

NAVAL POSTGRADUATE SCHOOL

Monterey, California



THESIS

THE CONDUCT AND ASSESSMENT OF A2C2 EXPERIMENT 7

by

Wendell L. Pasaraba

September 2000

Thesis Co-Advisors:

William G. Kemple
Susan P. Hocevar

Approved for public release; distribution is unlimited.

REPORT DOCUMENTATION PAGE		Form Approved OMB No. 0704-0188	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.			
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2000	3. REPORT TYPE AND DATES COVERED Master's Thesis
TITLE AND SUBTITLE : The Conduct and Assessment of A2C2 Experiment 7		5. FUNDING NUMBERS	
6. AUTHOR(S) Wendell L. Pasaraba			
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000		8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A		10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.			
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.		12b. DISTRIBUTION CODE	
ABSTRACT (maximum 200 words) Adaptive Architectures for Command and Control (A2C2) Experiment 7 is the latest in the series of experiments designed to investigate the effects of modifying current military organizational structures. It is a continuation of A2C2 Experiment 4, which compared the performance of a mission-optimized architecture to a non-optimized traditional architecture. The focus of A2C2 Experiment 7 involves the introduction of complex, unexpected tasks requiring multi-node coordination into the simulation scenario, and the examination of two disparate command and control architectures in dealing with these unexpected tasks. The two architectures, by design, differed in the amount of coordination required to accomplish the known scenario mission tasks. The "Autonomous" optimized architecture's design emphasized inter-nodal autonomy in performing mission tasks, while a "Interdependent" non-optimized architecture, resembling a traditional Joint Task Force (JTF) organization, operated with greater "inter-nodal" coordination. The research team expected the non-optimized architecture to have an advantage over the optimized architecture when dealing with the complex unexpected tasks, due to the higher coordination practiced in the "Interdependent" architecture. The experiment used the accuracy and latency scores of accomplishing each unexpected task as the two primary measures examined. A detailed statistical analysis is performed on the measures and the results discussed.			
14. SUBJECT TERMS Adaptive Architectures for Command and Control (A2C2)			15. NUMBER OF PAGES 146
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

THE CONDUCT AND ASSESSMENT OF A2C2 EXPERIMENT 7

Wendell L. Pasaraba
Lieutenant, United States Navy
B.S., University of Southern California, 1993

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN SYSTEMS TECHNOLOGY
(COMMAND, CONTROL, AND COMMUNICATIONS)**

from the

**NAVAL POSTGRADUATE SCHOOL
September 2000**

Author:

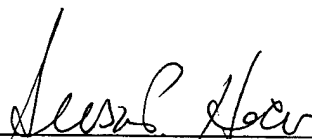


Wendell L. Pasaraba

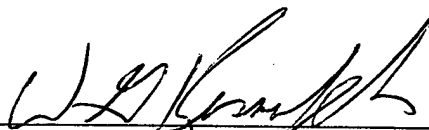
Approved by:



William G. Kemple, Thesis Advisor



Susan P. Hocevar, Thesis Co-Advisor



for Dan C. Boger, Chairman
C4I Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Adaptive Architectures for Command and Control (A2C2) Experiment 7 is the latest in the series of experiments designed to investigate the effects of modifying current military organizational structures. It is a continuation of A2C2 Experiment 4, which compared the performance of a mission-optimized architecture to a non-optimized traditional architecture. The focus of A2C2 Experiment 7 involves the introduction of complex, unexpected tasks requiring multi-node coordination into the simulation scenario, and the examination of two disparate command and control architectures in dealing with these unexpected tasks. The two architectures, by design, differed in the amount of coordination required to accomplish the known scenario mission tasks. The "Autonomous" optimized architecture's design emphasized inter-nodal autonomy in performing mission tasks, while a "Interdependent" non-optimized architecture, resembling a traditional Joint Task Force (JTF) organization, operated with greater "inter-nodal" coordination. The research team expected the non-optimized architecture to have an advantage over the optimized architecture when dealing with the complex unexpected tasks, due to the higher coordination practiced in the "Interdependent" architecture. The experiment scenario simulated a six node JTF conducting an amphibious operation. The experiment used the accuracy and latency scores of accomplishing each unexpected task as the two primary measures examined. A detailed statistical analysis is performed on the measures and the results discussed.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	A2C2.....	1
B.	BACKGROUND – A2C2 EXPERIMENT 4.....	2
C.	PURPOSE OF A2C2 EXPERIMENT 7.....	3
	1. Real World Motivation.....	4
	2. Experimental Questions.....	4
	3. Experimental Approach.....	5
	4. Anticipated Results.....	6
	5. Scope of Experiment.....	6
II.	EXPERIMENTAL DESIGN.....	7
A.	OVERVIEW.....	7
B.	SETUP.....	7
	1. Physical.....	8
	2. Lead Team.....	9
	3. Test Subjects.....	10
	4. Special Equipment.....	11
	5. Scenario.....	12
	6. Schedule of Trials.....	14
C.	HYPOTHESES.....	15
D.	ASSUMPTIONS.....	18
	1. Experimental Assumptions.....	18
	2. Statistical Assumptions.....	18
E.	STATISTICAL DESIGN OF EXPERIMENT.....	19
F.	MEASURES.....	20
G.	INSTRUMENTATION.....	24
H.	TESTING AND PILOT TRIALS.....	24
III.	DATA DESCRIPTION.....	27
A.	RAW DATA.....	27
B.	DATA CODING SCHEME.....	27
C.	DATA PROBLEMS.....	28
D.	DATA TABLE.....	29
E.	DATA REDUCTION.....	29
IV.	DATA ANALYSIS.....	31
A.	ANALYSIS PLAN.....	31
B.	METHODOLOGY.....	31
C.	RESULTS OF ANALYSIS.....	33
	1. Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in accuracy on the performance of Unanticipated Tasks.....	34

