coordinate the team behaviour, namely centralized control and coordination, and distributed control and coordination. Distributed control and coordination, in turn, can be implicit or explicit.

The simplest method to control a team is through centralized control and coordination. With this method there is a team member who makes all the decisions and gives orders to the other team members. The team members themselves have no need to know why they are carrying out actions or how the actions contribute to the overall team tactics. Team members under centralized control blindly follow orders given by the team leader. For example, an air defence controller vectors a pair of fighters onto a certain course. The fighters do not necessarily need to understand why they are flying in that direction and obey the order.

On the other hand, team members under distributed control have an understanding of the team's task, the role of each team member, and how the team works together. These team members can cooperate by observing the world (implicit coordination), through communication messages (explicit coordination), or both.

Implicit coordination involves no communication between team members and it occurs when the members of the team observe the world in order to determine what action should be taken and when. Many team tactics used in whole air missions use implicit coordination. Typically, a team has practiced its standard procedures and tactics together until their execution has become instinctive. The team members do not need to communicate explicitly in order to perform the steps of a given tactic. Through practice the team knows, without overt communication, the role of individual team members and how each member fits into the overall tactic used to complete the task. During a briefing before a mission begins a team will discuss and confirm the particular tactics applicable to the mission. The mission then proceeds with each individual of the team executing the particular tactic implicitly defined by their role in the team and by the situation in which they find themselves.

Explicit coordination involves communication between team members to coordinate their actions. However, unlike centralized control where team members must follow instructions received, with explicit coordination the messages provide information which is evaluated before any action is taken. In the case of air combat, explicit coordination can occur through data links between aircraft or, more commonly, through radio messages.

Each form of control has its benefits and problems. A team member under centralized control has a relatively easy role with little reasoning to do, it is always told what to do. But, if it is not explicitly told what to do it cannot do anything. When using implicit control each team member has its own view of the world and confusion can occur when

(*told (intercept $team $target) told $env)

(and
  (myname $self)
  (role-in-team $team $self $role)
  (max-missile-range $max-range))

```
                              ( START )
         (? (== $role LEADER))        (? (== $role WINGMAN))

            [ P2 ]                              [ P3 ]
                                   (! (fly-manoeuver-until (int-right $target)
   (! (fly-manoeuver-until (int-left $target)           (< $range $max-range)))
                      (< $range $max-range)))
                                                  [ P9 ]
            [ P8 ]                   (! (fly-manoeuver (collision-course $target)))

   (! (fire-missile $target))
                                                  [ P15 ]
            [ P12 ]                   (^ (leader-missile-status $target FIRED))

   (! (radio-message $team $self                          [ P10 ]
          (missile-fired-at $target)))
                                (? (should-fire $target))
            ( END1 )                                (? (not (should-fire $target)))
                                          [ P14 ]

                                (! (fire-missile $target))

                                                  ( END2 )
```

Figure 1: Team tactics for intercepting a target.

team members hold conflicting views of the world. Explicit coordination is more reliable than implicit coordination because information is explicitly sent between members of the team. However, it depends on a large number of communication messages and confusion can occur if expected messages are not sent between team members.

In general, all three methods of control need to be used by tactics. Centralized control is necessary when the aircraft under control is relying on the controller to provide information which is otherwise unavailable, e.g., a controller providing intercept instructions for a target that the aircraft was unable to detect. Implicit coordination is used to limit exposure to the enemy through radio emissions. Every time a radio message is transmitted there is a possibility that the sender may be detected; this can make explicit coordination through radio and data link dangerous. However, when pre-cise coordination is required, e.g., to inform team members that a missile has been fired, explicit coordination is used.

Example 3

In practice all three methods of control and coordination can be used within the one tactic. To illustrate this we will examine a pincer intercept plan (Figure 1). The pincer intercept involves the team, with two members, forming into two distinct but cooperating elements in order to intercept an enemy aircraft: the leader element which attacks the target from the left and the wingman element which attacks the target from the right. The various stages of the intercept are coordinated using different methods. The following briefly describes the five stages of a pincer intercept:

- The order to commence a pincer intercept is received by the team members.

- The leader and wingman split to obtain lateral separation from the target.

- Once the range to the target is less than the missile's maximum range, the leader begins the missile firing procedures and the wingman changes course towards the target.

- When the leader fires the missile a radio call is made to indicate a missile has been fired.

- Once the leader has shot, the wingman decides whether it should also fire at the target.

□

4.1 MODELLING TEAM TACTICS IN dMARS

Tactics are modelled as plans within the dMARS environment. Each agent has plans which define the tactics available to the agent. In order to model coordinated team tactics, plans are written as sets defining the procedures to be used by each member (or agent) within the team. Each agent of the team executes the portion of the tactic which is relevant to itself. The plans or portion of a plan which must be executed by each team member can be differentiated through the context or by branching within the plan. For example, if a team plan has two members (leader and wingman), the part of the plan that is relevant to the leader can be determined by testing if the current agent is the leader in the context condition, e.g., (role-in-team $team $self LEADER), or branching within the body of the plan, e.g., (? (== $role LEADER)) (see Figure 1).

All three methods of coordination and control discussed above can be modelled using dMARS plans. The confusion aspects associated with the different tactics discussed above carry through to the dMARS models of the tactics. For example, in real whole air missions implicit coordination can lead to errors in executing a tactic because decisions are based on incorrect data; similar errors can be modelled in a dMARS plan.

4.2 CENTRALIZED CONTROL AND COORDINATION

Centralized control and coordination can be modelled by having an agent react to an incoming message without any need for processing or evaluating the message. Commencing the pincer intercept plan is an example of centralized control. The plan is invoked through a told event, e.g., (*told (intercept $team $target)). There is no decision-making process at the beginning of the plan that gives the pilot agent a choice to follow the intercept command or to ignore it. The first arc of the plan just determines which branch of the plan is relevant for the leader, e.g., (? (== $role LEADER)), and which is relevant for the wingman, e.g., (? (== $role WINGMAN)).

4.3 DISTRIBUTED IMPLICIT CONTROL AND COORDINATION

Implicit coordination is modelled by examining changes in the world model without relevant communication between the team members. Typically, implicit coordination involves flying a manoeuvre until some geometric observation is made, such as reaching a predefined range to a target.

The initial stage of the pincer intercept uses implicit coordination. The leader and wingman begin by flying different manoeuvres. The leader flies an intercept from the left, e.g., (int-left $target), while the wingman flies an intercept from the right, e.g., (int-right $target). Both agents maintain this manoeuvre until the range to the target is less than the maximum missile range, e.g., (< $range $max-missile-range). The arc between nodes P2 and P8 in Figure 1 corresponds to the action of the leader, and the arc between nodes P3 and P9 to that of the wingman.

At no time do the leader and wingman need to communicate overtly with each other to achieve the coordination. The coordination occurs because both agents are observing the world and know when they should change manoeuvres, e.g., they should change manoeuvre when the required range condition is achieved.

4.4 DISTRIBUTED EXPLICIT COORDINATION

Explicit coordination can be modelled by using communication messages. Such messages convey information about the state of the sub-teams or request information. However, an agent receiving a communication message with explicit coordination evaluates the effect of the information before commencing an action, unlike centralised control where agents blindly obey the command messages. Such messages correspond to different speech acts as described by Searl [Sea75].

The agents performing a pincer intercept use explicit coordination to determine when to fire a missile. The leader sends a message to the wingman indicating that the leader has fired a missile, e.g., (! (radio-message $team $self (missile-fired-at $target))). The wingman, on receiving the message from the leader, will record that the leader's missile was fired. The tactic that was waiting for this information, e.g., (^ (leader-missile-status $target FIRED)), will now evaluate if it should also fire a missile at the target. If the wingman decides to fire it issues a fire command, e.g., (! (fire-missile $target)). In this case the wingman was not ordered to fire, but was informed that a missile was fired and from the operational procedures knew that it should now determine if it should fire a misile.

57

5 CONCLUDING REMARKS

With an increasing number of expert systems being widely available, a typical industrial environment may include multiple expert systems that are required to cooperate and coordinate their activity. One can view such a group of systems as a team and use control and coordination mechanisms used by humans and adapted to the automated scenario. A similar problem is the problem of modelling the behaviour of groups of humans either in a simulation environment or for the purpose of management and coordination of the group (e.g., business process management).

One of the main problems of modelling the behaviour of groups of humans or coordinating the behavior of human or artificial agents is the problem of representing their behaviour as a team. This problem include the problems of representing the team and modelling the team behavior, and in particular the problem of modelling and using different types of control and coordination methods. This is further emphasized when the team is required to work with limited time and communication facilities.

One example of such a problem is the modelling of aircraft and air-combat pilots that are engaged in air missions where the pilots are required to reason and act in a highly dynamic environment. Furthermore, they are required to coordinate their activities and to improve their performance as a group. Communication facilities may not always be available either because of technical limitations or because of operational requirements.

To overcome such problems pilots use a wide range of team structures and coordination methods. Different role assignments allow the pilots to pre-determine the behaviour of the team members that is specified in various team tactics used in varying situations.

In this paper we have described how teams are modelled and how this information is used during tactics selection. We have also shown how different control and coordination mechanisms are implemented as plans in the dMARS system. This model together with the PACAUS system is combined to form the SWARMM system.

As part of this work we have identified two key features that dominate the ability of a group of agents (human or artificial) to coordinate their activity. These are the ability to observe the activities of other agents and use pre-defined methods for coordination, and the ability to use explicit communication. Communication can be used either for a centralized control structure or a distributed one but as it is a limited resource it should not be used frequently.

It seems then that the tactics employed by air-combat pilots to control and coordinate their activities is typically a combination of different types of control and coordination methods. The SWARMM system is currently in its final stages of development and will be commissioned at the Air Operations Devision in the coming year for further investigation of whole air-mission scenarios for the Royal Australian Air Force.

References

[ASH+94] D. Appla, M. Selvestral, C. Hienze, D. McIlroy, G. Tidhar, and A. S. Rao. Smart whole air mission model (SWARMM): Technical description. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1994.

[Aus94] Australian Artificial Intelligence Institute, Melbourne, Australia. *Distributed Multi-Agent Reasoning System (dMARS): Technical Description*, 1994.

[CL91] P. Cohen and H. Levesque. Teamwork. Technical Report 504, SRI International, Menlo Park, CA, USA, 1991.

[Fox81] Mark S. Fox. An organizational view of distributed systems. *IEEE Transactions on Systems, Man, and Cybernetics*, 11(1):70–80, January 1981. (Also published in *Readings in Distributed Artificial Intelligence*, Alan H. Bond and Les Gasser, editors, pages 140–150, Morgan Kaufmann, San Mateo, CA, USA, 1988.).

[Fox93] Mark S. Fox. Enterprise integration. *Personal communication*, February 1993.

[GK93] B. Grosz and S. Kraus. Collaborative plans for group activities. In *Proceedings of the 13 th International Joint Conference on Artificial Intelligence*, pages 367–373, Chambry, France, 1993. Morgan Kaufmann Publishers.

[GL86] M. P. Georgeff and A. L. Lansky. Procedural knowledge. In *Proceedings of the IEEE Special Issue on Knowledge Representation*, volume 74, pages 1383–1398, 1986.

[How91] M. Howlett. A description of the criteria used and the process of gameplan selection for air to air intercepts. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, 1991.

[Jen93] N. R. Jennings. *Cooperation in Industrial Multi-Agent Systems*. World Scientific Publishing Company, River Edge NJ, USA, 1993.

[JM92]   N. R. Jennings and E. H. Mamdani. Using joint responsibility to coordinate collaborative problem solving in dynamic environments. In *Proceedings of the Tenth National Conference on Artificial Intelligence*, pages 269–275, San Jose, CA, USA, 1992.

[KLR+92] David Kinny, Magnus Ljungberg, Anand Rao, Elizabeth Sonenberg, Gil Tidhar, and Eric Werner. Planned team activity. In *Proceedings of the Fourth European Workshop on Modelling Autonomous Agents in a Multi-Agent World, MAAMAW '92*, Viterbo, Italy, 1992.

[LJN94]   J. E. Laird, R. E. Jones, and P. E. Nielsen. Coordinated behavior of computer generated forces in TacAir-Soar. In *Proceedings of the Fourth Conference on Computer Generated Forces Behavioral Representation*, Orlando, FLA, USA, May 1994.

[LNR87]   J. E. Laird, A. Newell, and P. S. Rosenbloom. Soar: An architecture for general intelligence. *Artificial Intelligence*, 33(3), 1987.

[Mus93]   R. Muscatt. Combat display program - specification and user guide. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1993.

[RMSM92] A. Rao, D. Morley, M. Selvestrel, and G. Murray. Representation, selection, and execution of team tactics in air combat modelling. In Anthony Adams and Leon Sterling, editors, *Proceedings of the 5th Australian Joint Conference on Artificial Intelligence*, pages 185–190. World Scientific, November 1992.

[Sea75]   J. R. Searl. *A Taxonomy of Illocutionary Acts*, volume 7 of *Minnesota Studies in the Philosophy of Science*. University of Minnesota Press, Minneapolis, 1975.

[Sha85]   R. L. Shaw. *Fighter Combat Tactics and Maneuvering*. US Naval Institute Press, 6th edition, 1985.

[Ste93]   S. Steuart. PACAUS - overview and source code description. Personal Communication, Air Operations Division, Aeronautical Research Laboratory, Melbourne, Australia, 1993.

[Tid93]   G. Tidhar. Team oriented programming: Preliminary report. Technical Report 41, Australian Artificial Intelligence Institute, Melbourne, Australia, 1993.

[Wer90]   E. Werner. Cooperating agents: A unified theory of communication and social structure. In L. Gasser and M. N. Huhns, editors, *Distributed Artificial Intelligence: Volume II*. Morgan Kaufmann Publishers, San Mateo, CA, USA, 1990.

A KNOWLEDGE-BASED STRATEGY FOR THE RE-ASSOCIATION OF FRAGMENTED SENSOR REPORTS

Keith P. Mason

Electronics and Surveillance Research Laboratory
Defence Science and Technology Organisation
P.O. Box 1500
Salisbury SA 5108
AUSTRALIA

ABSTRACT

This paper discusses a knowledge based strategy to assist in the task of identifying and tracking radar emitters from radar signals intercepted by a passive sensor system. The novelty of this strategy is that it utilises knowledge of the performance (and in particular the limitations) of the sensor processing algorithms as well as knowledge of environmental data (such as from geographic information systems) and knowledge of radar emitter characteristics to re-associate reports of radar emitters which have been fragmented by the sensor system. An illustration of this strategy from a laboratory software prototype is provided in this paper.

1   INTRODUCTION

The passive detection of radar signals and the identification and tracking of the platforms that carry them is an important source of surveillance information. Radars operate in one of two modes, either pulsed or continuous wave, and it is the processing of pulsed signal waveforms that are discussed in this paper.

Sensors deployed for detecting radar emitters commonly report pulsed signal intercepts containing parameters representing the time of arrival (toa), direction of arrival (doa) and signal parameters including radio frequency (rf) and pulse width (pw). Analysis of the pulses from a single emitter will allow other parameters to be derived, such as pulse repetition interval (pri), measures of agility of pri and rf (that is, whether pri or rf are constant or change over time), and the emitter's scan pattern. The values of an emitter's parameters can be used as a classifier of the type of emitting radar.

The signals intercepted by the sensor system's antenna will often comprise a superposition of emissions from several radars active in the environment. These signals will be interleaved in time of arrival order. For measurements of the derived parameters of the individual radars these signals need to be de-interleaved into streams containing pulses only belonging to a single radar emitter. Many algorithms for de-interleaving have been proposed (a selection of de-interleaving techniques are described in [4]), however none so far has been without weaknesses. The weaknesses can lead to the fragmentation of the pulses belonging to a single emitter into multiple reported pulse trains, or lead to the false merging of pulses from distinct emitters into a single reported pulse train. In this paper one particular de-interleaving algorithm known as the sequence search [8] is used for illustration.

This paper presents a strategy that uses knowledge of weakness of sensor processing algorithms to assist in emitter identification. A laboratory software prototype is being developed to evaluate the benefits of this strategy for future operational systems.
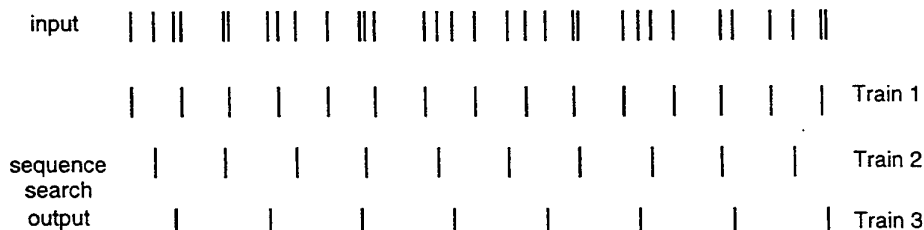

FIGURE 1 Sequence Search Applied to an Interleaved Sequence.