2.	Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in latency on the performance of Unanticipated Tasks.	36
3.	Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in accuracy on the performance of Multi-node Defensive Tasks.....	38
4.	Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in latency on the performance of Multi-node Defensive Tasks.....	41
5.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in accuracy on the performance of the seven Primary Mission Tasks.	43
6.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in latency on the performance of the seven Primary Mission Tasks.	46
7.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in accuracy on the performance of Single-node Defensive Tasks.	48
8.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in latency on the performance of Single-node Defensive Tasks.	50
9.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture on Mission Scores.	52
10.	Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture on Strength Scores.....	54
11.	Hypothesis: There is a difference in performance between the two architectures across the measures as a whole.....	56
12.	Hypothesis: Performance on the second trial is better than the performance on the first trial.	59
V.	CONCLUSIONS	61
A.	HYPOTHESIS RESULTS - INTERPRETATIONS	61
1.	Hypothesis Number One.....	61
2.	Hypothesis Number Two.....	61
3.	Hypothesis Number Three	62
4.	Hypothesis Number Four	64
5.	Hypothesis Number Five.....	65
6.	Hypothesis Number Six.....	65
7.	Hypothesis Number Seven	67
8.	Hypothesis Number Eight	68
9.	Hypothesis Number Nine	69
10.	Hypothesis Number Ten	70
11.	Hypothesis Number Eleven.....	71
12.	Hypothesis Number Twelve	72
B.	EXPERIMENT SUMMARY	73

VI.	LESSONS LEARNED AND AREAS FOR FUTURE EXPERIMENTATION	79
A.	EXPERIMENT PREPARATION.....	79
1.	Simulation Modification	79
2.	Test Subject Preparation	81
3.	Experiment Documents.....	81
4.	Data Recording Equipment.....	82
5.	Information Distribution	83
B.	FUTURE EXPERIMENTATION	83
1.	Continuation of Experiment 7.....	83
	APPENDIX A. ARCHITECTURES	85
	APPENDIX B. UNANTICIPATED TASK DESCRIPTION	89
	APPENDIX C. DDD-III TUTORIAL.....	91
	APPENDIX D. MISSION BRIEF.....	101
	APPENDIX E. PLAYER REFERENCE HANDOUTS	105
A.	UNANTICIPATED TASK INSTRUCTION SHEET	105
B.	TASKGRAPH.....	106
C.	QUICK REFERENCE GUIDE	107
D.	FRIENDLY ORDER OF BATTLE.....	108
E.	ENEMY ORDER OF BATTLE.....	110
F.	PRIMARY MISSION TASK REFERENCE GUIDE	112
	APPENDIX F. SAMPLE DEPENDENT VARIABLE FILE, DATA CODING SCHEME, AND DATA TABLE.....	115
A.	SAMPLE RAW DATA FILE.....	115
B.	DATA CODING SCHEME.....	122
C.	DATA TABLES	123
	LIST OF REFERENCES	125
	INITIAL DISTRIBUTION LIST	127

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

Figure 1.	Organizational Architectures.....	5
Figure 2.	Floor Plan Layout.....	8
Figure 3.	DDD-III Battle Space Screenshot.	12
Figure 4.	Taskgraph.....	14
Figure 5.	Dotplot of Unanticipated Task Accuracy.....	35
Figure 6.	Dotplot of Unanticipated Task Latency.....	37
Figure 7.	Dotplot of Multi-node Defensive Task Accuracy.....	40
Figure 8.	Dotplot of Multi-node Defensive Task Latency.....	42
Figure 9.	Dotplot of Primary Mission Task Accuracy.....	44
Figure 10.	Dotplot of Primary Mission Task Latency.....	47
Figure 11.	Dotplot of Single-node Defensive Task Accuracy.....	49
Figure 12.	Dotplot of Single-node Defensive Task Latency.....	51
Figure 13.	Dotplot of Mission Score.....	53
Figure 14.	Dotplot of Strength Score.	55
Figure 15.	Multi-node Defensive Task Probability Plot.	63
Figure 16.	Primary Mission Tasks Probability Plot.....	66
Figure 17.	Single-node Defensive Task Probability Plot.....	68
Figure 18.	Task Score Histogram.	74
Figure 19.	Unanticipated Task Score Histogram.....	75

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	Trial Schedule.....	15
Table 2.	Evaluated Task Characteristics.	21
Table 3.	Evaluated Tasks.	23
Table 4.	Architecture Performance Comparison.....	57
Table 5.	Order Effect.	59
Table 6.	Ceiling Effect on Primary Mission Tasks.....	75
Table 7.	Architecture Prediction Results.....	77

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. A2C2

Adaptive Architectures for Command and Control (A2C2) Experiment 7 is the seventh in the series of experiments conducted at the Naval Postgraduate School (NPS). The multi-institutional A2C2 program is sponsored by the Office of Naval Research and NPS participation is co-sponsored by the Institute for Joint Warfare Analysis (IJWA).

The A2C2 research program combines field-based, theoretically defined and experimentally designed research, and spans the spectrum from basic to applied research. The primary focus is model-based experimentation to examine adaptation within Joint Command and Control Architectures. The A2C2 program began in 1995, continuing a long tradition of Naval Postgraduate School command and control experimentation in collaboration with other command and control research activities. In addition to NPS, the A2C2 project members include researchers from private industry and four other universities. The first four A2C2 experiments were conducted at the basic research level, the fifth was a research calibration experiment, and the sixth was an actual transition event where A2C2 concepts and methodologies were applied to actual operating forces. The seventh experiment, completed in March 2000, continues the series of basic research experiments, following the research threads of the third and fourth.

B. BACKGROUND – A2C2 EXPERIMENT 4

A2C2 Experiment 4 is the experiment from which A2C2 Experiment 7 continues. A2C2 Experiment 4, a human-in-the-loop experiment simulating a Joint Task Force scenario, was designed to study the performance of three organizational architectures on several simulation tasks. These tasks, discussed later in Chapter II, Measures, varied in complexity, inter-nodal coordination required to accomplish the task, and in unpredictability of appearance. Because A2C2 Experiment 7 used only two of the three architectures from A2C2 Experiment 4, further discussion on the architectures will be limited to the two architectures used in both experiments. The two architectures were the "Autonomous" six-node optimized architecture and the "Interdependent" six-node traditional architecture. Each node represented a commander in a JTF conducting an amphibious operation. With the use of optimization modeling, the Autonomous Architecture was designed, among other things, to minimize the coordination required to prosecute tasks present in the simulation scenario. The Interdependent Architecture more closely resembled current military structures that are organized functionally. Compared to the Autonomous Architecture, the Interdependent Architecture required more inter-nodal coordination to accomplish the same scenario tasks. The results from A2C2 Experiment 4 showed that the Autonomous Architecture performed better than the Interdependent Architecture on predictable primary mission tasks, but suggested that the Interdependent Architecture may be more suited to performing more unpredictable tasks.

C. PURPOSE OF A2C2 EXPERIMENT 7

The preliminary results from A2C2 Experiment 4 suggested that the Autonomous Architecture, designed to be optimal for specific complex and predictable mission tasks, performed better than the more traditional, non-optimized Interdependent Architecture on the predictable mission tasks it was designed for. However, there was tentative evidence that the optimized structure was outperformed by the Interdependent Architecture in performing unpredictable tasks. This was thought to be due to the coordination required of the Interdependent Architecture in performing mission tasks; as the teams in this architecture were performing each successive task, they were also practicing coordination. In theory, this practiced coordination would make a team more adept at handling new and unpredicted tasks than a team whose organizational architecture did not foster practiced coordination. The results from A2C2 Experiment 4 indicated that the organization that minimized coordination outperformed the other structure on predictable tasks and the organization that practiced coordination outperformed on unpredictable tasks. [Ref. 4]

Because A2C2 Experiment 4 was not designed specifically to test for unpredictable tasks, additional research was needed to further test the impact of coordination on unpredictable tasks. A2C2 Experiment 7 was conducted to fulfill that need.

1. Real World Motivation

Today, as the force structure of the United States Armed Forces is being reduced, it is important to maximize the capabilities of the armed forces. The requirements are to do more with fewer personnel and fewer resources. This may lead future military command and control architectures to be optimized for specific mission tasks as explored in A2C2 Experiment 4. As these architectures become more optimized for the specific mission tasks they are assigned, it is critical that this optimization does not diminish the forces' capabilities to perform other tasks not used to define the optimal architecture. The ability to perform unexpected missions is also relevant as U.S. forces often shift from standard war-fighting missions to peacekeeping missions that require different capabilities. The threat of asymmetrical warfare also is applicable to this topic, because asymmetrical warfare is the enemy's use of unknown, unexpected, and unconventional means to defeat conventional forces.

2. Experimental Questions

One of the questions A2C2 Experiment 7 was designed to explore was the relationship between coordination capabilities and the adaptive response to unpredicted events; this had not specifically been examined in A2C2 Experiment 4. The primary question A2C2 Experiment 7 intended to investigate was whether an architecture that inherently required more coordination would benefit from this coordination when performing new and unanticipated tasks compared to a mission-specific, optimized architecture that required less coordination.

3. Experimental Approach

The experimental approach was similar to that of A2C2 Experiment 4, with the addition of Unanticipated Tasks to the scenario that were specifically designed for complexity and unpredictability. Unanticipated Tasks and other task categories are further discussed in Chapter II, Measures. The command and control architectures used in A2C2 Experiment 7, shown in Figure 1, were similar to those used in A2C2 Experiment 4. These two architectures were labeled Interdependent and Autonomous for A2C2 Experiment 7. The Interdependent Architecture resembled a more traditional architecture, and the Autonomous Architecture was optimized for specific predictable tasks. These architectures were then tested to observe how they performed the complex predictable and unpredictable tasks.

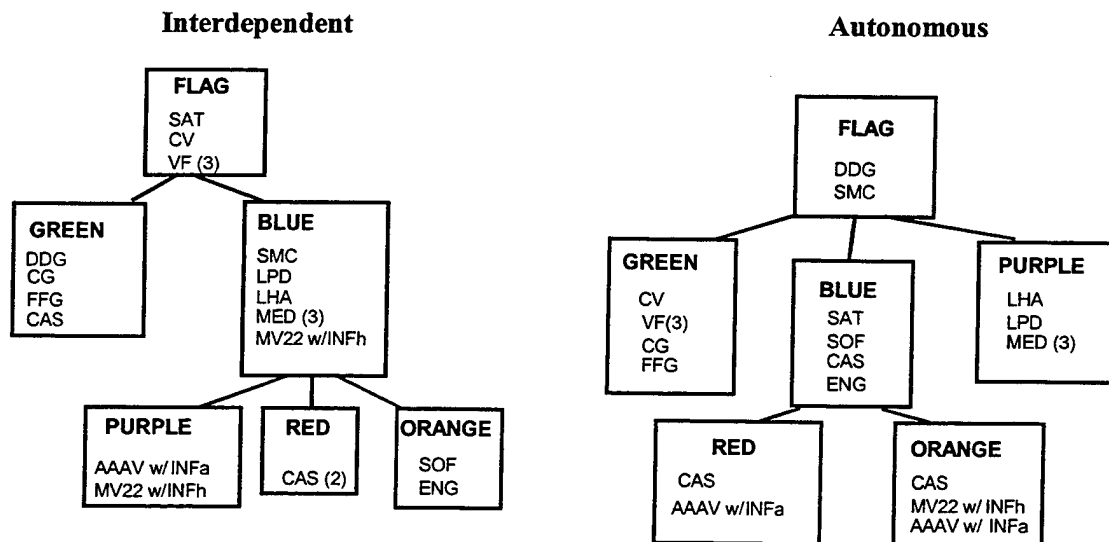


Figure 1. Organizational Architectures.

4. Anticipated Results

From the results of A2C2 Experiment 4, it was anticipated that the optimized Autonomous Architecture would perform better than the Interdependent Architecture in dealing with complex and predictable tasks. More importantly, we expected that the Interdependent Architecture would outperform the Autonomous Architecture in the prosecution of complex and unpredictable Unanticipated Tasks.

5. Scope of Experiment

The A2C2 research team examined the performance of the teams under the different architectures and compared the results with the modeler's expected results. NPS Joint C4I Systems students served as the Lead Team. In addition to supporting the A2C2 research team, the Lead Team focused on data collection and analysis efforts on the relationship between inter-nodal coordination and task performance. The lead team and its functions are discussed further in Chapter II, Lead Team. This analysis was scoped within the time available and the experience limits of the research team.

II. EXPERIMENTAL DESIGN

A. OVERVIEW

A2C2 Experiment 7 was developed by researchers at the Naval Postgraduate School and the Aptima Corporation, as part of the Office of Naval Research's Adaptive Architectures for Command and Control research project. The purpose of this experiment was to further examine the relationship between structural types and performance on predicted and unpredicted tasks. Specifically, the research question guiding the design of the experiment was: When faced with the need to respond to an unanticipated complex task, does a structural architecture that requires greater levels of inter-unit coordination provide a performance advantage over an architecture that reduces coordination by using a task-based design? [Ref. 5]

This chapter describes the details of the design of A2C2 Experiment 7. The following section gives a description of the setup of the experiment including equipment, lead team, test group, scenario driver, and schedule. Other sections describe the hypotheses, the assumptions pertaining to the data collected, the statistical design of the experiment, the measures used in analysis, instrumentation, and testing and pilot trials.