More information about radar systems can be found in [7], and on using radar intercepts for surveillance in [4]. Real time signal interpretation is discussed, for example, in [3]. Other knowledge-based approaches to the radar emitter domain can be found in [1], [2], [5], [6] and [9].

The next section details the problem of fragmentation of sensor reports of radar emitters. Section 3 describes the knowledge-based strategy used to repair fragmentation errors. Section 4 provides some examples illustrating this strategy. Section 5 provides a conclusion to this paper.

2   FRAGMENTATION

The sequence search tries to establish pulse trains for which pulses fall within narrow windows around a constant pri pulse train template. This works well when applied to constant pri pulse trains. However, when applied to a pulse train from an emitter with agile pri, the sequence search will find components of the pulse train that match the template, and report each separately, fragmenting the sequence of pulses from the emitter into several pulse train reports. This is illustrated below.

The sequence search starts by choosing two pulses separated by at least a set small time interval. It tries to use these two pulses to establish a constant pri pulse train. If a pulse train (with length greater than a threshold of say 8 pulses) is found, it is removed from the input data and the process is repeated. If a pulse train is not found then a different pair of pulses in the interleaved sequence is chosen (according to a strategy that gradually increases the time between the selected pulses) and the process is repeated. Figure 1 shows an input pulse train consisting of three interleaved constant pri pulse trains and each of the component pulse trains found by the sequence search. More details of the sequence search algorithm can be obtained from [8].

When faced with an emitter whose pri slowly slides linearly from one value (say t) to another, the sequence search will find pulses at 2t, 3t, 4t etc from the initial pulse as the small amount of slide is within the tolerance range of the search. However at some point the accumulated slide will be out of the tolerance range of the

search, so terminating the search and reporting the pulses already found as a pulse train. The process will then start again, and will find another pulse train with pri slightly different from the previous train. This pattern will continue until the end of the slide, where the pri jumps back to the start again and the whole pattern begins over again. This is shown in Figure 2. Thus the sequence search fragments the pulse train into a sequence of short pulse trains, rather than giving the desired single report containing all the pulses of the emitter.

There are a variety of emitter behaviours that include agile pri. These all result in fragmented reports from de-interleaving techniques such as the sequence search. As well as the sliding pri case described above, dwell and switch emitters, staggered pri emitters and interrupted pulse sequence emitters are described below and in the discussion of the re-association strategy in section 3.

Dwell and switch emitters are characterised by dwells of a number (say 80) pulses that have a constant pri and then switch to another pri for another dwell of (say 70) pulses, and so on, through a sequence of dwells. A sequence search de-interleaver will fragment the pulse train into separate reports for each dwell.

An interrupted pulse sequence emitter consists of a series of constant pri pulse trains, each with the same pri, separated by a gap larger than the pri. A sequence search de-interleaver will fragment the emitter into a series of reports, one for each component pulse train.

A staggered pri emitter continually cycles through a set of pri values (called stagger intervals). The number of stagger intervals is typically from 2 to several hundred. The cycle time of the stagger is called the frame interval. A staggered pulse train can be also be viewed as a number of time interleaved pulse trains with the same pri (equal to the frame interval) but off-set in start time. The sequence search is likely to find one pulse train for each stagger interval with pri equal to the frame interval but with off-set start times (that is, the sequence search de-interleaves the constant pri pulse train components of the stagger).

Since the sequence search uses a pri increasing from a small value, when processing a pulse train from a staggered pri emitter, a lot of potential pulse trains will

FIGURE 2 Sequence Search Applied to a Sliding PRI Emitter.

have been tested before searching for trains with the pri of the frame interval. It may be that some of these searches are able to match some pulses of the stagger as a constant pri train because the matching process includes a tolerance. This adds to the complexity of the fragmentation, as shown in Figure 3 where the first train found by the sequence search has a pri of one third of the frame interval.

It is clear from these examples that fragmentation of sensor reports complicates the process of postulating emitter entities from pulse trains reported by sequence search style de-interleavers.

3 STRATEGY

The approach of this paper to the re-association of pulse train fragments uses a generate and test paradigm. A library of emitter characteristics is used to generate candidates for each pulse train. Each candidate is then evaluated by comparing the expected fragmentation pattern for it with the reports from the de-interleaver. Knowledge of fragmentation patterns used in the evaluation process can be obtained analytically or empirically. The evaluation process also considers operational constraints (such as geographic location) on the candidates. The evaluations are then ranked to produce an interpretation of which emitters are active in the environment. The ranking process is weighted to minimize the number of emitters in the preferred interpretation.

3.1 Candidate Generation

The library of emitter characteristics is of central importance to the generation process of emitter candidates. The library function must be able to provide a list of emitter candidates that could produce a pulse train with the measured primary parameters of rf and pw, and the derived parameter of pri. The library must also be able to provide information about the fragmentation pattern that would be expected to be produced by the de-interleaver for each emitter candidate.

For example, a library search that returns a dwell and switch emitter also needs to return information on the emitter's other dwells, including sequencing. A search that returns a staggered emitter will need to return all of the stagger intervals. Interrupted pulse sequence emitter candidates should be reported with information on the duration of the interruption, and the number of pulses per fragment. A sliding pri emitter candidate requires characteristics of the pri variation to be returned. An emitter that only emits constant pri pulse trains needs to be identified as such by the library function.

3.2 Candidate Evaluation

An evaluation is performed for each candidate retrieved by the library search. The evaluation process will depend on the nature of each candidate and will measure the extent that the fragmentation pattern expected for a candidate is observed in the output from the de-interleaver. A library of emitter characteristics is also consulted to determine whether a candidate has any operational constraints, such as being at a certain geographic location, and these constraints are also considered as part of the evaluation.

Candidate evaluation yields a metric consisting of class and number, and also a list of pulse trains that supports the candidate. The number is a measure of the supporting evidence for the particular candidate class. For example, *dwell(9)* indicates that 9 subsequent fragments have been found to support the dwell and switch candidate, and *stagger(82%)* indicates that fragments covering 82% of the stagger intervals have been reported. The evaluation number is set to zero if an operational criterion associated with a particular candidate is not met.

Since constant pri emitter candidates exactly match the template of the sequence search they are not expected to be fragmented by a sequence search (excluding long term affects such as scan pattern), and so there is nothing to seek amongst a de-interleaver's output to support such a candidate. The default evaluation of a constant pri emitter therefore is *simple(1)*.

However, due to the possibility that a fragment pulse train from a complex emitter could be recognised as a

FIGURE 3 Sequence Search Applied to a Staggered PRI Emitter.

(spurious) constant pri emitter, it is prudent to look further to minimize the number of falsely postulated emitter entities. For example, if an emitter has previously been established with the same parameters (such as from an earlier scan), then this is additional evidence that supports a constant pri pulse train truly characterising an emitter (rather than being a spurious fragment), and so the evaluation in this case is *simple(2)*. Otherwise if there are multiple reports from the de-interleaver on the same bearing then it may be the case that fragmentation has occurred, and the candidate is ranked as *simple(0.5)*, indicating that care should be taken to check fragmentation possibilities before postulating a constant pri emitter on the basis of this pulse train report.

3.3   Candidate Ranking and Emitter Track Initiation

The aim of candidate ranking is to choose between multiple interpretations of the signal environment. That is, to choose between the various ways the pulse train reports from the de-interleaver could be associated into groups that corresponded to emitter entities. It is the approach of the strategy of this paper to attempt to rank candidates so that the smallest number of emitters is postulated. For example, a set of pulse trains might be interpreted as being six dwells from a dwell and switch emitter, or six emitters of constant pri and short duration. This ranking strategy prefers to postulate a dwell and switch emitter entity with six pulse trains assigned to it, rather than to postulate six emitters each with one pulse train assigned.

The ranking strategy eliminates from further consideration any candidate with a zero evaluation number and, except for simple candidates, any candidate which does not reach a (heuristic) threshold for sufficient evidence to establish the candidate class. For example, a dwell and switch emitter candidate has a threshold of 3 dwell fragments, a stagger emitter candidate has a threshold of 70% of total intervals observed, interrupted pulse train candidates and sliding pri candidates have a threshold of 4 pulse train fragments.

As candidates which have achieved their threshold are supported by several pulse train fragments, a potential for ambiguity exists if one or more pulse trains provide support for more than one of these candidates. However constraints (such as dwell and switch fragments being contiguous) restrict the cases where cross class ambiguities can believably exist. One case of ambiguity, for example, is where fragments from a stagger happen to have the same characteristics as an interrupted pri emitter. Such ambiguities can be resolved by a fixed order ranking strategy of processing dwell and switch candidates, then stagger, slide, interrupted, and lastly simple candidates.

Ambiguities of pulse train association may remain within a class, but these can be resolved by the amount of evidence supporting each candidate. The candidate with the largest evaluation number in the class is chosen first and an emitter entity is created for it and its supporting trains. The process is repeated with the remaining fragments and their candidates. It is unlikely the evaluation numbers would be equal for candidates referring to different but overlapping associations of pulse trains if the library is complete. Thus this process assigns (without ambiguity) the pulse trains to an emitter entity (the identity of which may be ambiguous).

When the above process comes to consider pulse trains which have only simple candidates, emitter entities should be created for those which have evaluation numbers of 1 or 2. However for those with an evaluation of 0.5 there is the possibility that they are spurious fragments as there are other emitters on the same bearing and it is prudent to attempt to associate them with established complex emitters. The analysis of fragmentation can be quite complex and the details of this association is beyond the scope of this paper.

The main requirement for the above strategy to be successful is that the de-interleaver reports for fragmented pulse trains will have enough components for the evaluation function to return values above the thresholds for the emitter types active in the environment.

4   ILLUSTRATION

A laboratory sequence search de-interleaver yields reports of the format:

```
train(index,rf,pw,min_pri,av_pri,max_pri,
             doa,start_toa,stop_toa).
```

The following is an example set of eight pulse train reports from this de-interleaver:

```
train(1,9000,2.0,1330,1330,1330,53,445462,499992),
train(2,9000,2.0,1130,1130,1130,53,364832,442802),
train(3,9000,2.0,1040,1040,1040,53,291942,363702),
train(4,9000,2.0,970,970,970,53,223972,290902),
train(5,9000,2.0,1230,1230,1230,53,138132,223002),
train(6,9000,2.0,1490,1490,1490,53,34092,136902),
train(7,9000,2.0,1810,1810,1810,53,22,32602), and
train(8,9000,2.0,9898,9898,9898,53,1800,40000).
```

Using a library of emitter characteristics, a laboratory prototype of the strategy described in this paper generates candidates of the form:

cand(index,train_index,emitter,type,constraints_list). The following candidates could be generated for the above pulse train reports:

```
cand(1,1,ajax_c3.2,dwell,
             [1630,1810,1490,1230,970,1040,1130,1330]),
cand(2,1,acme_123,simple,[]);
cand(3,2,see_far,interrupted,[4000]).
cand(4,2,ajax_c3.2,dwell,
             [1330,1630,1810,1490,1230,970,1040,1130]),
cand(5,2,vigilant_v23,simple,[]),
cand(6,3,vigilant_v1225,simple,[]),
cand(7,3,ajax_c3.2,dwell,
             [1130,1330,1630,1810,1490,1230,970,1040]),
cand(8,4,ajax_c3.2,dwell,
             [1040,1130,1330,1630,1810,1490,1230,970]),
cand(9,4,ajax_c1.3,dwell,[1810,1490,1230,970]),
cand(10,5,ajax_c3.2,dwell,
             [970,1040,1130,1330,1630,1810,1490,1230]),
cand(11,5,ajax_c1.3,dwell,[970,1810,1490,1230]),
```

```
cand(12,5,vigilant_v23,simple,[]),
cand(13,6,ajax_c3.2,dwell,
         [1230,970,1040,1130,1330,1630,1810,1490]),
cand(14,6,ajax_c1.3,dwell,[1230,970,1810,1490]),
cand(15,6,acme_123,simple,[]),
cand(16,7,ajax_c3.2,dwell,
         [1490,1230,970,1040,1130,1330,1630,1810]),
cand(17,7,ajax_c1.3,dwell,[1490,1230,970,1810]),
cand(18,7,acme_123,simple,[]), and
cand(19,8,long_sight,simple,[]).
```

These candidates show, for example, that pulse train 6 (which has a reported pri of 1490) has 3 candidates: (candidate 13) ajax_c3.2 which is a dwell and switch emitter with 8 dwells of pri (in sequence) of 1230, 970, 1040, 1130, 1330, 1630, 1810 and 1490, (candidate 14) ajax_c1.3 which is a dwell and switch emitter with 4 dwells of pri (in sequence) of 1230, 970, 1810 and 1490, and (candidate 15) acme_123 which is a constant pri emitter.

Each candidate is evaluated by looking for the expected fragments for the candidate type being reported by the de-interleaver. Evaluations are structured as eval(candidate_index,score,[train_list]) where train_list is are the pulse trains that support the candidate. Evaluation of the above yields:

```
eval(1,dwell(0),[1]),
eval(2,simple(0.5),[3]),
eval(3,interrupted(0),[2]),
eval(4,dwell(1),[1,2]),
eval(5,simple(0.5),[2]),
eval(6,simple(0.5),[3]),
eval(7,dwell(2),[1,2,3]),
eval(8,dwell(3),[1,2,3,4]),
eval(9,dwell(0),[4]),
eval(10,dwell(4),[1,2,3,4,5]),
eval(11,dwell(1),[4,5]),
eval(12,simple(0.5),[5]),
eval(13,dwell(5),[1,2,3,4,5,6]),
eval(14,dwell(2),[4,5,6]),
eval(15,simple(0.5),[6]),
eval(16,dwell(6),[1,2,3,4,5,6,7]),
eval(17,dwell(3),[8,5,6,7]),
eval(18,simple(0.5),[7]), and
eval(19,simple(0.5),[8]).
```

These show, for example, that for pulse train 6, candidate ajax_c3.2 evaluated as dwell (5) and is supported by trains 1, 2, 3, 4, 5 and 6; candidate ajax_c1.3 evaluated as dwell(2) and is supported by trains 4, 5 and 6; candidate acme_123 evaluated as simple(0.5) and is supported by train 6.

Ranking is used to choose between emitter candidates, and so to postulate emitter entities. The ranking process starts with dwell and switch candidates, and chooses candidate 16 as it has the highest evaluation number, dwell(6). An emitter entity is created for this candidate, and assigns its 7 supporting pulse trains. This leaves only one pulse train to consider. It has only one candidate with an evaluation of simple(0.5). As there is no possible fragmentation of the previously postulated emitter that could account for this pulse train, a second emitter entity is created. Emitter entities in the prototype have the form emitter(index,identity,type,rank,train_list), and the preferred interpretation of emitters active in the environment is:

emitter(1,ajax_c3.2,dwell,6,[1,2,3,4,5,6,7]), and
emitter(2,long_sight,simple,0.5,[8]).

5 CONCLUSION

This paper has presented a knowledge-based strategy to re-associate fragmented emitter reports from a de-interleaver based on the sequence search. This strategy has been demonstrated for several emitter types. The method could be extended to other emitter types and to other de-interleaving techniques.

ACKNOWLEDGEMENTS

REFERENCES

[1]   Mason, K.P. (1992). *Information Fusion in a Knowledge-Based Classification and Tracking System.* Industrial and Engineering Applications of Artificial Intelligence and Expert Systems. Lecture Notes in Artificial Intelligence, volume 604, pages 666-675. Springer Verlag.

[2]   Mason, K.P. and Hood, S.T. (1989). *KBESM - A Prolog Blackboard System.* Australian Computer Journal, volume 21, number 3, pages 100-107.

[3]   Hood, S.T. and Mason, K.P. (1987). *Knowledge-based Systems for Real-time Applications.* In Applications of Expert Systems, R. Quinlan (ed), pages 102-119, Addision Wesley.

[4]   Wiley, R.G. (1993). *Electronic Intelligence: The Analysis of Radar Signals.* 2nd Edition. Artech House.

[5]   Byrne, C.D., Miles, J.A.H. and Lakin, W.L. (1989). *Towards knowledge-based Naval Command Systems.* 3rd IEE International Conference on Command, Control, Communication and Management Information Systems, pages 33-42.

[6]   Roe, J., Cussons, S. and Faltham, A. (1990). *Knowledge-based Signal Processing for Radar ESM Systems.* IEE Proceedings, Vol. 137, Part F, No. 5, pages 293-301.

[7]   Skolnik, M.I., (1981). *Introduction to Radar Systems.* 2nd edition McGraw Hill.

[8]   Mardia, H.K. (1989). *New Techniques for the De-interleaving of Repetitive Sequences.* IEE Procedings, Part F, Communications Radar Signal Processing, vol136(4), pages 149-154.

[9]   Marshall, C.B. and Simons, P.R. (1990). The Application of Real-time AI Techniques to Pulse Train De-interleaving. Military Microwaves '90. Conference proceedings, London, July 11-13,1990.