B. SETUP

The following paragraphs describe the laboratory environment used to support A2C2 Experiment 7. The physical layout of the simulation hardware and the communications equipment are described first. Other paragraphs detail the

lead team, test subjects, special equipment used for the experiment, scenario, and the schedule of the trial by the A2C2 research team.

1. Physical

The experiment was conducted using eight personal computer terminals running the Linux operating system. Six of the terminals were used by the six team members, one per terminal. The seventh terminal existed as the main server, and the eighth terminal was utilized by the experiment observers to monitor the progress of each trial. Placed in between the player terminals were temporary partitions, to limit informal visual and verbal communications between the nodes. The partitions were necessary to simulate the remoteness of each commander from other commanders in an actual JTF. The floor plan layout is illustrated in Figure 2.

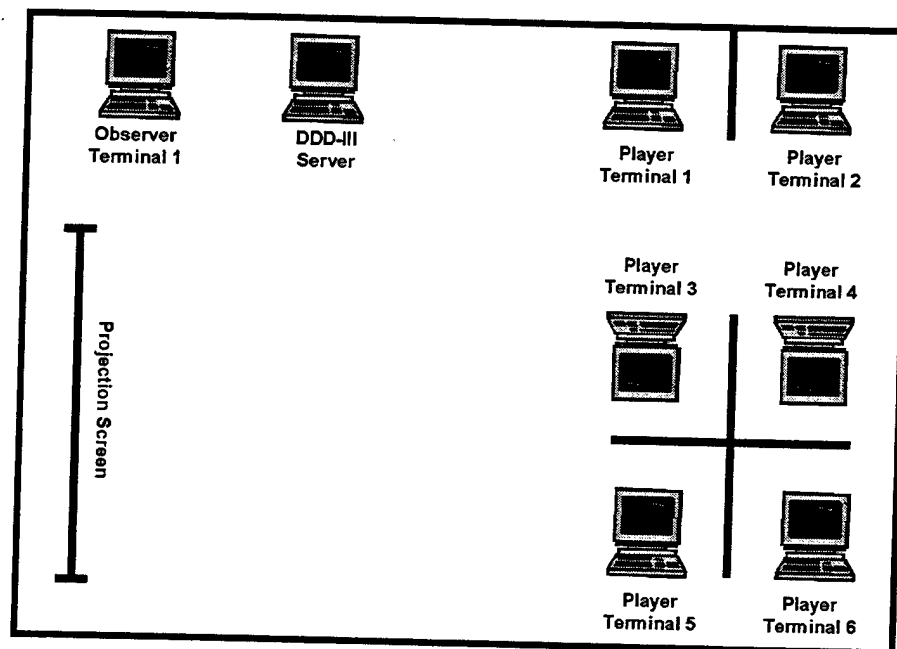


Figure 2. Floor Plan Layout.

Headsets with built-in microphones at each terminal simulated the Joint Task Force voice network. The headsets allowed for free two-way communications among the six players in the architecture. The communications equipment was also critical in monitoring and recording all voice communications between the nodes. During the subject training portion of the experiment, the headset system also allowed the trainers to instruct the players on the progress of each scenario. Observers monitored group dynamics, and examined the quality of task performance by listening to the team voice communications.

An audio tape recorder connected to the voice network recorded all voice communications to preserve the communications activity for further analysis. The visual output of Distributed Dynamic Decisionmaking (DDD-III) simulation, described later, was captured on videotape as well. This was accomplished by recording the video output from the observer terminal. Data capture of this nature allowed for future examination of the experimental results. To supplement the visual display from the computer monitors, a video projector projected the simulation image onto a large projection screen allowing additional personnel to observe the conduct of the experiment.

2. Lead Team

A lead team of six NPS officer students from the Joint Command, Control, Communication, Computers, and Intelligence (JC4I) Systems Curriculum contributed to the NPS effort in the A2C2 project. The lead team members were also students in course CC4103, C4I Systems Evaluation, in their next to last

quarter at NPS. The lead team was comprised of all U.S. Navy officers; three members had fleet operational experience, one had support experience, and two others were newly commissioned officers en route to training pipelines.

Before and during the experiment, the lead team performed the support and administrative tasks of: producing training documents, training subjects, preparing the laboratory, aiding in the debugging of the simulation, piloting the simulation scenarios, conducting the experimental runs, collecting data, and serving as observers. The lead team also served as subject matter experts on DDD-III, providing valuable assistance and advice during the trial runs. As part of the course CC4103, the Lead Team had the additional tasking of creating their own research questions, collecting data, and analyzing relevant results from the experiment.

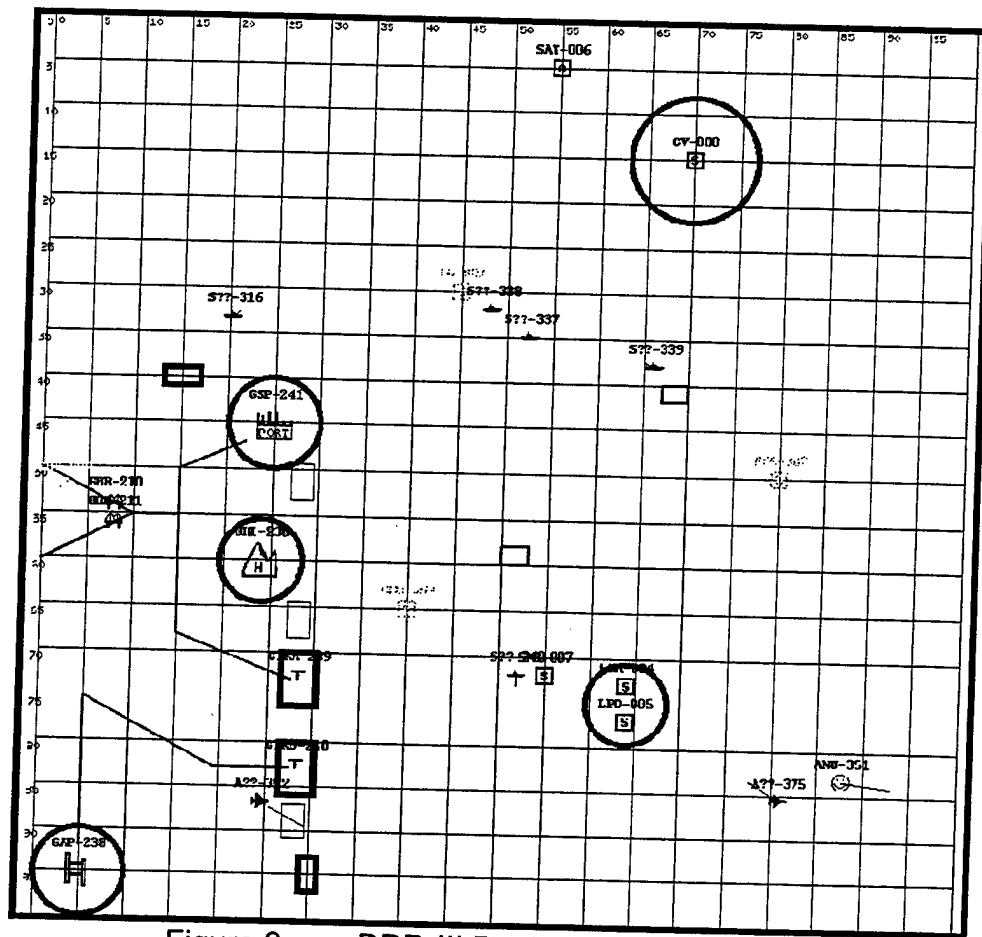
3. Test Subjects

The test subjects were all students from the Naval Postgraduate School. The 36 participants were all military officers chosen from the Systems Management and JC4I Systems curricula. Their ranks ranged from O-3 to O-5 and represented all four branches of the Department of Defense, the Coast Guard, and foreign allies. The test subjects came from various operational and support backgrounds, coming from several line and staff corps communities. Several of the subjects were from various militaries of U.S. foreign allies, including Brazil, Canada, Germany, and Greece. These 36 subjects were divided into six teams of six members each. The foreign officers were distributed

throughout the teams, so that at most a single foreign officer was present in any team. An effort was also made to balance the operational experience of the teams by evenly distributing individuals with operational backgrounds and support backgrounds throughout the teams.

4. Special Equipment

The Distributed Dynamic Decisionmaking III (DDD-III) program, working on eight personal computer (PC) terminals running the Linux operating system, was selected as the simulation engine for the experiment. The simulation was developed by Professor Dave Kleinman at the University of Connecticut and is now supported through cooperation between NPS and the Aptima Corporation. The DDD-III was designed to meet the needs for empirical research in Adaptive Architectures for Joint Command and Control research. The DDD-III is a multi-player, real-time simulation that provides a team of decision makers with an air, sea, and ground environment, a variety of task classes comprising a mission, and controllable platforms that contain sub-platforms, sensors, and weapons. This flexible research tool provides the ability to conduct controlled experiments in a laboratory environment, using problems that are abstractions of real world command and control scenarios. DDD-III allows the experiment designer to translate specific mission requirements found in a real world military environment into a simulation. A screenshot of the simulation is shown in Figure 3. [Ref. 7]



5. Scenario

The teams are set in a scenario where they have just stood up a Joint Task Force comprised of a Carrier Battle Group (CVBG) and an Amphibious Ready Group (ARG) to come to the aid of Country Green, a U.S. ally, which has been attacked by hostile Country Orange. This scenario is described in detail in Appendix D, Mission Brief. In order to drive out enemy Orange forces from Country Green, the JTF must conduct an amphibious operation to secure critical points of entry for follow on U.S. forces. To accomplish this, the JTF is required to complete in sequence seven primary mission tasks, while engaging enemy

forces and defending JTF assets from enemy counter-attack. A graphic depicting these tasks is presented in Figure 4. The seven primary mission tasks are:

1. To land heliborne infantry to secure the hill overlooking the two beaches where the sea-borne portion of the amphibious operation will commence.
2. To secure and hold North Beach to facilitate the landing of infantry from the ARG via Advanced Amphibious Assault Vehicles (AAAV). Infantry landing on North Beach is needed to seize and defend the seaport.
3. To conduct a similar landing on South Beach to provide ground forces necessary to secure the airport.
4. To seize and defend Country Green's international airport with units from the ARG to facilitate the entry of follow on forces via air.
5. To identify and destroy the enemy advance force lead vehicle operating forward of its mobile missile launchers with U.S. Special Operations Forces (SOF).
6. To destroy the bridge used by the lead vehicle with SOF, to prevent the mobile missile launchers from crossing and attacking the ARG.
7. To seize and defend the seaport to allow sealifted follow on forces and Maritime Pre-positioned Ships (MPS) to offload in port.