The Development of An Intelligent Radar Tasking System

Tristan Chiu

BHP Engineering,

PO Box 1237, North Sydney, NSW 2059

Dr. Douglas Kewley, Christopher Crouch

High Frequency Radar Division,

Defence Science and Technology Organisation,

PO Box 1500, Salisbury, SA 5108

Laurie Lock Lee

BHP Information Technology,

PO Box 216, Hamilton, NSW 2303

ABSTRACT

After years of research effort, the Australian Government is now committed to build the Jindalee Over-The-Horizon Radar Network (JORN), which will enable a comprehensive and cost effective surveillance of the northern and western approaches of Australia.

In order to realise the full potential of the technology embodied within the radar system, the Department of Defence is keen to incorporate the experience and higher level knowledge of the research scientists of the Defence Science and Technology Organisation (DSTO) into the control of JORN.

A team of BHP knowledge engineers has worked closely with DSTO scientists and software engineers. A software application has been developed to capture and disseminate the radar tasking knowledge from the scientists. This application has demonstrated that the technology, if successfully deployed, could have a major impact on the efficacy of JORN facilities.

1. Background

Australia is at present undertaking the Jindalee Over-The-Horizon Radar network (JORN) Project, which will install a network of radar transmitters and receivers to provide a comprehensive and cost effective surveillance of the northern and western approaches of Australia. The ability to illuminate and identify a desired target is very much dependent on variable refractive characteristics of the ionospheric layers. Coupled with the need to track multiple aircraft and ship targets against unwanted reflections from waves and ground objects, this makes the operation of the radar network extremely complex.

Being a highly complex system, the Department of Defence is concerned about the ability of radar controllers (RADCON) to operate the system relative to its full capability. The only opportunity available for obtaining sufficient skills, before the completion of JORN, is through limited operation of the existing experimental radar at Alice Springs. Due to the Department's policy of rotating site personnel on a frequent basis, most of the RADCON on site can be better equipped by having the guidance of an experienced operator. In considering the objectives of JORN, this experience is important to the security and safety of the nation.

DSTO High Frequency Radar Division (HFRD) is the custodian of the OTHR technology in Australia. Over the past 30 years, HFRD scientists have built up a wealth of knowledge in interpreting ionospheric conditions and configurating radars for surveillance tasks. In particular, their expertise in tuning radar parameters to suit changing environmental conditions and surveillance requests is recognised worldwide. It is this expert knowledge that the Department is keen to retain for the benefits of the RADCON.

The Department has targeted the capturing and dissemination of radar tasking knowledge as the pilot study, and planned to develop the Intelligent Radar Tasking System (IRTS) in several stages. Following the Feasibility Study completed by BHP Engineering (BHPE) in 1992, BHPE was again awarded the contract in 1993 to develop a Concept Evaluator.

The objective of this recently completed stage of work was to develop an off-line decision support system for managing the radar tasking. By formally documenting perceived best practice, the system also enables HFRD scientists to assess the technology and

refine their radar control experience. One of the key features of IRTS is to prompt the RADCON about the feasibility of radar tasks in a proactive manner. The success of the Concept Evaluator could lead to installing the system at the existing radar control centre at Alice Springs. The ultimate objective is to implement IRTS as part of the decision support enhancement for the JORN facilities.

2.   The Over-The-Horizon Radar Technology

The OTHR concept is simple. A radar signal is beamed skyward from a transmitter and then refracted by the ionosphere, located at about 100 - 300km above the earth, down to the surface. When this transmission signal strikes a solid object, it will be back scattered via a similar return path back to a receiver, located at some distance from the transmitting site. This propagation is relatively unaffected by the curvature of the earth's surface and is capable of propagating signals over great distances.

Although the performance of OTHR is subject to the ionospheric conditions, it is regarded as the most cost effective means of wide-area surveillance, e.g. it can cover an enormous area by comparison with normal line-of-sight radar, and is estimated to be about one tenth the cost of a space orbiting surveillance scheme. Apart from defence applications, its detection capability is able to assist Customs and Immigration officers, provide valuable weather information, and enhance search and rescue operations.

The ionosphere consists of several layers, whose height change from day to day and over time. One of the essential requirements for effective radar tasking is the selection of appropriate frequencies for different ranges of target illumination areas, so that there is a high degree of confidence in the range and height of the tracking objects. Target types, sizes, approach velocities and "clutter" (reflections from fixed objects) require other strategies in radar management to provide accuracy in identification.

The construction of JORN facilities is the culmination of 30 years of research and development by DSTO scientists to develop Australia's own Over-The-Horizon radar technology.

3.   The Knowledge of Radar Tasking

The knowledge required for radar tasking involves three main steps, namely:

- defining the surveillance task;
- choosing radar parameters for the task; and
- making sure that the radar schedule can be performed.

The thread of the knowledge processing is via the performance and timeline requirements of a task. During the tasking process, values of task predicted performance and timeline are calculated. These values are then compared with the requirement thresholds derived during the task definition stage.

The requirement threshold of a task is described as a generic definition of the minimum acceptable target detection and tracking capability with consideration of its task and target specification. This requirement is treated as the threshold, such that a performance worse than this value implies that the viability of performing the task could be compromised.

Further breakdowns of essential HFRD's expert knowledge for radar tasking are described as follows:

a.   Defining a task

A task, with minimum set of parameter values, is raised according to requirements specified by the Request Agency. This includes the translation of Request Agency's specification to definition of surveillance area and classification of the task. Request Agency is responsible for planning surveillance tasks within the Department of Defence.

b.   Assigning appropriate parameters for a task

After a task has been defined, initial values for the full task parameter set will be assigned. The initial values are based on the classification of the task and the target. Operational experience is also applied in determining these values.

c.   Minimum requirement of a task

Expert knowledge is required to specify the minimum requirement of key monitoring parameters. These parameters are used for measuring the performance of a task. Values of minimum requirements depend on target type, task type, previous operational records and expert's knowledge in ionospheric physics.

d.   Interpret the propagation advice

After receiving the propagation advice from the existing HFRD software, expert knowledge is applied to adjust task parameters and surveillance area configurations. The purpose is for all tasks to achieve the 'minimally adequate' status, such that the corresponding minimum requirement for each task will be met while using minimum radar resources. This is to ensure that the task can be executed under the prevailing environmental conditions.

e.   Allocation of scheduler priority

This step is to allocate the scheduler priority for each of the tasks on the radar schedule. The determination of the scheduler priority is subject to the importance of the task, the predicted time required to dwell on the surveillance area, and other task parameters.

f.   Timeline Calculation

The timeline calculation enables the RADCON to investigate whether each scheduled task can meet its radar timeline requirement. If the requirement is not met, the calculation also indicates the severity of the problem, so that proper 'WHAT IF' analysis can be carried out to improve the situation.

g.   Overall Feasibility Assessment (OFA)

The Overall Feasibility Assessment is required if one of the following situations occurs:

• tasks are to be added to/deleted from the radar schedule:

• change of environmental conditions;

• change of other operational conditions, such as notification of poor tracking problems; or

• analyse 'WHAT IF' scenarios for existing tasks without degrading their viabilities.

OFA contains experts' knowledge in trading-off radar parameters and their effects.   Typical trade-off parameters include:

• task azimuthal extents;

• radar aperture;

• radar bandwidth; and

• task scan time.

The overall strategy of the trade-off process in OFA involves the consideration of:

• choosing tasks which are designated as lower priority;

• choosing tasks which have longer scan time, because potentially there are more scope to meet the timeline requirements during the trade-off processes; and

• choosing tasks which have higher performance indices, because potentially there are more scope to meet the radar performance requirements during the trade-off process.

This assessment introduces the requirement "deadbands" to maximise the number of tasks that the radar can perform at a given time. In order to accommodate more tasks during the trade-off process, the predicted performance of some tasks may be reduced.   If this situation arises, the experts' knowledge is required to investigate whether the timeline requirement can be accomplished by reducing the predicted performance of a task from "minimally adequate" to "marginally acceptable" within a predefined performance deadband.

If the trade-off approach is unable to resolve the timeline problems, then one or more task(s) may need to be removed from the radar schedule. The order of removing tasks is subject to the expert's interpretation of the priority and relative viability of each scheduled task.

h.   Proactive Task Monitoring

One of the key features to demonstrate the application of HFRD expert knowledge is the provision of advice to the RADCON in a proactive manner. In order to facilitate this feature, radar tasks can be stored in a "Proactive Task List" for monitoring purposes. These tasks are monitored in the following two aspects:

• feasibility under the prevailing ionospheric conditions; and

• feasibility of adding the task to the current radar schedule.

If the requirements of both aspects are met, then the RADCON will be alerted. This decision support feature helps the radar controller to:

• implement the most appropriate radar task with total confidence that the modified radar schedule can be executed without jeopardising the viability of other tasks and with minimum operator effort;

• initiate extra tasks to the radar schedule so that both the RADCON and Request Agency can understand the prevailing environmental conditions better; and

• make full use of the radar resource.

In this regard, knowledge from HFRD scientists is required to establish the criteria of the feasibility of a radar task.

i.   Knowledge not covered in this project

In order to maintain a manageable knowledge domain, the scope of the radar tasking knowledge covered in this project does not include the following external factors:

• temporal or seasonal influence;

• global optimisation of the radar operation;

• any specific information about locations for

surface surveillance;

- direct knowledge of sun spot activities; and
- direct knowledge of radar operational and hardware setup problems.

4.   The Intelligent Radar Tasking System

Over the past 15 months, a team of BHP knowledge engineers has worked closely with HFRD scientists and software engineers. During the project many knowledge acquisition workshops were conducted, each of them concentrating on one aspect of the radar tasking knowledge. The radar is sufficiently complex to result in the expertise having to be shared amongst multiple expert scientists. Knowledge sources were analysed from different angles and consolidated by means of iterative discussion and verification. The result was a summary of expert radar tasking knowledge that took in the consensus of many expert HFRD scientists.

There are five main features contained in the IRTS software, they are:

a.   Initial Viability Assessment

When a task is defined, either manually or via a number of automated processes, this assessment ensures the task can accomplish its designated performance requirements as well as to minimise the potential demands on radar timeline.

If the predicted performance of a task is below its required threshold value, then IRTS will initiate procedures for improving the viability of the task. This is achieved by:

- trade-off task parameters
- re-define task coverage extent

On the other hand, if the predicted performance is significantly above the requirement level, an investigation will be instigated to trade-off parameters to minimise the potential demands on the radar timeline.

During the assessment, advice from existing HFRD software will be sought to interpret the environmental conditions.

b.   Environmental Data Update

As the ionospheric conditions vary so does the feasibility of performing radar tasks. Periodic checking of new propagation advice is conducted to obtain latest environmental information for the IRTS assessment. Apart from obtaining performance prediction based on existing task parameters and

geographical configuration, IRTS also requests for the most appropriate task parameter values and coverage details under the prevailing environmental conditions.

c.   Overall Feasibility Assessment

This process is initiated to ensure that individual revisit requirements of each task will be met, while still achieving their sensitivity requirements. Features of OFA are described in the previous section.

If the result of OFA indicates that the timeline of the radar schedule cannot be met, the RADCON has the following three options:

- do nothing, submit all tasks to the radar schedule with the expectation that performance of some tasks may be compromised;
- delete task based on IRTS's advice; or
- re-assign a task to be a "burst mode task".

IRTS enables the RADCON to redefine a task as a "burst mode task", which temporarily pre-empts other tasks on the radar schedule, hence the total radar resource can be deployed solely on this task. After a pre-defined period of time, the task will become dormant for a relatively long time before the next burst of operation. This operation mode is particularly useful for ship tasks, that they are not likely to move outside of a coverage area over relatively long periods of time (e.g. tens of minutes), as the environmental conditions can change dramatically over such timescales.

The calculation of radar resource utilisation is continuously updated and displayed on the screen. This information provides a useful indication of the projected radar loading during the trade-off process.

Under the normal operational conditions, the OFA process is initiated manually by the RADCON. However, subject to the RADCON's discretion, IRTS allows the OFA to be executed automatically whenever a new set of environmental conditions leads to significant alteration of task parameters.

d.   Proactive Task Monitoring

As described previously, IRTS facilitates the proactive tasks monitoring as a decision support mechanism to the RADCON. This process is undertaken as a background investigation during the normal radar operation.

On the Proactive Task List display, IRTS utilises icons and colour schemes to indicate to the RADCON the likely performance of proactive tasks under the prevailing environmental conditions and radar

schedule.

The other feature of providing decision support to the user is the production of reports, generated automatically on a fixed time interval. These reports help the Request Agency to understand the current status of environmental conditions and radar capabilities. The information is particularly useful for reviewing existing surveillance tasks and planning new missions.

e.   Effect of Radar and Task Restrictions

In the real radar operational environment, it is conceivable that not all features of the OTH radar are available at all time. To mimic this scenario, there are in-built facilities in IRTS to impose restrictions on individual radar or task parameters. Each of such restrictions will disable different part of the IRTS software, some only restrict a small part of the system capabilities, others may inhibit the entire trade-off process.

The IRTS software is designed using object oriented methods, this approach has the advantage over functional methods, in that each object in the design can be mapped directly to the actual physical items. Although the structure of the radar tasking expertise is characterised in an object oriented model, the behaviour of these objects is described in procedural form. This form of knowledge representation is deemed to be most appropriate in eliciting knowledge from HFRD scientists, their "theoretical" experience is different in comparison with the knowledge captured from the usual expert "practitioner". The IRTS software contains more than 100,000 line of codes.

Graphical user interfaces are regarded to be critical for gaining the acceptance of RADCON and HFRD scientists. In this regard, an integrated graphical development environment called 'G2' was used in this project. 'G2' is an integrated real-time expert system development tool, with inbuilt graphic-oriented environment for GUI applications. This software tool enables IRTS to conform with the screen display 'style guide' promoted by HFRD.

5.   Status of the IRTS Intelligence

Through the development of the IRTS Concept Evaluator, a methodology for identifying and adopting best perceived operating procedures in radar tasking has been established. In our view, when the system is fully reviewed and implemented at the existing Alice Springs radar site, it will be able to assist an inexperienced RADCON to attain the skill level of a "Competent Operator", as classified in the

following Table:

Skill Level	Key Features of Skill
New Operator	New starter without prior knowledge on radar control
Novice Operator	Inexperienced operator with limits exposure to OTHR operation.
Competent Operator	Controls and schedules OTHR, conforming with standard operating procedures, on his(her) own rights.
Experienced Operator	Complements standard operating procedures with own knowledge. Sometimes overrules procedures with previous experience.
Expert Operator	Has high degree of understanding on aims of radar tasks, ionospheric physics and common knowledge of OTHR. Can perform prediction based on prevailing environmental conditions and tasking requests.

Table 1: Classification of RADCON Skill Levels

Future enhancements to IRTS, especially with experience gained through using the standard operating procedures, should enable the software to accomplish the skill level of an "Experienced Operator".

The skill level of an "Expert Operator" (e.g. The OTHR expertise resident in the minds of HFRD scientists) can only be achieved via extensive investment in time and effort. This investment is considered to be too costly, in terms of both economic and technical viabilities, to be incorporated as artificial intelligence into the computer system.

6.   Concluding Remarks

The IRTS Concept Evaluator, which was completed on schedule, has demonstrated that the capability of the existing experimental radar at Alice Springs could be significantly extended by the intelligent management of the resources available. If successfully deployed, the technology could also have a major impact on the efficacy of OTHR facilities.

7.   Acknowledgments

Case-Based Decision Making for Intelligent Decision Support

Frada V. Burstein and Helen G. Smith
Department of Information Systems
Monash University, Melbourne, Australia
Frada.Burstein@monash.edu.au

Abstract

In this paper we discuss the application of case-based reasoning to intelligent decision making support. The cases recording past decisions become a form of organisational memory to support future decision making. If the knowledge is represented in cases then there is a need for techniques to retrieve the relevant cases. We propose to use multicriteria decision making theory here if an appropriate expression of evaluation criteria for cases is found. The development of an appropriate multicriteria expression would thus provide the adaptation of the existing knowledge to the specific decision making situation. Decision support systems which have a 'memory' of cases and can effectively retrieve and adapt past experience to support the current problem would then be considered to be intelligent.

1. Introduction

The concept of an Intelligent Decision Support System (IDSS) has recently gained popularity. It is viewed as an extension of decision support systems using some sort of knowledge base as a source of information for the decision maker. We have studied the nature of such systems and discussed the role of the knowledge-based component in IDSS [1]. Our conclusion was that knowledge bases built for decision support purposes are quite distinct from knowledge bases developed for traditional problem-specific expert systems, and these differences must be reflected in the method of knowledge based system construction and its content.

In this paper we investigate the role and place of cases as a technique for representing knowledge in an IDSS. The advantage of using a case base is that it is dynamic as

opposed to conventional knowledge bases which typically provide quite static knowledge structures [2]. We continue this approach by investigating techniques for comparing cases and assessing their similarity. In particular we focus on the use of multiple qualitative and quantitative criteria in the similarity metrics for case retrieval. We propose to use multicriteria evaluation of cases within IDSS framework as a mechanism to increase a flexibility of the system in providing user-specific and decision situation-specific decision support.