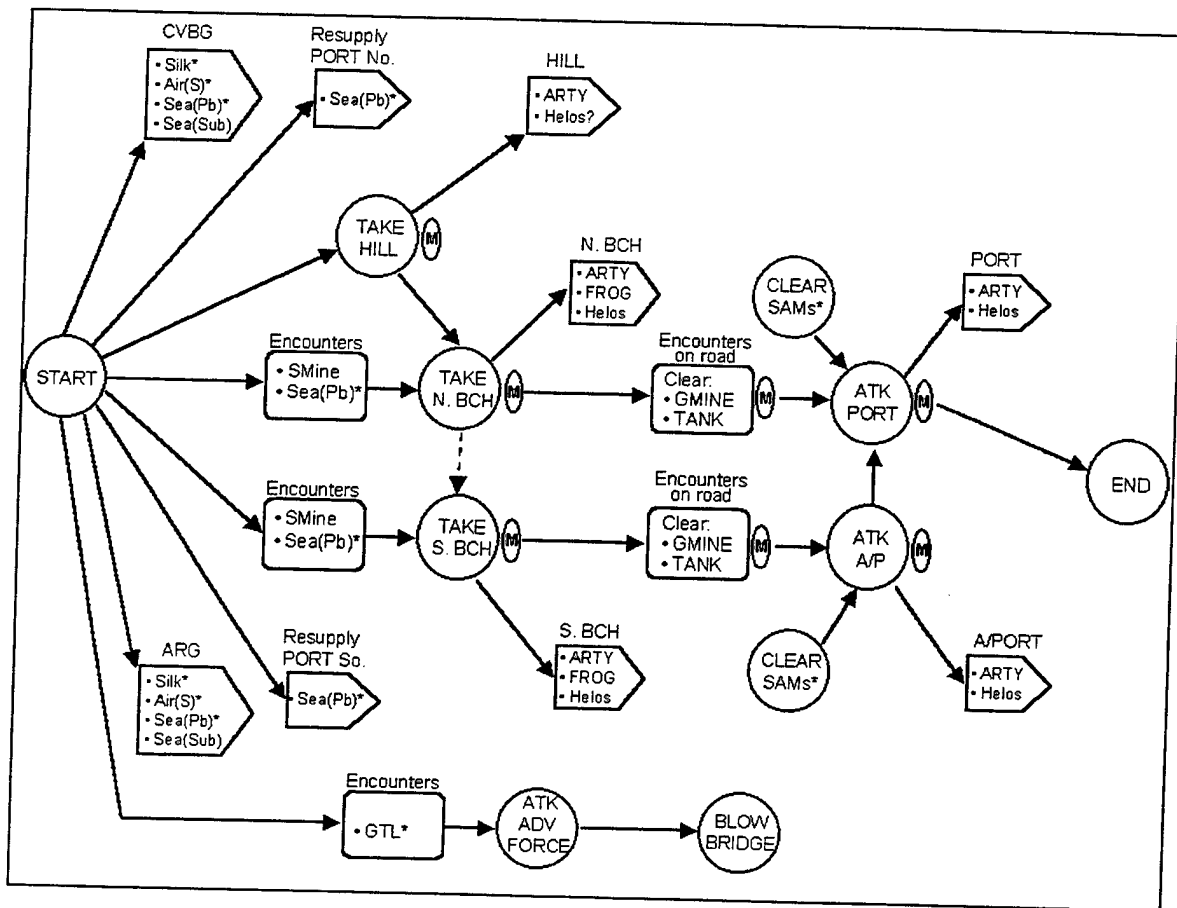


Figure 4. Taskgraph.

6. Schedule of Trials

A2C2 Experiment 7 spanned eight days from February 22, 2000 to March 2, 2000. The first two days consisted of six training sessions, three sessions each day. The schedule is presented in Table 1. The initial training did not require the participants to attend the training sessions with their assigned team members. Rather, the make up of the training sessions was determined solely by the availability of the participants. The remainder of the days consisted of team training and experiment trials to collect data on the six teams. Each day, two trials were conducted; with no team performing two trials in a single day.

Each team performed the simulation using both structures to provide a within-teams comparison. Scheduling was balanced in terms of the order in which teams used each structure to control for an order effect. Each trial consisted of two training runs without the Unanticipated Tasks present in the runs and one recorded data run with Unanticipated Tasks present. An effort was made to create as much time separation as possible between the trials each team had to perform. This was done to reduce team members retention of details from the previous trial. This was intended to minimize the amount of retention on the Unanticipated Tasks from the first trial to the second.

First Trial			Second Trial		
Day 1	Day 2	Day 3	Day 4	Day 5	Day 6
Team 1 Interdepen.	Team 5 Interdepen.	Team 6 Autonomous	Team 1 Autonomous	Team 6 Interdepen.	Team 5 Autonomous
Team 4 Autonomous	Team 3 Interdepen.	Team 2 Autonomous	Team 3 Autonomous	Team 4 Interdepen.	Team 2 Interdepen.

Table 1. Trial Schedule.

C. HYPOTHESES

A total of 12 hypotheses were tested. The first eight examined the performance of the two architectures on the four evaluated task types: Unanticipated Tasks, Multi-node Defensive Tasks, Primary Mission Tasks, and Single-node Defensive Tasks, described later in Section F, Measures. For each task type, the accuracy and latency results from DDD-III were assessed, and a hypothesis was associated with each accuracy and latency result.

Hypotheses number one and two dealt with the Unanticipated Tasks. It was expected that the Interdependent Architecture would benefit from its advantage of practiced coordination, and outperform in latency and accuracy the Autonomous Architecture when it came to dealing with Unanticipated Tasks that were complex and unpredictable.

Hypotheses number three and four involved the accuracy scores and latency of the Multi-node Defensive tasks. The slight performance differences favoring the Interdependent Architecture in these tasks in A2C2 Experiment 4 drove the experimental questions in the current experiment. Like the first two hypotheses, the Interdependent Architecture was predicted to perform better on these tasks due to their mild similarity to Unanticipated Tasks in complexity and unpredictability.

Hypotheses five and six involved the accuracy scores and latency of the Primary Mission Tasks. It was believed that the Autonomous Architecture would outperform the Interdependent Architecture on the seven Primary Mission Tasks as observed in A2C2 Experiment 4. The Autonomous Architecture should perform these tasks better than the Interdependent Architecture because the Autonomous Architecture was specifically optimized to perform these seven Primary Mission Tasks.

Hypotheses seven and eight are associated with the accuracy scores and latency of the Single-node Defensive Tasks. These tasks are the least complex of all the tasks and are moderately unpredictable. The lack of coordination

needed to perform these tasks should not favor either one of the architectures, but based on the results from A2C2 Experiment 4 it was expected that the Autonomous Architecture would perform better.

The next two hypotheses are the first of four hypotheses that do not deal solely with the four evaluated tasks types. Hypotheses nine and ten look at the Mission and Strength Scores, scores produced by the DDD-III simulation at the end of each trial. The Mission Score is an aggregate of how well a team performed all the tasks. The Strength Score is a function a team's performance against enemy attacks. In light of these facts and the results from A2C2 Experiment 4, the optimized Autonomous Architecture is forecasted to have the superior scores.

Hypothesis eleven explored the performance of the ten previously mentioned measures as a whole. This hypothesis examined the two architectures' performances, compared across the range of the all tasks. The focus of this hypothesis was to compare the predicted results to the actual outcomes on each measure.

The twelfth and last hypothesis examined the effect of the order of the trials conducted on the performance of each measure. Despite which architecture was conducted first, it was anticipated that the second architecture conducted would perform better. This hypothesized improvement in performance could be explained by the learning and teamwork gained in the first trial.

D. ASSUMPTIONS

Various assumptions were made concerning the experiment and the data that was collected. The following paragraphs detail these assumptions.

1. Experimental Assumptions

One of the experimental assumptions is that the Unanticipated Tasks were sufficiently well designed in terms of complexity and surprise to produce a performance difference between the two structurally different architectures, if it actually existed. Another important assumption was that all the subjects possessed the basic skill level to operate the simulation.

2. Statistical Assumptions

The paired t test, the nonparametric one-sample Wilcoxon signed rank test, and the distribution-free sign test were the methods used to test the hypotheses. In order to use the paired t test, several assumptions must be made. In this case, each datum is the difference between a team's performance in the Interdependent Architecture and that same team's performance in the Autonomous Architecture (e.g., the difference between Team A's accuracy on the first Unanticipated Task in the Interdependent Architecture and Team A's accuracy on the first Unanticipated Task in the Autonomous Architecture). For each paired t test, the differences are assumed to be independent and normally distributed with the same variance.

The second method used to test the hypotheses was the distribution free or nonparametric one-sample Wilcoxon signed rank test. Generally, the nonparametric tests perform nearly as well as the t tests on populations with

normal distributions and provide performance increases when used on non-normal populations [Ref. 2]. The assumption associated with this test is that the population is a random sample from a continuous and symmetric distribution. Note that if two random variables are from the same continuous distribution, as in the null hypotheses, their differences have a symmetric distribution.

The last test is another nonparametric test, the sign test. In conducting the sign test, performances on the individual tasks were assumed to be independent Bernoulli trials, with the architectures equally likely to perform best on any given task under the null hypothesis. Thus, the number of tasks on which a specific architecture performs best follows a binomial distribution.

E. STATISTICAL DESIGN OF EXPERIMENT

To improve the comparison of the performance between the two architectures, a block (within team) design was used. The experiment design consisted of six blocks of two trials. Each block was made up of two trials, one on each of the two experiment architectures performed by one of the six teams. The within team design was used to control for player experience and other performance variables that could lead to systematic differences between the six teams. In the within team design, each team was tested on each of the two architectures and the results were compared within the team, not against the other teams. Some of the problems with using a within team design are order and transfer effects. In order to eliminate or ameliorate these effects, counterbalancing was employed. This was accomplished by having half of the teams perform one architecture in their first trial, while the remaining teams

performed the other architecture on their first trial. Any benefits of learning and experience between the first and second trial would be applied to both architectures. Finally, due to the limited number of subjects and experimentation time, it was more economical to use a within team design over a between teams design.

F. MEASURES

A2C2 Experiment 7 measures were taken from three sources. The DDD-III derived simulation data, ratings from observers, and ratings from the test subjects. The data involved in this analysis originated from the DDD-III computer-generated data. It consisted of DDD-III task data recorded in the dependent variable files, and the Strength and Mission Scores given at the end of each trial. Two key measures were the accuracy scores and latency on specific tasks. Accuracy scores were the individual scores attained from performing a specific task. These scores were based on the synchronized, simultaneous attack on a target by all nodes involved in the attack, and on the use of the appropriate cumulative asset package required for a successful attack. Task latencies reflected the amount of time it took for a task to be completed once it appeared on the computer screen. Two other measures examined were the final Mission and Strength Scores given at the end of each simulation run. The Mission Score was based on the aggregate performance of all offensive tasks, and the Strength Score was based on the number of enemy penetrations into friendly defenses and collisions with enemy forces. DDD-III tasks were divided into four categories; seven Primary Mission Tasks, six Single-

node Defensive Tasks, three Multi-node Defensive Tasks and four Unanticipated Tasks. The four task categories were then further classified by their degrees of complexity and unpredictability, as shown in Table 2.

Tasks	Complexity/Coordination	Unpredictability
Unanticipated Tasks	High	High
Multi-node Defensive	Moderate	Moderate
Primary Mission	High	Low
Single-node Defensive	Low	Moderate

Table 2. Evaluated Task Characteristics.

The experiment was specifically designed to compare the performance between the two architectures on Unanticipated Tasks. Each Unanticipated Task was designed to be highly complex in required inter-nodal coordination. This was accomplished by requiring two to three nodes to coordinate in order to accomplish each of these tasks. The Unanticipated Tasks were also designed to be highly unpredictable. Unpredictability was integrated into these tasks by not revealing them to the test subjects until the first trial. Though the players were aware of the Unanticipated Tasks at the beginning of the second trial, the Unanticipated Tasks in the second trial were different in placement, timing, and description from those used in the first trial. These tasks were also designed to be distinct from any of the other tasks present in the scenario to prevent the subjects from gaining familiarity by processing those other tasks in previous runs. The four Unanticipated Tasks in each scenario were also categorized into four

classes that were unique enough that practice on one Unanticipated Task would not transfer to another class. The utilization of these four classes of Unanticipated Tasks was necessary to maintain the same level of difficulty across architectures.

Multi-node Defensive tasks, the next category of tasks, were the most similar to the Unanticipated Tasks in complexity and unpredictability. The three Multi-node Tasks examined were; the destruction of Tanks, the elimination of Silkworm Missile sites, and the neutralization Surface to Air Missile (SAM) sites. While the Unanticipated Tasks required the coordination of two to three nodes for accomplishment, Multi-node Defensive Tasks only required two nodes. The Multi-node Defensive tasks were also somewhat unpredictable. During the course of each scenario, the test subjects did not know where and when these tasks would appear, but they had experienced them during the practice trials.

The Primary Mission Tasks were considered to be of high complexity because each required the coordination of three or more assets from multiple nodes to be successfully completed. These tasks were classified as low in unpredictability because the players largely knew when and where these tasks would appear. During the course of the training and trial runs, the scope and location of the seven Primary Mission Tasks never changed. The teams also had to complete the Primary Mission Tasks in a specific order that was constant throughout the trials. It is also important to restate that the Autonomous

Architecture was optimized to conduct this type of task. The seven Primary Mission Tasks, as well as the other tasks are presented in Table 3.

The last set of evaluated measures was the Single-node Defensive Tasks. This group was comprised of six tasks that were rated low in complexity and moderate in unpredictability. The six tasks were to defend against: artillery, Frog missile launchers, hostile fixed-wing aircraft, hostile helicopters, patrol boats, and submarines. These tasks were the easiest to accomplish because only one node was required to attack them, but they were also the most numerous of all the tasks. Single-node Defensive Tasks also did not appear at the same place or time in each scenario, making them somewhat unpredictable.

Unanticipated Tasks	Multi-node Defensive Tasks	Primary Mission Tasks	Single-node Defensive Tasks
UT 1 UT 2 UT 3 UT 4	Silkworm Sites SAM Sites Tanks	Hill North Beach South Beach Airport Lead Vehicle Bridge Seaport	Artillery Frog Launcher Hostile Aircraft Hind Helicopter Patrol Boat Submarine

Table 3. Evaluated Tasks.

Data were also taken from the observers in the form of questionnaires that they filled out during each trial. The observer surveys focused on the performance of each team as they prosecuted each Unanticipated Task and the overall teamwork in performing the entire game scenario.