2. Intelligent Decision Support

Intelligent Decision Support Systems are an extension of decision support systems. In our view, a decision support system (DSS) is a system to support decision making in an unstructured or poorly structured situation. We consider DSS being a system that "involves the application of behavioural science and information technology to aid human judgment in important decision situations"[3]. In such an approach the stress is on the assistance of the human side of the decision process. The decision maker bears responsibility for the final decision, not the system itself.

In line with this approach, we view decision making as a human activity and a DSS as support for that activity rather than a replacement for it. This view then requires that any intelligent support for decision making does not supplant the human decision maker in the way that is assumed by most current expert systems.

In addition, the emphasis on "important decision situations" reflects a necessary concern with decisions that require more than just a representation of well-formulated rules as in Simon's "programmed decisions" [4]. These important decision situations are frequently "unstructured" or "poorly structured" in the sense that the facts and relationships pertinent to this situation (domain knowledge) are not readily available and the procedures for coming to a decision are poorly understood.

71

The following diagram in figure 1 proposes a view of IDSS in which DSS tools are extended by some form of knowledge base in order to achieve the "intelligence" required by the term. The user, faced with a particular task in a specific problem domain, has access to data bases, knowledge bases and a library of models to support decision making.
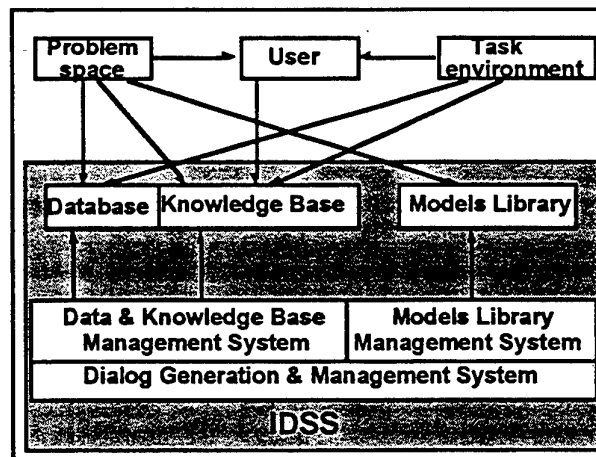
Fig. 1. Components of IDSS

We distinguish a problem space and a task environment similar to those in [5] approach. It assumes a problem space being a personal interpretation of the problem by the decision-maker, whereas a task environment is a general, physical and social environment in which the problem takes place. From this point of view a problem space, task environment and a user are sources of specific knowledge about the problem that can be represented and stored in both data and knowledge bases. However the model library contains general decision making approaches and technics that have been used in the problem domain . The user might not be aware of them, or needs assistance from the IDSS to use the most appropriate model stored in that library.

Our concern is with the knowledge base in particular and the form which that knowledge base might take. We have investigated the process of knowledge modelling for intelligent decision support [6] and concluded that within the IDSS framework this

process should be different from the one used for conventional knowledge-based (expert) systems construction.

An expert system is the most common form of a knowledge base. The term "expert system" being used here is representing a technology rather than to imply any specific modelling of an expert. For expert systems technology to be of use, the knowledge must be stable since considerable time is required to acquire the appropriate knowledge, build a system and validate that system before it can be used. Problem domain knowledge is rarely that stable, particularly for ill-structured decision making problems as a result of frequent change within the organisation.

The distinction made above between the problem domain knowledge and the procedures for making a decision can be reflected in the intelligent support provided. While both are important in decision making, general problem-solving intelligence is perhaps easier to make available, since many problem-solving techniques are well described and the appropriate use of these techniques is well understood. Knowledge about which decision-making model to apply may be considered to be sufficiently static to serve as the basis for an expert system. On the other hand, the provision of problem domain intelligence is difficult since decisions made in the real context are often required within very short time frames. As noted above, the classical model of knowledge acquisition for expert system development requires considerable time. In addition, the process of building the expert system has added structure to the problem domain and decision support is replaced by programmed decisions.

Nevertheless, the distinction between ill-structured and well-structured decision situations may be fuzzy. One manager may have significantly more experience and have developed better mental structures for the decision than another with less experience in that particular type of problem. In many organisations, the knowledge of one manager is lost to other decision makers as he or she moves to other roles within the organisation. Making the knowledge of one manager available to others within the organisation helps

to add structure to a decision making situation without necessarily removing the need for the manager to actually make the decision.

Adding intelligence requires a problem domain knowledge base which can support rapid retrieval of relevant knowledge which can then be applied within the time frames of real decision situations. A classical expert system is inappropriate in this situation.

The other form of knowledge that a decision maker can benefit from can be acquired from the previous similar decision situations (cases). Cases may be obtained from the experience of other decision makers. This kind of knowledge does not become obsolete with time, and can provide useful information for the future decisions in a similar circumstances. Case-based decision support system operates as an 'experienced' adviser to the decision-maker, but with the additional intention of recording new experiences (learning). Concept of learning in case-based decision support systems is different from a common understanding of it within AI community in a context of machine learning. Learning by induction needs a large collection of training data whereas case-based learning is triggered by each and every new case which is recognised as being significantly different in some sense and important example for a future use in decision support situation.

We consider such an approach perspective for intelligent decision support. We propose to use multicriteria decision making theory here if an appropriate expression of evaluation criteria for cases is found. The development of an appropriate multicriteria expression would thus provide the adaptation of the existing knowledge to the specific decision making situation. Decision support systems which can effectively retrieve and adapt past experience to support the current problem would then be considered to be intelligent.

We briefly describe the main issues of case-based reasoning and outline those that we consider beneficial to be incorporated into IDSS framework.

3. Case-Based Reasoning in Decision Support Context

In solving new problems, humans rely on past experience. Expert decision makers are those whose past experience is extensive and rich and who can make use of their past experience. By analogy, an IDSS should keep the past experience of decision makers and make it available for decision support in similar decision situations. Rather than trying to predict the future use of this experience, it would seem most suitable to record each instance (case) of a given type of decision then use case-based reasoning to model the process of applying similar past experiences to the current decision making situation.

Case-Based Reasoning (CBR) originated as a psychological theory of human memory organisation and of the cognitive processes of learning, planning and problem solving rather than as an artificial intelligence technique [7]. Since then, artificial intelligence researchers have applied CBR theory in systems which model the expert's problem solving ability. CBR offers a number of advantages over rule-based systems.

Firstly, in some situations cases represent the expert's knowledge more accurately. In fact, it has been found that experts usually tend to use knowledge in the form of particular episodes (cases) rather than as rules.

Secondly, knowledge in rule-based expert systems is limited to those rules that have been identified and stored. The process of maintaining a rule-based system is normally a manual one requiring further knowledge acquisition. The system is not capable of adapting its own knowledge to new situations whereas CBR has an adaptive mechanism built in.

Finally, current expert systems do not have memory, so they do not recognise problems that they have already solved or failed to solve. Without a memory, expert systems cannot accurately model a human expert's behaviour. The case-based approach provides a facility similar to the expert's memory.

Reasoning using a case base involves selecting cases that are relevant to a new problem if they are, in some defined way, "similar" to the new problem, ie.,

New Solution = Past Solution(s) from the Case Base + Measure of Similarity.

The choice of the measure of similarity, as well as the indexing rules, will influence the success of the new solution.

Learning in CBR system involves both successes and failures from the reasoning phase. When cases are selected successfully they are adapted and the new case indexed (using the current indexing rules) and stored. When a case fails to meet the user's requirements it is used to improve the indexing rules to avoid similar failures in future.

Case-based reasoning has certain assumptions behind it. These assumptions are relevant from both the psychological and computer application view points.

a)   A case is available for the situation at hand.

That means that a decision maker (or a case-based system) has a memory (storage of information) containing:

- past decision situations which include some relevant to the situation at hand,

- problem solving activities relating to those past decisions and

- the decision made, together with an evaluation of its effectiveness.

b)   The case is correctly indexed in memory so that it can be retrieved for the current situation.

This assumes the existence of an appropriate storage and retrieval mechanism. It should be mentioned that cases may relate to each other as well as to the new situation.

c)   The case is well understood and can be adapted for future use.

This assumption is based on the dynamic nature of memory. New cases similar to those already stored must be related to the existing ones, possibly altering the indexing of

all cases. In this way the contents of memory are continually evolving. In humans, this evolution provides the basis for learning. In a similar way, a dynamic case base enables the system to perform a limited form of learning specific to the problem domain it addresses. This can be contrasted with conventional knowledge bases where the knowledge is static [7].

d) Learning is triggered by failure.

Failures as well as successes can be useful aids to learning. For example, failures in a case-based IDSS may include:

- cases recording decisions which were not successful, enabling the decision-maker to anticipate and avoid problems [8];

- cases which are retrieved but are not relevant to the current situation, triggering reindexing of the case base, ie., adaptation of the system's memory.

Both instances are a form of learning. However, as it was mentioned before this type of learning is different from machine learning and is based on adaptation rather then 'pattern' recognition.

A case-based system generalises and integrates new facts into memory, giving the effect of learning. Classical artificial intelligence allows that a system which is capable of learning from experience is intelligent. Therefore a decision support system incorporating case-based learning can be called a 'true' intelligent decision support system.

4.   Comparison of Cases for Intelligent Decision Support

As mentioned above, in the context of decision support, stored cases play a role similar to the memory of an experienced decision maker. Generally, memory provides people with knowledge about the world which is important for understanding the world through reflection on past experiences relevant to the current situation. Memory is defined by the

objects it stores, the mechanism used to store them and the search procedure to rediscover them. Remembering involves retrieving the relevant information from memory. In a computerised case base within IDSS, remembering means retrieval of the case(s) that are relevant to a particular decision making situation.

Many authors have discussed the techniques for representing cases [7, 9, 10, 11, 12]. Each of these techniques has been developed to serve a particular application requirement. In the decision support context, case descriptions should include their input conditions (the tasks they relate to), goals they attempt to satisfy and the outcome (success or failure) in satisfying those goals. The indexing mechanisms are the most significant in identifying relevant situations and matching these to the views of the decision maker as closely as possible. The collection of characteristics used to describe cases might involve several quantitative as well as qualitative parameters. Thus the problem would be to find a common measurement on which to build similarity metrics applicable to both types of parameters. Multicriteria decision making provides a number of techniques for comparison of possible decisions based on non-homogeneous assessment characteristics. Multicriteria decision making techniques may be applicable as a mechanism for similarity assessment in the context of IDSS.

5.   Multicriteria Decision Making and Cases

Multicriteria decision making is a normative decision making approach to modelling the decision maker's preferences from among a wide collection of alternatives. In this approach a set of possible choices (eg. courses of action) is determined. This set is finite but tends to be rather too large for a decision maker to be able to find the 'best' decision without any support. Furthermore, each of the choices is characterised by a number of different attributes reflecting the assessment criteria of the decision maker. That is those attributes considered important by the decision maker in order to achieve the main goal of the decision making process. For most practical decision making problems several, often conflicting, assessment criteria need to be taken into account [13]. This set of criteria has

to be determined at the very beginning of the decision making problem formulation and the definition of essential criteria, their range and domain, and the appropriate assessment procedure for each of them is a non-trivial task. Once the values of each criterion have been assessed, a corresponding binary preference relation can be defined showing, for each pair of possible decisions, that one choice is better than the other according to this particular criterion. The set of the best (maximal, effective) decisions corresponding to this preference relation can then be determined.

The main task of a decision support procedure is to aggregate the information provided by the set of assessment criteria (and/or related binary preference relation) and derive a corresponding aggregated preference relation on the initial set of choices. The aggregated preference relation in this context is a set of ordered pairs of all decisions corresponding to the preferences expressed by the set of criteria. The decision support procedure in this approach generates the set of effective (satisfactory) decisions based on the information provided by the set of assessment criteria and their values with respect to the overall goal of the decision making process. This set usually is more manageable than the original set of all possible alternatives, so the decision maker can use it to make a final choice. The 'proper' decision support procedure comes up with the decision(s) that is (are) the best on at least one of the given criteria. Therefore the final choice would also be preferred by the overall multicriteria procedure. This is an important property that is useful for the case comparison and retrieval as well. So if the DSS has to come up with the cases relevant to some particular decision situation they need to be 'the closest' to it in at least one particular sense.

This generalisation can be extended even further. We can suppose that, as well as a set of possible decisions, a set (vector) of preference relations is defined to describe which decision is preferred according to a specific characteristic.

This can be represented by the following tuple:

$< X, R(X), F(R), X(R), X' >$, where

- $X=\{x, y, \dots, z\}$ is the set of possible decisions;
- $R(X) = \{R_1, \dots, R_N\}$ is the set of preference relations (vector preference relation);
- each R_m (m=1,...,N) is a binary preference relation corresponding to the m-th component of the given information (I_m) considered for the comparison of the decisions, ie.,

  $R_m = \{(x,y) \in X \times X : x$ is preferred to y according to $I_m(x)$ and $I_m(y)\}$ for all m=1, ..., N.

The function F(R) denotes the procedure of aggregation of the vector preference relation. As a result of this aggregation the combined preference relation can be built. This preference relation should reflect all the preferences provided by individual R_m (m=1, ..., N). In this case the set of maximal (effective) decisions X(R) can be used as a basis for the decision support procedure. The set X' denotes a sub-set of the set X(R) of effective decisions that would be produced by the decision support procedure and provided to the decision maker for final choice. It is clear that X(R) represents the maximal outcome from the DSS. It is advisable to be able to control the volume of X' according to the decision maker's requirements.

In our research of multicriteria decision making problems we build the aggregated preference relation based on the Pareto domination principle according to the two arguments function that expresses numerically the value of preference of one decision over another. This function is called the Superiority Degree [14, 15, 16]. Generally speaking, Superiority Degree (SD) is a function that satisfies a simple condition for any two alternative decisions:

$$f(x, y) = - f(y, x)$$ for any decision x and y from the set of all choices X.

This condition is usually referred to as an asymmetry condition.

The advantage of such a function is that it provides a general approach to the comparison of the decisions described by the set of different characteristics, no matter whether they have qualitative, quantitative or even fuzzy values [16]. Because the procedure of measurement is based on the overall utility of the decision for the general decision making goal, it provides a basis for comparison of any pair of possible decisions.

This makes it attractive in the context of any nature of the alternative decisions (in particular, the decisions can be represented by cases from a case base).

One of the suggested ways to calculate SD for multicriteria decision making problem is:

$$f(x,y) = \sum_{i=1}^{N} \mu_i(x,y) / \sum_{i=1}^{N} |\mu_i(x,y)|,$$

where

$$\mu_i(x,y) = \begin{cases} 1, & \text{if } K_i(x) > K_i(y); \\ 0, & \text{if } K_i(x) = K_i(y); \\ -1, & \text{if } K_i(x) < K_i(y); \end{cases}$$

The other form of SD is considering the differences in importance of the criteria expressed as corresponding weights. The difference $[K_i(x) - K_i(y)]$ between the values of criteria can also be meaningful from the viewpoint of the preference between decisions. If that happens a large difference indicates a stronger preference. This information can be also incorporated into the measure of SD. So in the most general form SD can be calculated by formula:

$$\mu_i(x,y) = \begin{cases} w_i[K_i(x) - K_i(y)], & \text{if } K_i(x) > K_i(y); \\ 0, & \text{if } K_i(x) = K_i(y); \\ -w_i[K_i(x) - K_i(y)], & \text{if } K_i(x) < K_i(y); \end{cases}$$

The advantage of using the SD over other measures of preferences is that it allows the construction of a transitive binary preference relation, regardless of the nature of the set of criteria. To do this the Integral Superiority Degree is constructed (ISD):

$$F(x,y) = \sum_{\forall z \in X} [f(x,z) + f(z,y)].$$

This provides a good basis for the comparison of all the alternatives and their linear ordering according to the preferences $R(\alpha)$:

$$R(\alpha) = \{(x,y) \in X \times X : F(x,y) \geq \alpha, 0 \leq \alpha \leq 1\}.$$

81

We call this relation α-level preference relation [14]. The set of maximal decisions of this binary preference relation can have a controllable volume depending on the value of α. This is a useful property to be used in a decision support procedure, as was mentioned before. This would mean that the sub-set of 'the best' decisions provided to the decision maker for the final choice can be increased and decreased according to the demands of the decision situation. This approach is similar to the ELECTRA type multicriteria procedures [13] but gives broader results because of the more general approach to the definition of the preference relations.

In this generalised form, multicriteria decision making theory is applicable to the comparison of cases from the case base.

If the current decision situation, and the set of existing cases, can be described by the same set of characteristics (qualitative and/or quantitative) then the retrieval of the relevant cases can be based on their Superiority Degree (SD) and Integral Superiority Degree (ISD) built in a similar way to that suggested for multicriteria decision making. For each decision situation, the system will calculate a measure of similarity for each of the cases based on the criteria set by the decision maker. This measure of similarity will enable the most relevant cases for that particular decision to be retrieved.

6.   Multicriteria Case Comparison

To propose a multicriteria comparison, let us consider first a general form of representation of cases and the existing approaches to the similarity assessment.

Let each case be described by a set of characteristics. Thus if C and D are cases in a case-base they can be described as sets :

$C = \{c_1, ..., c_N\}$, and

$D = \{d_1, ..., d_N\}$,

where N - is a total number of characteristics (attributes).

The level of importance of each attribute i may be represented as a weight w_i (i = 1, ... , N). It is usually assumed that these weights are numbers in the range between 1 and 0.

Under these assumptions the similarity measure between each two cases C and D can be introduced by a general formula:

$$S(C,D) = F(w_i, M(c_i, d_i)),$$

where F and is a function representing an aggregation procedure of single similarity measures M between the values of attributes and weights of the attributes.

One of the most commonly used approaches in similarity assessment is based on the weighted sum of the similarity measures between attributes and is known as a Nearest Neighbour Algorithm [12]

$$S(C,D) = \sum_{i=1}^{N} w_i M(c_i, d_i).$$

To keep the values of this function between 0 and 1 it should be normalised in one of two ways:

a) divided by the total number of attributes:

$$S(C,D) = \sum_{i=1}^{N} w_i M(c_i, d_i) / N; \text{ or}$$

b) a total sum of their wights:

$$S(C,D) = \sum_{i=1}^{N} w_i M(c_i, d_i) / \sum_{i=1}^{N} w_i.$$

This form of the measure of similarity is based on the similar assumptions as the SD described above. The difference is that this function does not express the relevance of a particular case to a decision situation as it is based of a physical value of the characteristics for the comparison. The other disadvantage is that it is not asymmetric, so that S(C,D) = S(D,C). This is true if the measure is supposed to reflect a 'pure distance' between points in the characteristics space. For more general purposes, if for example the

83

measure of similarity has to reflect the order of comparison, the asymmetry of the similarity measure is a more natural condition. The weakest asymmetry condition would be $S(C,D) \neq S(D,C)$.

Furthermore, the above proposed measure do not have explicitly expressed form for the measure between the characteristics and their values. This measure is basically dependent on the kind of a case-based tool used for the development of the case-based reasoning system.

Moreover, this kind of a measure assumes that all the cases are described using the same characteristics and/or information is available about the values of those characteristics for all the cases. Generally speaking, this is not always true. So there is a need to make a distinction between the cases that can be compared according to the same characteristics and those having only some of the characteristics matching those that are used in the representation of the cases in the case base. Using multicriteria representation of cases, instead of pure list of characteristics, allows a case-base to be linked to a particular decision situation and makes comparison of cases specific to the decision-situation.

There are various case-based reasoning applications which use this measure reported in a wide collection of the CBR literature [12]. However, in the context of decision support it is important to allow decision maker to make changes to a list of characteristics and have a flexible measure of importance of the characteristics, and their values, depending on the goal (purpose) of the decision making process. It is impossible to reflect these changes in the measures used in conventional CBR.

We propose multicriteria representation of cases in addition to the characteristic based one. It would be a means of linking existing cases to a situation that a case-based intelligent decision support system is addressing. With such an approach a decision maker has to first identify a list of characteristics that are relevant and important for the current decision situation and then assess the level of usefulness of the values of these

characteristics within the particular decision context. The level of usefulness will be taken as a value of criteria of that characteristics. The selection of the relevant cases from the existing case base will be based on the criteria values instead of the values of characteristics. Comparison between cases then can be quantified by calculating the SD between cases. A multicriteria decision making approach in a general form would then be applicable for identification of the most relevant case (or set of cases) to assist decision maker in a current decision context.

7. Conclusions

The nature of IDSS and the type of decision situations where it can be applied implies a need for readily adaptable knowledge from the problem domain as well as a knowledge of problem solving techniques and approaches. In addition, some form of intelligence is essential for IDSS. A system that can learn from experience and adjust itself to a specific situation will make the decision support intelligent.

In this paper we have studied the role of the case-based approach to intelligent decision support. Past knowledge from similar decision situations may be a useful source of information for decision support. The stored cases recording past decisions become a form of organisational memory to support future decision making. The built-in learning ability of the case-base will facilitate the adaptation of the IDSS and provide the most relevant past experience to a specific decision situation.

If knowledge of past experience is represented as cases then there is a need for techniques to retrieve the relevant cases. We argue that multicriteria decision-making can be used if an appropriate expression can be found. The development of an appropriate multicriteria expression requires input from the decision maker in each case, thus adapting the existing knowledge to the specific decision making situation. Such input can be expressed in terms of importance of characteristics and values in a current decision context.

References

[1]   Burstein, F.V., Smith, H. G., Arnott, D. R. and O'Donnell, P. A. (1993). "On the Nature of Intelligent Decision Support Systems", in Proceedings of ANZIIS'93: Australian and New Zealand Conference on Intelligent Information Systems, Perth.:IEEE.

[2]   Kolodner, J.L. and Riesbeck, C. (1986). Experience, Memory, and Reasoning. Lawrence Erlbaum Ass., Pub.: USA.

[3]   Arnott, D.R. and O'Donnell, P.A. (1994). "What are Decision Support Systems?", in *Reading in Decision Support Systems*, (2nd ed.), Arnott, D.R. and O'Donnell, P.A.(ed.), Monash University, Melbourne.

[4]   Simon, H.A. *The New Science of Management Decision.* New York: Harper & Row, 1960.

[5]   Klein, M.; and Methlie, L.B. (1990). *Expert Systems: A Decision Support Approach.*, Addison-Wesley P.C.

[6]   Burstein, F.V. and Smith, H.G. (1994). 'Knowledge Modelling For Intelligent Decision Support'. in *Proceedings of 5th Australasian Conference on Information Systems*, Monash University, Melbourne, pp.635-647.

[7]   Slade, S. (1991). "Case-Based Reasoning: A Research Paradigm". *AI Magazine*, Spring, pp.42-55.

[8]   Hammond, K.J. (1989). *Case-Based Planning: Viewing Planning as a Memory Task.* Academic Press, Inc.: US.

[9]   Schank, R.C., Abelson, R.P. (1977). *Scripts, Plans, Goals and Understanding: An Inquiry into Human Knowledge Strategies* Lawrence Erlbaum Associates, Inc: USA.

[10]   Kolodner, J. L. (1993). *Case-Based Reasoning* . California: Morgan Kaufmann.

[11]   Richter, M. M.; Wess, S. Althoff, K.-D. and Maurer, F.(Eds) (1993). EWCBR-93: First European Workshop on Case-Based Reasoning (vol. I& II) Germany: EXXAI, ACM, SFB 314, 1-5 November 1993.

[12]   Watson, I and Marir, F. (1994) 'Case-Based Reasoning: A Review', *The Knowledge Engineering Review*, vol. 9, No. 4.

[13]   e' Costa, C. A. Bana, (ed.). (1990). Readings in multiple criteria decision aid, Springer-Verlag.

[14]   Burstein, F.V. (1980). "On one approach to multicriteria decision making problems." In: *Bulletin of the Academy of Sciences of the Georgian SSR*, Metsniereba, pp.517-520.

[15]   Burstein, F.V. (1983). *Multicriteria Decision Making Man-Machine Procedures Research and Design (with respect to the vector preference relation).* PhD Thesis: USSR Academy of Sciences.

[16]   Zhukovin, V.E., Burstein, F.B., Korelov, E.S. (1987). "A Decision Making Model with Vector Fuzzy Preference Relation", Fuzzy Sets & Systems, 22:71-79.

Software Tools For AP-3C Operators

Phil Silver

Air Operations Division, DSTO Salisbury
Phil.Silver@dsto.defence.gov.au

Abstract

The Australian P-3 maritime surveillance aircraft are being fitted with improved sensors and more powerful data processing capabilities. Following the introduction into service of the refitted aircraft (designated AP-3C) there will be opportunies to improve the performance of the platform through better data management, better presentation of data to the operators, automatic methods of target detection and other system enhancements. These improvements can be implemented by software changes. The paper suggests areas where there is scope for productive R&D.

Software Tools For AP-3C Operators

Phil Silver

Head Avionics Technology
DSTO

Objective

- Example of real system
- Potential for software tools
- Stimulate research interest

Any opinions my own.

The AP-3C is a real tactical data system with real scope for enhanced effectiveness through software changes during the life-of-type of the aircraft.

Potential for Australian expertise to be applied.

Scope

- Orion P-3C systems
- AP-3C Data Management System
- Scope for system improvements

Presentation will give brief description of systems in Orion P-3C maritime patrol a/c

Concentrate on the DMS of the updated P-3C (designated AP-3C).

The AP-3C will be delivered with new sensors, spare computing capacity and potential for greatly increased computing power for future applications.

Identify some possibilities for increasing AP-3C effectiveness.

Orion P-3C Systems

- Radar
- Electronic Support Measures (ESM)
- Acoustics
- Magnetic Anomaly Detector (MAD)
- Infra-red Detection System (IRDS)
- Data Management System (DMS)
- Others (Comms, Nav, Intercom, ECM etc.)

Role of maritime patrol aircraft - to detect, classify and identify surface and sub-surface craft

ESM is a system for detecting and analysing electronic emissions over a very wide frequency band. It provides bearing to target and emitter characteristics.

Acoustics - sonobuoy deployment, monitoring, target detection, fixing and classification

MAD - submarine detection

IRDS - day/night imaging - limited range

4

Current Aircraft

- USN role
  - hunt nuclear submarines
  - fleet support
- Radar - weather radar
- ESM
  - retrofitted
  - limited capability
- Data Management System
  - Central computer
  - Drum mass memory
  - Overloaded and inadequate

USN role

       Current aircraft reflect past USN roles for P-3

       Operates in conjunction with fleet

       Hunts for nuclear submarines

       Emphasis on acoustics

       Not equipped for surface surveillance

Radar

       Modified weather radar.

       Limited capability for detection of ships/submarines

ESM

       New ESM being installed at present

       Formerly some aircraft had limited capability ESM

       Others none at all

DMS

       Central computer 1960s vintage

       Overloaded - flashing displays, lost data etc

Updated Aircraft AP-3C

- multi-mode surveillance radar
- state-of-the art ESM
- operator work stations
- multiple data bus avionics architecture
- multi-processor data management system
- provisions for DMS growth
- scope for development of software tools to aid operators

Radar	All the modes one would expect including imaging.
Oper. Stations	All displays generated by local computers
	High display refresh rates
	1280 x 1024 multi-colour displays
Architecture	Standard data buses for flexibility
DMS	Spare processing power and memory
	Provision for plug-in processor and memory expansion
	Hardware available to allow increased software functionality for enhanced capabiltiy of AP-3C.

TACCO Tactical Environment

- 64 deployed sonobuoys
- 8 weapon splash points
- 100 track-while-scan radar contacts
- 100 ESM contacts/tracks
- 5 IRDS contacts/tracks
- 50 acoustic contacts/tracks
- 5 MAD contacts/tracks
- 200 data link tracks
- 5 visual contacts

Indicative list of the capabilities of the DMS and the decision making responsibilities of the TACCO.

Information derived from the unclassified AP-3C SOR.

TACCO Functions

- Control and management of the DMS
- Control and management of armament and ordnance functions
- Selection for launch, setting and release of search stores
- Selection, setting and release of weapons
- Construction of sonobuoy patterns for sonobuoys
- Selection and control fly-to points
- Computer aids for selective tracking of targets
- Control of sonobuoy plot stabilisation
- Control and use of display aids for operator interpretation of data
- Control of DMS system test
- Control of airborne crew training functions
- selection of digital coastline maps
- backup controls for NAVCOMM

Derived from the AP-3C SOR.

All functions are significant.

TACCO has a high work load.

Data Handling Methodology

- AP-3C methodology same as P-3C
- Contact assignment currently sensor operator responsibility
  - Sensor operator assigns contacts, fixes, tracks
  - Prior to assignment sensor data held in sensor database
  - Assigned contacts held in tactical data base
  - Tactical data base accessible by all operators

Sensor operators process target data locally and assign the contact data to the DMS after determining that it is of tactical importance.

Conversely, a contact will not be entered into the tactical data base unless the sensor operator is convinced that it is significant.

Potential For Software Tools

- Fusion of sensor contact data
  - On-board sensors
  - Other platforms and sources
  - Changes to DMS architecture?
- Operator display improvements
  - Enhanced clutter filtering tools
  - Colour enhancement
  - Innovative displays
- Target classification aids
  - Tools for imaging radar
  - Fusion of imaging sensor data?
- Optimal search paths

On-board sensors - Principally radar and ESM

Other platform sensors

        command and control issue - who makes decisions?

        data link transmits contacts data, classification data and quality estimates

Architecture changes

        Is it appropriate for sensor operators to have veto over detections/decisions of their sensor?

        Is capacity of data link adequate to handle the large number of contacts

Clutter filtering tools - Some already provided. What is the best way?

Colour - Used to descriminate types of features, objects. New ways?

Innovative displays - Walt Disney graphic designers, 3D

Tools For Imaging Radar - assessing shapes, lengths etc.

Optimal Search Paths - travelling salesman problem

20

Australian Self-Sufficiency

- US export restrictions on software
- No guarantee of best algorithms from foreign vendors
- Vulnerability to policy changes by foreign governments
- Potential for Australian R&D

Software release restrictions affect almost all Australian defence systemacquistions.

Platform capabilities are largely determined by software.

Australia is vulnerable to the availability and support of foreign software.

Short tem policy changes by foreign governenst have and will contiue to cause difficulties with software release and support.

The AP-3C role is different from those of other maritime surveillance aircraft.

It is a good candidate for indigenous R&D.

Summary

- Example of modern surveillance system
- Considerable potential for enhanced performance
- Real application for sensor fusion, software tools

Brief Analysis of a Multi-Sited Multi-Sensor Inshore Surveillance System for US Navy

Ian Croser

CEA Technologies Pty Ltd
Canberra
06 280 7932

CEA Technologies has exported a surveillance system. Ian Croser will present a demonstration of some of the decision support aspects of this system, developed for the US Navy.

GDFS/MIUW

JMCIS

NTDS

Ship Sensors

Ship Equivalent

Inter-unit Communication Network

- Intelligence Input
- Group Oriented Track Management

- Real Time Track Management
- Real Time Sensor Management
- Real Time C³ I
- Analysis and Training

- Automated and Manual Detection
- Tracking
- Reporting

Control

Data

Intelligence

JMCIS
COMMAND INFORMATION SYSTEM

GRAPHICAL DATA FUSION SYSTEM

SENSOR SYSTEMS

Objects: ships, helos, aircraft, submarines (others)

JMCIS Operators

GDFS Operators

Acoustic Operators

Graphical Data Fusion System

GDFS

for the

Mobile Inshore Undersea Warfare

System Upgrade

Mission Statement

Amphibious Warfare Support
Port and Harbor Defense
Special Operations Support
Area Surveillance
Training Operations
Logistics Operations Support

Capabilities Statement

Sensor Management
Track Management
Mission Planning/Analysis
C^3I Support

FIGURE 7: MIUW SYSTEM OPERATOR FUNCTIONS

OPERATOR FUNCTIONS

SENSOR MANAGEMENT
- Calibration & Setup
- Status Monitoring
- Operational Control & Adaption
- Performance & Quality Management
- Sensor Mode Selection/Control
- Self Test Operation

TRACK MANAGEMENT
- Detection Assistance
- Localization
- Track Assistance
- Classification
- Identification
- Data Fusion Decisioning
- Measurement & Data Readout
- Integration of External Tracks
- Track Selection

MISSION PLANNING, REVIEW ANALYSIS & TRAINING
- Recording Control
- Replay Control
- Measurement & Data Readout
- Data Archiving Selection
- Compilation of Track Logs
- Mission Analysis
- Performance Prediction
- Training Scenario Input
- Map Setup

C3I
- Selection of Tracks for Prosecution
- Selection of Tracks for Transmission
- Sensor Conflict Resolution
- Threat Assessment
- Assistance to Prosecution Resources
- Navigational Assistance
- Reception and Collation of External Track Reports
- Provide Operational Guidelines to Sensor Operators

Operator

Log Archival Manager

Video → Image Buffer

SNAPSHOT MANGER | Relational list

Archived Contact Log Files

Relational links

Relational links

24 HOUR LOG FILE

2200 hrs | 2300 hrs | 0000 hrs | 0001 hrs | 0002 hrs

Operator

Relational link

Snapshot image files

Relational links

Case Files

HIGH INTEREST TRACKS DATA

USS SHARK | Ship 7 | Ship 22 | Ship 55 | Ship 1

Contact Log Data

Incident/Event Enhanced Data

TRACK FILTER AND SELECTOR

Selects recording interval for Contact Log tracks.

Selects rules & tracks for Incident/Event Recording.

Selects updates for HIGH INTEREST TRACKS.

System Tracks

Sensor Tracks

System State

Operator Name

Operator Comments

HIT LIST MANAGER

Operator Selections

Archived Files

Files for Replay (Logs and Images)

Selected Archived Files

FILE TRANSLATOR & PRINTER

Converts files to Textual representation of data.

Operator file & function selection

DAT

(Logs & Images)

To Other Systems

Printer

Printed Data & Images

105

Sensor based situational awareness
as a hazard paradigm
for optimisation of ATC systems design
To be published in the SPIE Aerosense '95 Proceedings
on Air Traffic Control Technologies - April 1995

Neale L. Fulton

Commonwealth Scientific and Industrial Organisation,
Division of Information Technology, Canberra, Australia, 2601

ABSTRACT

This paper discusses a systems engineering approach to design and implementation of Air Traffic Control Systems (ATCS). Preservation of situational awareness by optimum use of available sensors is used as a unifying paradigm for airspace structural design which yields significant increases in reliability of operation as measured by the potential to detect collisions and effect avoidance.

Strategic and tactical data required for continuous situational awareness is dependent on efficient and timely capture of sensor information. Analytical relationships between airspace structure and sensor search and acquisition functions are mathematically related. The reliability of ATCS airspace structures as mission critical components and probability of failure of these functions are derived.

Modelling is used to show strong interdependencies between visual acquisition, cruising rule and tactical communications. The limitations of various airspace structures in use are identified. System reliability is baselined against well known acceptance standards. Improvements of five orders of magnitude in performance and reliability are demonstrated with flow on effects to the reliability of overall ATCS design.

The sensor paradigm is used to postulate an extension to current separation criteria and facilitate identification of fundamental failure modes for ATCS design. New flow model criteria enabling critical airspace structures, performance and geographic areas to be identified by simulation or real time performance monitoring are identified thus enabling quantitative measures required to baseline and improve system performance. The paper concludes by showing how modelling/ real time monitoring can be used to predict system trends and capacity problems well in advance of actual system failure.

Keywords: see and be seen, visual acquisition, situational awareness, collision avoidance

1. INTRODUCTION

The central tenet of the International Civil Aviation Organisation's (ICAO) Principles of Human Centred Automation[1] states that 'aviation systems (aircraft and ATC) automation exists to assist human operators (pilots and controllers) in carrying out their responsibilities'. This tenet is further elaborated by 'Human operators should never be held liable for failures or erroneous decisions unless they have full control and command of the system', and 'automation must not be designed in such a way that it can subvert the exercise of the human operator's responsibilities'. To achieve these objectives and philosophies a detailed analysis of the needs and limitations of pilot and controller alike are required. Of particular concern is the impact which the structure of airspace has on operator limitations in terms of system safety and reliability.

Historically since the early 1950's airspace structure has been based on either the Quadrantal or Hemispherical cruising rule [2,3] which in turn were further augmented with increasing degrees of positional communications in later decades. More recently there has been an international aviation industry objective to standardise procedures particularly with the introduction of satellite technology for the Communications, Navigation and Identification functions. This process has in some countries resulted in changed airspace structure in terms of cruising rule and specific implementations of voice communications. Trends have been to standardise airspace structure on the Hemispherical rule and minimise positional communications for Visual Flight Rules (VFR) traffic. Both VFR and Instrument Flight Rules (IFR) pilots, who have experienced operations under both cruising rules are concerned with the resultant reduction in reliability and safety of separation / segregation standards. The reductions come from the lessening of the degree of situational awareness and visual acquisition performance when operating under these conditions.

A sensor paradigm for visual acquisition provides an objective basis for comparison of system performance. A degradation in acquisition performance of 10^5 (for typical General Aviation aircraft speeds of 160 knots) for the unalerted Hemispherical rule as compared with the alerted Quadrantal rule has been demonstrated [4]. This paper shows that for the unalerted Hemispherical rule to achieve the same reliability as the alerted Quadrantal rule there would have to be a three fold reduction in aircraft closing speed. Also seen is the limited future potential of reliable performance of the Hemispherical rule as the average speed of the airborne fleet increases with time[5]. There is a need to re-assess ATC designs, services and structure from a total systems engineering reliability and safety perspective. This paper proposes a systematic approach incorporating human factors for both airborne and ground based operators yielding objective performance measures and criteria for the collision avoidance functions.

The paper reviews the role of situational awareness in Section 2, the pertinent aspects of Systems Engineering in Section 3 and develops a frame work for constructing a reliability model in Section 4. Section 5 shows how the use of a sensor paradigm leads naturally to an extension of the existing separation metrics. Section 6 demonstrates how a graphical user interface can be used to display and alert an air traffic control operator to potential conflicts. Conclusions are summarised in Section 7.

2. SITUATIONAL AWARENESS

A pilot must handle large quantities of real time, often continuous, data and perform several demanding tasks concurrently, usually under severe time constraints. In Perry's [61] review an FAA report notes "many human shortcomings including limited cognitive abilities when dealing with complex situations, poor vigilance over long periods of monitoring and vulnerability to error when dealing with large amounts of written or spoken data". The ICAO Meeting[5] also expresses this concern.

Performance advantages come with optimisation of the situational awareness and the reduction of semantic distance with respect to information presented to the pilot. The concepts of situational awareness and semantic distance[7,8] have evolved from aircraft cockpit design and human factor interface research over many decades. Adam[9] defines situational awareness by posing the questions 'What is situational awareness? It's simply knowing what is going on so you can figure out what to do! What are other aircraft's intentions, my intentions and my options?' Situational awareness can be defined more formally as: 'the extent to which the pilot has the knowledge needed to perform a specified task or tasks'[8].

Situational awareness (SA) has two components: tactical SA (e.g. visual acquisition) and Global SA (e.g. knowledge of expected traffic flow ahead). Pilots are in the aircraft to make good tactical decisions and execute them. However the probability of the correctness of a tactical decision is in direct proportion to the situational awareness of the pilot. W. Cotton [7] notes 'the potential hazards of reduced situational awareness are tremendous. Most accidents attributable to pilot error occur because the pilot was unaware of danger: the malfunctioning of some onboard system, hazardous weather, proximity to terrain or aircraft'.

Ballas, Heitmeyer and Perez[8] define semantic distance as the difference between the users intentions based on a real-world-model metaphor and the meaning of the same intentions presented as expressions available in an human-machine interface. In other words, if the pilot must perform a large amount of thought to interpret data in order to reconstruct a mental image of the real-world scene, the semantic distance is large. There have been constant endeavours in aviation to reduce semantic distance for the representation of real-world scenarios[10]. The evolution of the Horizontal Situation

Indicator, Artificial Horizon, Weather Radar and Head Up Display are but a few examples. In the context of collision avoidance, both situational awareness and semantic distance provide the conceptual basis for the definition of performance metrics for the design of Air Traffic Control systems.

3. SYSTEMS ENGINEERING REVIEW

Systems Engineering provides a disciplined approach to the reliability and safety engineering aspects of Air Traffic Control design.

Reliability analysis is used to assess and evaluate design options, ensuring that the system and its various components will meet the prescribed levels of performance. Stress-strength characteristics of the systems components are of primary concern in reliability assessments. These characteristics can be determined for both physical components and processes upon which the system is dependent. For example, a communications channel has a certain information carrying capacity which describes its conceptual strength. The required rate of transfer of information to be delivered in order for the system to operate reliably can be defined as a stress. In visual acquisition the probability of a collision (which must include the probability of a missed detection) can be conceptualised as a system stress whereas the system strength can be identified when this probability exceeds a certain threshold. A conservative design would ensure that stress is always less than strength for reliable operation, and that any operation alleviating short period build-ups of stress do not cause deleterious effects in other functions of the system.

Safety analysis systematically evaluates the causes of, and effects on safety of operation due to systems component failures. Human Factors Error Analysis and Failure Mode Effect and Criticality Analysis are major tasks within reliability analysis. MIL-STD-882B [11] which forms one basis for system safety design analysis states:

> *'Decisions regarding the resolution of identified hazards will be based on assessment of risk involved. To aid achievement of the objectives of system safety, hazards shall be characterized as to the hazard severity categories and hazard probability levels, when possible.*
> ..... **the priority for system safety is eliminating hazards by design.....**'

Johnson [12] cites 'human intervention as a primary factor in the cause and exacerbation of accidents in safety-critical systems' as the common finding of a number of international bodies concerned with safety-critical systems. Johnson [12] also notes Rasmussen's assertion that 'designs should only be accepted if the human contribution to risk can be measured'. Hazard severity categories are defined to provide a qualitative measure of the worst credible mishap resulting from personnel error; environmental conditions; design inadequacies; procedural deficiencies; or system, subsystem or component failure or malfunction.

Qualitative definitions of hazard severity exist for design and maintenance of mission critical aircraft systems [13] and components. Catastrophic failure signifies death or system loss where continued safe flight and landing ceases to be possible. Hazardous failure signifies severe injury, severe occupational illness, or major system damage, continued safe flight and landing are at severe risk; there is a potential for catastrophe. Clearly the risk of mid-air collision qualifies as a catastrophic failure at worst or hazardous failure at best.

Hazard classifications also assign permissible probabilities of occurrence where the probability that a hazard will be created during the planned life expectancy of the system is described as potential occurrences per unit time, events, population, items, or activity. Systems designs subject to catastrophic failure criteria are required to achieve a probability of occurrence of less than one in 10^{-9} and Hazardous failure criteria require a probability of occurrence less one in 10^{-7}. These criteria, when expressed in appropriate metrics, form an objective basis for hazard assessment.

4. ATC SYSTEM RELIABILITY MODEL

Air Traffic Control (ATC) systems are established to assure the orderly flow of traffic through a given volume of airspace providing a means of separation between aircraft. Most airspace models depict the physical scenario of geographic terrain, airspace classification boundaries, radio frequency boundaries, etc. Systems Engineering requires that other models be created to address the design as a whole and in a unified manner. In the context of this paper we are concerned with the instantiation of the reliability model shown in Figure 1. The model is decomposed as two co-centered

Figure 1: ATC collision avoidance reliability model
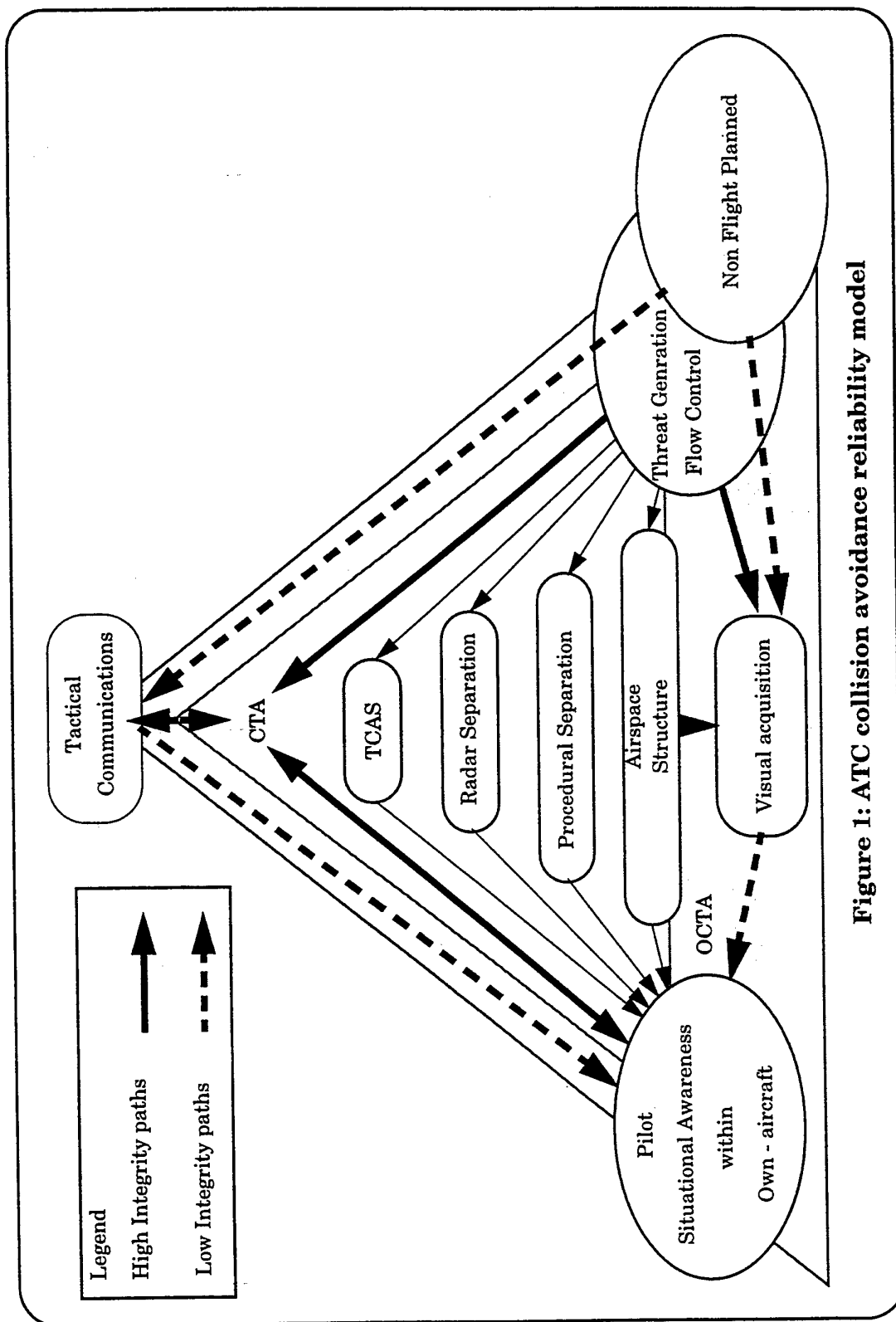
triangles where the inner triangle represents the services provided by ATC and the outer triangle represents the basic services which are capable of being provided by pilots independent of ATC services.

The lower left vertex of the triangle represents management of situational awareness within own-aircraft. Maintenance of situational awareness is based on information flows derived from a number of different channels, including visual acquisition, communications, airspace structure and ATC services (only some of services are shown in the inner triangle). Situational awareness therefore represents the strength of the system.

Airspace structure is defined by intangible airspace attributes including the cruising rule; structure of communications boundaries and areas; separation and segregation standards. The design of airspace structure, as defined, is a major determining factor in all safety, reliability and economic assessment.

It is airspace structure which determines the base level of operation of the airways system in terms of failure rate. It is particularly important to include these attributes in design assessments when goals are set to ascertain how given quantifiable levels of safety in terms of equipment and services can be achieved. The setting of this base level directly impacts the cost of additional services and equipment required to be provided to achieve a given overall acceptable level of system reliability. The higher the base failure rate of the system the greater the differential stress imposed on additional system component performances and therefore the greater the component performance required in reducing the overall system failure rate to an acceptable level.

Stress, from a collision perspective, is imposed on the system by the presence of all aircraft which have potential to be in a specified proximity of own-aircraft. This stress is represented by the lower right vertex of the triangle.

The inner triangle and the apex of the triangle (tactical communications comprising of VHF, UHF or HF communications) represents the mediums by which system stresses are communicated to own-aircraft. Assessments of separation and segregation standards need to include all aircraft within a given physical distance of own-aircraft and should not discriminate, for collision purposes, between IFR and VFR categories[14]. Management of situational awareness for own-aircraft at the system level must account for all aircraft within separation or segregation correlation volumes. Within a particular situation in Controlled Airspace (CTA) the pilot of own-aircraft may only be aware of some of these threats as ATC filters the data to only essential information in high workload scenarios. A major failing of present ATC systems is that currently separation standards are stated in the most part, only in terms of flight planned traffic (predominantly IFR) and fail to recognise the presence of or take account of other non-flight planned VFR traffic[15].

Tactical communications and the other ATC services can be used as a means of optimising visual acquisition. Radio communications is usually a shared resource between pilots and ATC operational staff. As such its design must be optimised for both classes of user within the constraints of physical channel capacity [16].

4.1 Threat generation

Past work in collision avoidance has concentrated on the evaluation of mid-air collision through phenomenological studies such as the gas model formulation that Alexander[17] has used to estimate interaction distances and collision frequencies as a function of aircraft density metrics. May[18] provides a method of predicting the number of near mid-air collisions in a defined airspace. The work of Alexander pertains more to operations Outside Controlled Airspace (OCTA) and originates from the United States of America whereas the work of May pertains more to traffic corridors and originated in the United Kingdom but was first developed as the need for an analytic technique was seen for the reorganisation of Swedish airspace.

Graham and Orr[19] and Britt and Schrader[14] describe the collision scenario in terms of relative velocity and position space vectors. Holt and Marner[20] provide a general treatment of collision hazard expressing relationships between separation standards, velocity and acceleration limits while Britt and Schrader[14] provide techniques for evaluating collision warning systems. All the referenced models provide useful descriptions for macroscopic behaviour but due to the lack of detail of the interaction behaviour, (for example, visual acquisition) do not permit, by themselves the optimisation of the overall system characteristics. Figure 2 (overleaf) shows representative values typical of current threat densities and characteristics.

Figure 2: Aircraft density and interactions per hour
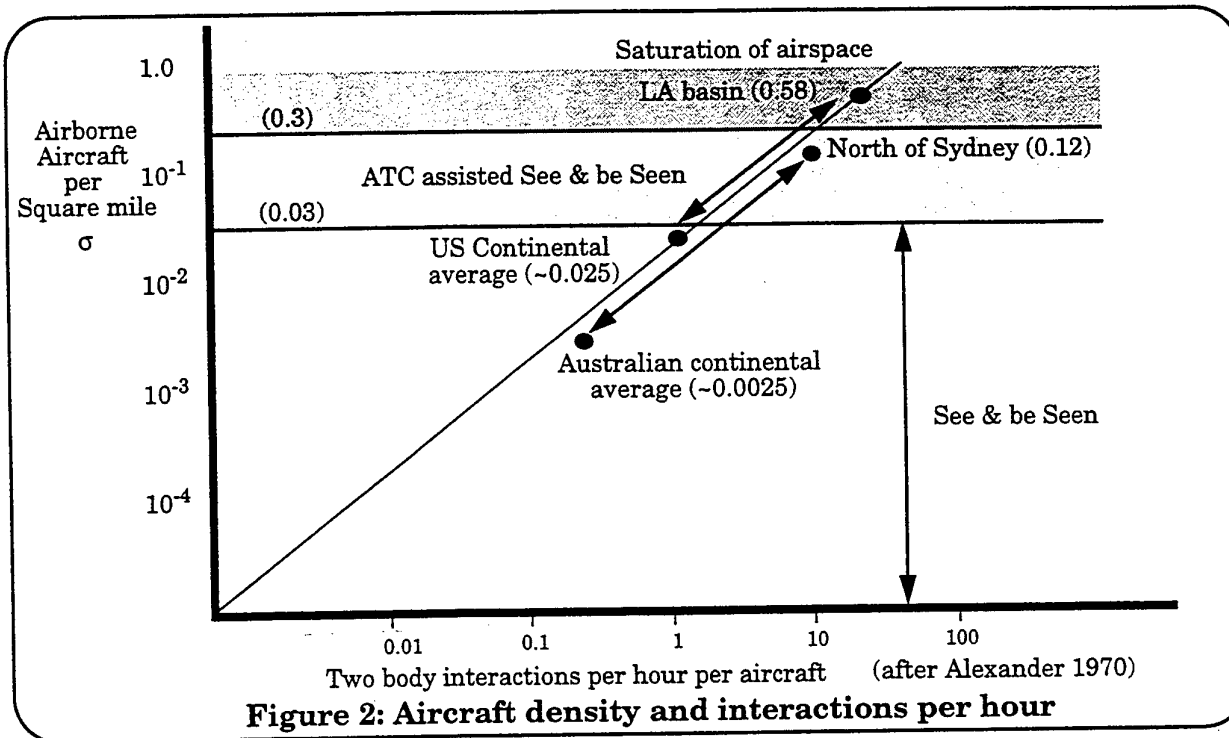
4.2 Visual acquisition

Optimal performance of the ATC system can be gained through a more thorough analysis of the performance of visual acquisition. Fulton[4] analyses the ability of a pilot to reliably perform visual acquisition using standard sensor systems analysis and compares the results against prescribed catastrophic and hazardous failure rates for aerospace systems[13].

Perhaps the most widely known phenomena of a mid-air collision is the fact that when two aircraft are on collision course then the relative velocity vector between the two aircraft is stationary[14, 19]. When the two aircraft are at the same altitude this translates to the threat aircraft maintaining a stationary azimuthal bearing until impact[14]. This phenomena can be used to greatly simplify the analysis of the situation to the point where the ICAO scan[21] can be easily simulated. This scan assumes that the outside airspace over a 90 degree Field of Regard (FOR) can be scanned in 20 seconds using nine fixations comprised of a Field of View (FOV) of 10 degrees with two seconds dwell and one return sweep of the cockpit for flight control purposes of two seconds.

The simulation assumes that the pilot is maintaining this scan pattern. Thus as the line of constant bearing is bracketed by a FOV dwell there is an opportunity for the pilot to detect the threat.

Harris[22] provides the probability of detection (P_d) of a DC3 aircraft expressed as a function of range and deviation in azimuth from fixation boresight. From this data the probability of missed detection (P_{md}) can be determined ($P_{md} = 1-P_d$) for any range and azimuth. A large number of near mid-air collisions (NMAC's) reported involve transport/General Aviation (GA) and GA/GA aircraft combinations[23]. The simulation therefore linearly scales the results of Harris to that of a typical GA aircraft with a wing span of 30 feet. P_d and P_{md} are calculated out to the limit (or near limit) of visual acuity being 1 arc minute[24, 25] for 20/20 vision. Hovanessian[26] describes the single-look probability of detection P_d at range applied repeatedly to determine the overall probability of detection for a threat moving towards an observer. For each fixation bracketing the threat the range of the threat is known, so P_d and P_{md} can be found. Since each fixation bracketing the threat is an independent event, $P_{md\_overall}$ for that scenario can be found from the product of each independent P_{md}. In general:

$$P_{md-overall} = P_{md(n)}......x\ P_{md(3)}\ x\ P_{md(2)}\ x\ P_{md(1)} \tag{1}$$

where the iterative variable n takes limits defined by the limit of acuity and the threshold of reaction.

The work of the FAA Advisory Circular 90-48-C [27 p15 refers] is also implemented in the model. Essentially this says that in visual acquisition it will take up to 12.5 seconds to fixate, recognise and initiate avoiding action. Therefore the scan starts at the range determined by this time and the simulated closing speed and propagates backout in 20 second intervals determining ranges and P_{md}'s until the limit of visual acuity is reached.

Each $P_{md-overall}$ as a function of closing speed can be plotted to determine the total effectiveness of the ICAO scan as a component of an Air Traffic Services system. The reliability of the scan pattern can be measured by the probability of failure which is $P_{md-overall}$ which in turn can be compared with required reliability levels for system components when suffering catastrophic failure in order to establish mission success/failure criteria. Performance criteria for threats not on the fixation boresight and the influence of alerted communications on visual acquisition are also modelled[4].

A comparison in performance of the Quadrantal and Hemispherical rules with respect to visual acquisition is provided in Figure 3 where the relative closing speed in knots for a given probability of missed detection (failure) is given as a parameter.
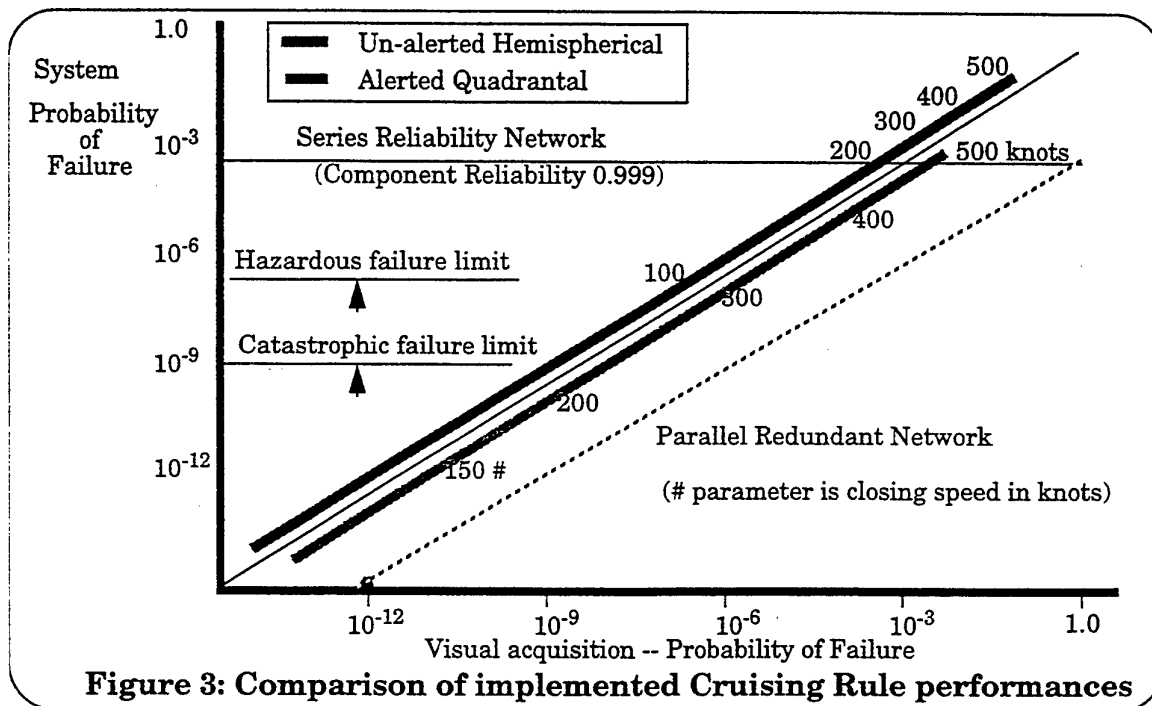

Figure 3: Comparison of implemented Cruising Rule performances

4.3 Redundancy in service implementations of a ATC system

An Air Traffic Control system is built on system redundancy: we are used to thinking, for example, that should the radar services fail then a procedural protocol service will provide separation which in turn should this fail will be backed up by communications and visual acquisition. The implied reliability model is a parallel construct. But this assumption needs to be stringently tested. If any one activity, service or component is dependent on another service, activity or component for its reliability performance then the resultant systems level reliability model will be a series model. Series redundancy is conceptualised by a paradigm of the weakest-link-in-the-chain where the overall system will only be as strong as the failure rate of the weakest link. In a system treated as the following three components: ATC (A), Communications (C) and visual acquisition (V) the reliability of the system is given by:

$$R_{SeriesSystem} = Reliability_A \cdot Reliability_C \cdot Reliability_V$$

Reliability is the probability of success so in terms of failure this translates to:

$$FailureProbability_{SeriesSystem} = FailureProbability_A + FailureProbability_C + FailureProbability_V + O2$$

If, on the other hand, the service is capable of independently providing a reliable segregation / separation function then the ATC designer will have the potential to utilise a parallel redundancy model. Parallel redundancy has a multiplicative effect in terms of reducing the overall failure rate where the overall system failure rate is proportional to the product of the individual failure rates. The reliability of the parallel redundant system is given by:

$$R_{ParallelSystem} = 1 - (1 - Reliability_A) \cdot (1 - Reliability_C) \cdot (1 - Reliability_V)$$

and in terms of failure rate for the parallel redundant case:

$$FailureProbability_{ParallelSystem} = FailureProbability_A \cdot FailureProbability_C \cdot FailureProbability_V$$

The variation in performance of visual acquisition under differing closing speeds manifests as changes in instantaneous failure probability which directly reflects in the base failure rate of the overall system. The resultant system reliability is dramatically different as shown in Figure 3. At low closing speeds visual acquisition alone can meet system performance goals, whereas at high closing speeds visual acquisition needs to be augmented in a parallel instantiation with communications in order to achieve the overall system goals. Most modern instantiations of ATC systems have high closing speeds and poor applications of communications resulting in series reliability models with high failure rates.

The segregation function should be capable of being maintained by either communications or by visual acquisition independently in a parallel model. Under the alerted Quadrantal rule this is true and overall system reliabilities of 10^{-9} can easily be achieved for closing speeds up to 300 knots. Under unalerted Hemispherical rule for the same individual threat speeds neither communications (vis the absence of) nor visual acquisition alone is capable of providing segregation performance to acceptable standards and the overall system reliability remains that of the weakest service, being 10^{-3}. (Figure 3 assumes a communications probability of failure of 10^{-3} being a typical hardware failure rate for this component but as shown in the next section this is not the only communications failure mechanism to be accounted for). Thus, as shown in Figure 3, in the series system example, for closing speeds below 200 knots the overall systems reliability is 10^{-3}. For closing speeds above 200 knots the system reliability becomes that of visual acquisition, while the parallel systems redundancy model achieves a failure rate 10^{-3} below that of visual acquisition for all closing speeds.

It is therefore imperative that when designing ATC system services the greatest degree of independence between service functions is maintained, by design, to ensure the overall system design performance is not jeopardised.

4.4 Tactical Communications

Fulton[4] shows that tactical communications can influence the performance of visual acquisition by two orders of magnitude with consequential impacts on the overall system failure rate. It is therefore imperative to ensure that all tactical communications can perform at high levels of integrity and reliability. Failure mode assessments must include not only the hardware component failures but also the theoretical and operational information theory aspects particularly channel capacities and real time response times from the airborne perspective.

ATC mean service rates can be determined from direct measurement of service times and their relative frequencies in the various operational scenarios. A similar exercise can establish peak system level airborne request rates and their relative frequencies based on aircraft density metrics. Adequate margins must be applied to these information theory based analyses to avoid infinite queuing times / lengths as request rates approach service rates.

Degradation in service rate and response times are directly affected by phenomena such as over-transmissions, communications repeater delays greater than the human auditory reaction times of 110 ms[25], or non-optimal phraseology and operational language.

A common system occurrence when using satellite communications is repeated transmissions due to frequency overlaying, whilst improving the utilisation of scarce ATC services can in fact cause the overall system to loose operational efficiency and channel capacity through the requirement for transmission repeats. Additional system imposed stresses on pilots increase in proportion with the increased need to filter irrelevant (to own aircraft) information transmitted in real time.

Figure 4: Channel capacity and queuing theory responses
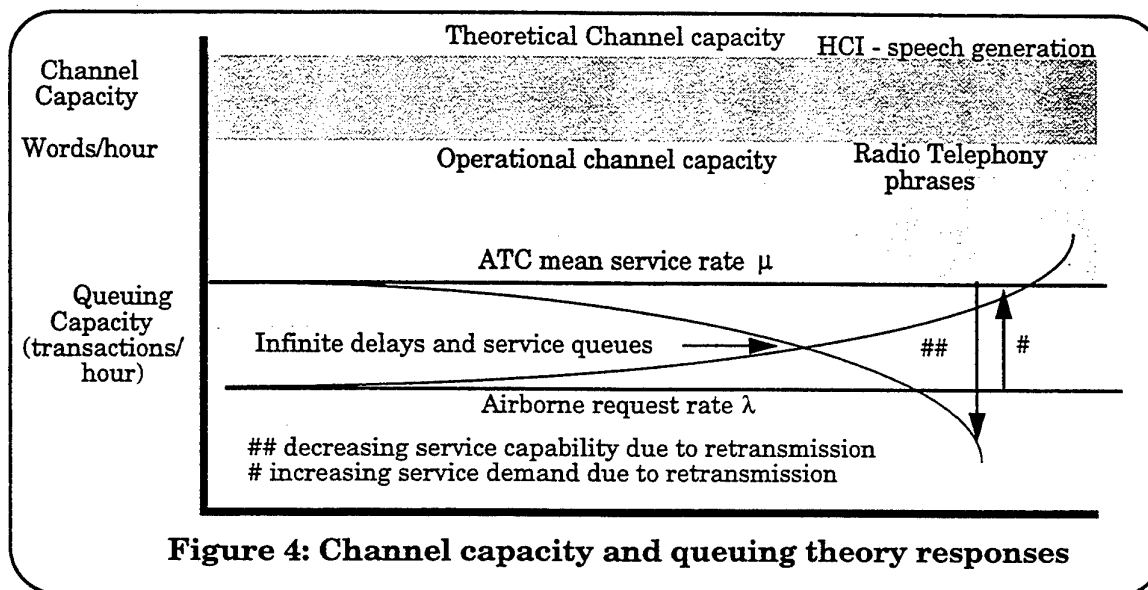
Airborne real-time requirements also need to be considered and will impose a maximum service response time on ATC units to provide onward clearances, necessitating in some instances prioritising of responses in what can otherwise be a random service request situation.

Additionally the design of air traffic control geographic radio frequency boundaries need to be assessed in terms of continuity of situational awareness. In many instances aircraft in close spatial proximity geographically may be unaware of each other from a visual and communications standpoint resulting in extremely low overall situational awareness and potential for resultant high system failure rates in these modes of operation.

5.0 PERFORMANCE METRICS FOR ATC SERVICES

Reliability analyses have been performed for individual sub-components of the ATC system such as the Instrument Landing System. Historically measurement of segregation / separation performance has been based on incursions of time and distance criteria[28]. Often the violation of a separation standard is subjectively judged by ATC observation on the Plan Position Indicator (PPI) radar screen. The ability to perform visual acquisition to a given standard, one of the most fundamental components of the ATC system, has received less analysis and attention than many other components although failure here can reduce the performance of the overall system by many orders of magnitude.

Previous statistical analyses on Mid-air and Near Mid-air Collisions[23,29] have defined random variables which are essentially derived from the threat perspective (e.g. gas theory phenomenological model) and only peripherally address causal effects derived from the visual sensor based paradigm. These analyses are thus limited in their effectiveness in providing insight to fundamental design criteria and failure modes. Progress in the design and modelling of sensors such as Forward Looking Infra-red (FLIR), millimetre wave (MMW) and Electro-optics make it clear that new extension to

existing metrics is feasible for visual acquisition to enable the optimum design, monitoring and future growth of ATC systems.

It is proposed that metrics which encapsulate the sensor based visual acquisition paradigm be developed:

1. Airspace scenario criteria: Traffic density which is a primary determinant for collision pair generation as a function of airspace classification, altitude band, lateral and vertical velocities, and stress of weather restricting available altitudes, or under-utilising other altitudes (e.g. VFR levels in Instrument Meteorological Conditions (IMC)).

2. Own-aircraft visual acquisition criteria: Visual acquisition is a function of Field of Regard (a function of relative flight path angle and windshield vision), Field of View, time to point of closest approach, and time for acquisition (a function of pilot work load, relative closing speed (a function of closure angle and of cruising rule), required probability of detection threshold, alerted communications, and time to complete a set of scans through the whole of the Field of Regard).

3. Media transmission / threat detection criteria: signal to noise ratio, sun's position, visibility/haze, camouflage, collision lights and contrast ratios.

6.0 REAL TIME MONITORING

Advances in computer visualisation, artificial intelligence, scheduling algorithms, simulation techniques and readily available computing power now make it practical to consider both simulation of total airspace management in terms of airport traffic flow modelling and the real time monitoring within Air Traffic Control systems. Current flow simulations already include performance with respect to individual aircraft types, realistic navigation performance, the correct modelling of Standard Instrument Departures (SIDs), Standard Arrival Routes (STARs) and instrument approaches. Traditional conflict resolution metrics with respect to horizontal, vertical and longitudinal criteria can now be extended to include estimates of pilot's field of regard, flight profile and probability of visual detection as a real time function.
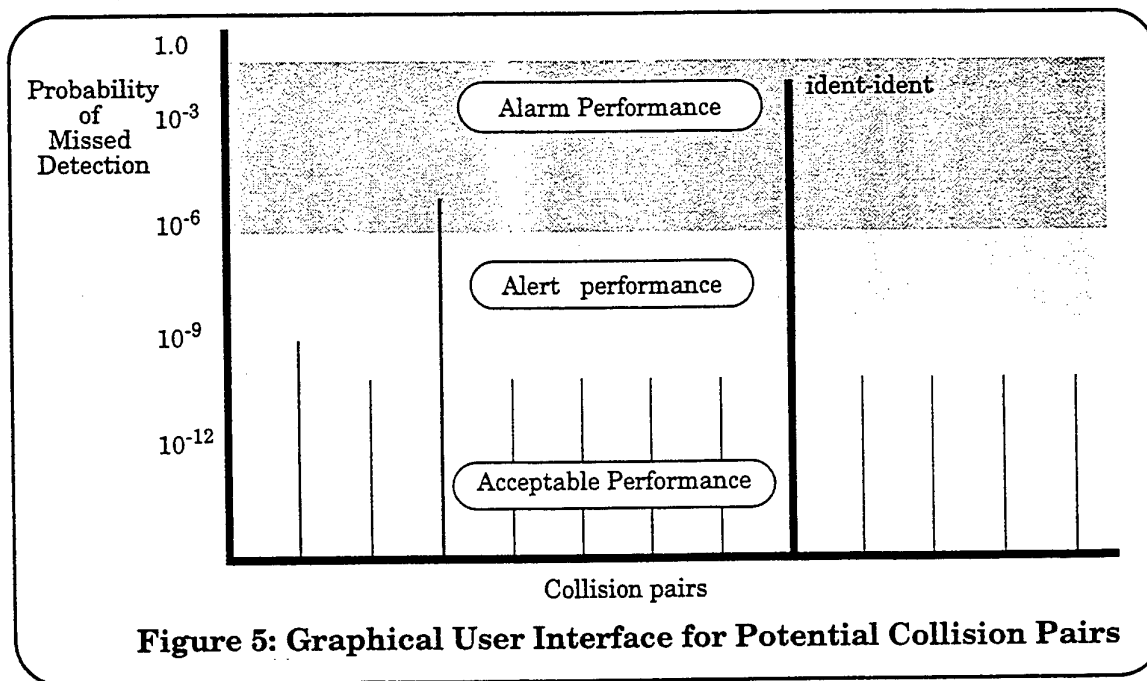
Figure 5: Graphical User Interface for Potential Collision Pairs

Monitoring of traditional separation standards and visual acquisition real time performance provides the necessary information to baseline system failure performance. Modelling and real time monitoring provide the necessary

information to identify geographic areas in which weak system performance occurs and to produce trend analyses of system stress data for management and control purposes. Figure 5 illustrates one representation of the visual alert data on a graphical user interface. Each line in the histogram represents a collision pair. The greater the probability of missed detection the greater the propensity for system failure. Both visual and aural alarms can be generated as the estimates of system failure exceed certain specified criteria shown as Acceptable, Alert and Alarm performance.

7.0 CONCLUSIONS

In this paper we have presented a strategy to develop a reliability model for an Air Traffic Control system. Advances in sensor technology, simulation and computing power now make significant advances in ATC systems design, monitoring and control possible. However to capture the benefits the performance of the ATC system must be understood and modelled. The impact of human factors in prescribing design criteria is identified as is the need to provide a system which can meet well identified reliability criteria. The role of visual acquisition and tactical communications in the overall performance of the system is emphasised as a critical and often neglected system component. The performance of visual acquisition, tactical communications and cruising rule structure are inter-related. Visual acquisition of other aircraft in near proximity to own-aircraft forms one of the foundations of the operation of any Air Traffic Control system, and indeed there is a legal responsibility for pilots to perform this task. Yet the reasonable physical limits with which this human endeavour can be achieved have been inadequately addressed from a systems engineering and reliability perspective. In particular, airspace structure comprised of cruising rule design, communications management, and situational awareness as defined in this paper requires serious engineering review. System baselining as proposed provides the method and approach for future growth and performance advances.

8.0 REFERENCES

1. ICAO, Human Factors in CNS/ATM Systems, Human Factors Digest, ICAO Circular 249-AN/149, Chapter 4, pp 21-25, ICAO, Montreal, Canada

2. ICAO, Manual of Air Traffic Procedures for Pilots, ICAO Doc 8090-AN/871, pp 5-4, pp A-1, ICAO, Montreal, Canada, 1 August 1960

3. The Canada Gazette, Part II Volume 90, cruising Altitudes, SOR/56 151 Tables 1 (Green or Red airways or Air Routes) & Table 2 (IFR else-where), Montreal, Canada, May 9, 1956

4. Neale Fulton, "An objective analysis of the 'See and be Seen' Hazard in modern Air Traffic Control Systems," CSIRO Technical Report, Canberra, Australia, March 1994.

5. ICAO, Rules of the Air, Air Traffic Services and Search and Rescue Divisions, Report of the Meeting Montreal, ICAO Doc 7909, RAC/SAR, Montreal, Canada, 21 October - 14 November 1958,

6. Tekla S. Perry, Senior Editor, "Improving the world's largest, most advanced system," IEEE Spectrum, pp 22-36, February 1991

7. William B. Cotton, "Air Safety: the view from the cockpit," IEEE Spectrum, pp 71-74, August 1975.

8. James A. Ballas, Constance L. Heitmeyer and Manuel A. Perez, "Evaluating Two aspects of Direct Manipulation in Advanced Cockpits," CHI '92, pp 127-134, 1992

9. Eugene C.Adam, "Tactical Cockpits - The coming revolution" IEEE AES Systems Magazine, pp 20-26, March 1994.

10. Peter W. Pruyn and Donald P. Greenburg, "Exploring 3D Computer Graphics in Cockpit Avionics" IEEE Computer Graphics & Applications, pp 28-35, May 1993.

11. MIL-STD-882B, System Safety Program Requirements incorporating Notice 1, US Department of Defense, 1 July 1987

12. C.W. Johnson, "A probabilistic logic for the development of safety-critical interactive systems", Int. J. Man-Machine Studies(1993) 38, Academic Press Limited, pp 333-351

13. J.W. Saull, Chief Surveyor, Airworthiness Division, CAA, "Objectives of Maintenance - Airworthiness," Aerospace, pp 41- 47, July/August 1987.

14. Charles L. Britt, and Systems James Schrader, "A Statistical Evaluation of aircraft Collision-Hazard Warning System Techniques in the Terminal Area," IEEE Transactions on Aerospace and Electronic Systems Vol. AES-6, No. 1, pp 10 - 21, January 1970.

15. ICAO, ICAO Rules of the Air, ICAO, Doc-4444-RAC/501/12, Volume 1 Second Edition, ICAO, Montreal, Canada, 1982

16. C.E.Shannon, and W.Weaver, The Mathematical Theory of Communication, University of Illinois Press, 1972

17. Ben Alexander, "Aircraft Density and Midair Collision," Proceedings of the IEEE, Vol 58, No. 3, pp 377 - 381, March 1970

18. G.T.E. May, "A Method for Predicting the Number of Near Mid-air Collisions in a Defined Airspace", The Journal of The Institute of Navigation, Volume 24, No. 2, pp 204-218, 1971

19. Walton Graham and Robert H. Orr, "Separation of Air Traffic by visual Means: An Estimate of the Effectiveness of the See-and Avoid Doctrine," Proceedings of the IEEE, Vol. 58 No. 3, pp 337 - 361, March 1970

20. John M. Holt and Gene R. Marner, "Separation Theory in Air Traffic Control System Design," Proceedings of the IEEE, Vol. 58, No. 3, pp 369 - 376, March 1970

21. ICAO, Pilot Skills to make "look-out" more effective in visual Collision Avoidance, ICAO Circular 213-AN/130, pp 7, ICAO, Montreal, Canada, 1989,

22. Harris, J.L, "Visual Aspects of Air-Collision in Visual Search," National Academy of Sciences, Washington D.C. 1973

23. Roger Rozelle, Editor, "Aviation Statistics: Near Midair Collision Update," Flight Safety Digest Vol. 8 No.12, pp 9-13, Dec. 1989,

24. Bruce J. Schachter, "Computer Image Generation for Flight Simulation," IEEE CG & A, October 1981.

25. Editors Dr. Cott and Dr. Kinkade, Human Engineering Guide to Equipment Design, pp 50, pp 300, Joint Army-Navy-Airforce Steering Committee, U.S. Government Printing Office, Catalog Number D 4.10: EN 3, Stock Number 008-051-00050-0, 1972

26. S.A. Hovanessian, Radar System Design and Analysis, Artech House Inc, 1984,

27. BASI, Limitations of the See-and-avoid principle, pp15, BASI Research Report, Canberra, Australia, April 1992,

28. ICAO, Procedures for Air Navigation Services, Rules of the Air and Traffic Services, Twelfth Edition - ICAO Doc 4444-RAC/501/12, pp 3-1 to 3-12, ICAO, Montreal, Canada,1985

29. Joel M. Yesley, "A study of the Relationships between Near Midair Collisions (NMAC's), Midair Collisions (MAC's), and some potential causal factors", Federal Aviation Authority, DOT/FAA/ASV-91-1, October 1990

AUSTRALIA

1. **DEFENCE ORGANISATION**

a. **S&T Program**
> Chief Defence Scientist ⎫
> FAS Science Policy ⎬ shared copy
> AS Science Industry and External Relations⎪
> AS Science Corporate Management ⎭
> Counsellor Defence Science, London (Doc Data Sheet)
> Counsellor Defence Science, Washington (Doc Data Sheet)
> Scientific Adviser to MRDC Thailand (Doc Data Sheet)
> Director General Scientific Advisers and Trials/Scientific Adviser Policy and Command (shared copy)
> Navy Scientific Adviser
> Scientific Adviser
> Air Force Scientific Adviser
> Director Trials

> **Aeronautical and Maritime Research Laboratory**
> Director

> **Electronics and Surveillance Research Laboratory**
> Director

> **Chief of Maritime Operations Division**
> RLMOR
> G. Gibbon
> Mario Selverstrel, AOD
> Clinton Heinze, AOD
> Keith P. Mason, EWD
> Douglas Kewley, HFRD
> Christopher Crouch, HFRD
> Phil Silver, AOD

> **DSTO Library**
> Library Fishermens Bend
> Library Maribyrnong
> Library DSTOS (2 copies)
> Australian Archives
> Library, MOD, Pyrmont 2 copies

b. **Forces Executive**
> Director General Force Development (Sea)
> Director General Force Development (Land), (Doc Data Sheet)
> Director General Force Development (Air)

c. **Navy**
> SO (Science), Director of Naval Warfare, Maritime Headquarters Annex, Garden Island, NSW 2000. (Doc Data Sheet)
> CMDR Chris Donald, ASSTASS PD Campbell Park, Canberra

d. **Army**
> ABCA Office, G-1-34, Russell Offices, Canberra (4 copies)

e. **Air Force**
> WGCDR Rick Owen, DRR-AF, Russell Offices, Canberra

f.   **S&I Program**
         Defence Intelligence Organisation
         Library, Defence Signals Directorate (Doc Data Sheet only)

g.   **B&M Program (libraries)**
         OIC TRS, Defence Central Library
         Officer in Charge, Document Exchange Centre (DEC), 1 copy
             *US Defence Technical Information Centre, 2 copies
             *UK Defence Research Information Center, 2 copies
             *Canada Defence Scientific Information Service, 1 copy
             *NZ Defence Information Centre, 1 copy
             National Library of Australia, 1 copy

2.   **UNIVERSITIES AND COLLEGES**

         Australian Defence Force Academy
             Library
             Head of Aerospace and Mechanical Engineering
         Senior Librarian, Hargrave Library, Monash University
         Librarian, Flinders University

3.   **OTHER ORGANISATIONS**

         NASA (Canberra)
         AGPS

OUTSIDE AUSTRALIA

4.   **ABSTRACTING AND INFORMATION ORGANISATIONS**
         INSPEC: Acquisitions Section Institution of Electrical Engineers
         Library, Chemical Abstracts Reference Service
         Engineering Societies Library, US
         American Society for Metals
         Documents Librarian, The Center for Research Libraries, US

5.   **INFORMATION EXCHANGE AGREEMENT PARTNERS**
         Acquisitions Unit, Science Reference and Information Service, UK
         Library - Exchange Desk, National Institute of Standards and
         Technology, US

6.   **ADDITIONAL DISTRIBUTION**
         Paul Compton, School of Computer Science and Engineering, University of New
         South Wales, Sydney
         Key Centre for Advanced Computing Studies, School of   Computing   Sciences,
         University of Technology, Sydney, PO Box 123,   Broadway, NSW   2007
             Andrew Blair,
             John Debenham
             Jenny Edwards
         Gil Tidhar, Australian Artificial Intelligence Institute, 171 La Trobe St, Melbouren, 3000
         Trisan Chui, BHP Engineering, PO Box 1237, North Ryde, NSW, 2059
         Laurie Lock Lee, BHP Information technology, PO Box 216, Hamilton, NSW, 2303
         Department of Information Systems, Monash University, Melbourne
             Frada V. Burstein
             Helen G. Smith
         Ian Croser, CEA Technologiers Pty Ltd, Canberra
         Neale L.Fulton, CSIRO, Division of Information Technology, Canberra

SPARES (10 copies)   **Total number of copies: 76**

DEFENCE SCIENCE AND TECHNOLOGY ORGANISATION DOCUMENT CONTROL DATA	1. PRIVACY MARKING/CAVEAT (OF DOCUMENT)

2. TITLE DSTO Workshop on Intelligent Decision Support Systems, Held 7 July, 1995, Pyrmont	3. SECURITY CLASSIFICATION (FOR UNCLASSIFIED REPORTS THAT ARE LIMITED RELEASE USE (L) NEXT TO DOCUMENT CLASSIFICATION) Document (U) Title (U) Abstract (U)
4. AUTHOR(S) G. Gibbon, Editor	5. CORPORATE AUTHOR Aeronautical and Maritime Research Laboratory PO Box 4331 Melbourne Vic 3001

6a. DSTO NUMBER DSTO-GD-0079	6b. AR NUMBER AR-009-627	6c. TYPE OF REPORT General Document	7. DOCUMENT DATE January 1996

8. FILE NUMBER	9. TASK NUMBER	10. TASK SPONSOR	11. NO. OF PAGES 117	12. NO. OF REFERENCES

13. DOWNGRADING/DELIMITING INSTRUCTIONS -	14. RELEASE AUTHORITY Chief, Maritime Operations Division

15. SECONDARY RELEASE STATEMENT OF THIS DOCUMENT

Approved for public release

OVERSEAS ENQUIRIES OUTSIDE STATED LIMITATIONS SHOULD BE REFERRED THROUGH DOCUMENT EXCHANGE CENTRE, DIS NETWORK OFFICE, DEPT OF DEFENCE, CAMPBELL PARK OFFICES, CANBERRA ACT 2600

16. DELIBERATE ANNOUNCEMENT

No limitations

17. CASUAL ANNOUNCEMENT   Yes

18. DEFTEST DESCRIPTORS

Command and Control systems, Modelling; Operations research; Decision support techniques