The third source of data came from questionnaires that the test subjects completed. These questionnaires were similar in scope to the observer survey forms. Data obtained from observers and test subjects were primarily analyzed by the A2C2 research team. These data were not used in this analysis.

G. INSTRUMENTATION

The data collection instrumentation consisted of the dependent variable file compiled by the DDD-III at the end of each trial run. Accuracy scores and latency in each task were extracted from these files for analysis using the statistical analysis package MINITAB. Mission and Strength Scores were manually recorded by observers upon the completion of each trial run and then subjected to MINITAB statistical analysis.

H. TESTING AND PILOT TRIALS

The Lead Team of six students from the Joint C4I curriculum and members of the A2C2 research team at the Naval Postgraduate School tested the early DDD-III scenarios. Pilot trials were run on all training and trial scenarios. In addition to being standard good procedure, the pilot trials were conducted to facilitate the placement and timing of the four Unanticipated Tasks present in each trial. An additional function of the pilot trials was to familiarize the experiment staff on the equipment and processes involved in the conduct of the experiment. The knowledge gained by the experiment staff from the pilot trials would prove to be essential to training the test subjects and the monitoring of each actual trial. The test runs were also necessary to make software and

hardware modifications to the DDD-III configuration, as the program was transitioning from being a Unix-based program to being a Linux-based program.

THIS PAGE INTENTIONALLY LEFT BLANK

III. DATA DESCRIPTION

A. RAW DATA

A raw data file, containing information on the tasks completed during each session, was created automatically by DDD-III after each run. The files contain information on the nodes involved in completing each task, the accuracy achieved for that task, and the time elapsed to conduct the task. An example of a raw data file is included in Appendix F. Mission and Strength Scores were manually recorded by experiment observers after each trial.

B. DATA CODING SCHEME

The measures described in Chapter II, Experiment Design, were automatically collected by DDD-III or manually collected by observers as described above. Task accuracy scores and latency were manually extracted from the dependent variable file and placed into Excel and MINITAB spreadsheets. The data coding scheme in Appendix F details how the data table was coded.

In instances where the one-sample Wilcoxon signed rank test was used the estimated median of differences was used to determine the actual favored result. This test always subtracted the Autonomous Architecture's value from the Interdependent Architecture's value. In the first four hypotheses, the Interdependent Architecture is favored over the Autonomous Architecture. When the Interdependent Architecture's accuracy scores are favored, a positive estimated median of differences is expected, because the smaller accuracy

score is subtracted from the larger accuracy score. A negative estimated median is expected when the Autonomous Architecture's accuracy is favored. The opposite results to those above are expected when dealing with latencies, since lower latency values are favored over higher values.

When using the sign test, the data was coded to conform to MINITAB. Three possible outcomes were possible in this test. Outcomes that favored expected results were given a value of one. Outcomes counter to what was expected were given the value of three, and ties were assigned with the value of two. With these coded values, MINITAB was then used to perform a sign test with the test null median set to two and the alternative set to less than two, since the expected value is one.

C. DATA PROBLEMS

One of the problems with the data was that some of the teams failed to process several of the Unanticipated Tasks. An unaccomplished Unanticipated Task would receive an accuracy and latency of zero. An accuracy score of zero was suitable for data analysis, but a latency of zero was not, since shorter latencies were favored over longer latencies. Therefore, a latency of 450 seconds was assigned to a team for each uncompleted Unanticipated Task. The latency of 450 seconds was chosen because each Unanticipated Task could only be accomplished within a seven and a half minute window from the time it appeared.

This problem also occurred with Primary Mission Tasks. A team failed to accomplish the Lead Vehicle and Bridge Primary Mission tasks and received a zero latency for each task. No minimum time window was assigned to these tasks, so no standard time could be given. To assign the appropriate latencies, the times of the other teams were considered. Taking the other latencies in the similar tasks into account, the latency for the missed Lead Vehicle task was given a time of 1000 seconds and the missed Bridge task was assigned 1500 seconds. Both times were longer than any one team took to perform those individual tasks, thus awarding the proper penalty for not performing those tasks.

D. DATA TABLE

A condensed summary of the data collected by DDD-III for all trials in this experiment is shown in table form in Appendix F.

E. DATA REDUCTION

The applicable data from the dependent variable files was manually extracted from 3.5" disks. This data was then coded as described above and input into Excel and MINITAB spreadsheets. The statistical analysis of the coded data for the measures of interest will be described in the next chapter.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DATA ANALYSIS

Chapter III showed the data that were collected and how those data were reduced before the analysis. This chapter shows the details of that analysis starting with the analysis plan. The analysis methodology is next discussed, followed by the results of testing each hypothesis. Due to the relatively small sample sizes and the exploratory nature of the research, a probability of rejecting the null hypothesis when it is true (Type I error (α)) of 0.1 was selected as the criterion for rejecting all null hypotheses tested.

A. ANALYSIS PLAN

To examine the 12 hypotheses discussed above, the analysis plan called for the use of parametric and nonparametric analyses, namely the paired t test, the one-sample Wilcoxon signed rank test, and the sign test. This required developing measures that could support the analysis and could be extracted from the DDD-III dependent variable files that were saved at the end of each trial. These measures are discussed in Chapter II, Measures. The MINITAB statistical package was selected to perform the analyses.

B. METHODOLOGY

Hypotheses one through ten were first tested using a paired t test of the target measure's means for each architecture. Next, a nonparametric one-sample Wilcoxon signed rank test was used on the difference in a team's two accuracy scores and latencies on each individual task and the difference in their Strength and Mission Scores. After reviewing the data, it was determined that

the accuracy scores did not have a normal distribution. Because the nonparametric one-sample Wilcoxon signed rank test does not require a normal distribution, and it considers both the direction and magnitude of the differences, it was chosen as the alternative.

The one-sample Wilcoxon signed rank test produced an estimated median that was used to determine which architecture performed better. When using this test, accuracy results favoring in the Interdependent Architecture were expected to have a positive estimated median, while those favoring the Autonomous architecture were expected to have a negative estimated median. The opposite of these results were expected for latencies. Latencies favoring the Interdependent architecture were expected to have negative estimated medians and those favoring the Autonomous Architecture to have positive medians. When the results of the one-sample Wilcoxon signed rank test did not clearly favor either architecture, a sign test, another nonparametric test, was tried to see whether the favored architecture could be determined by considering only the direction of the differences, ignoring the magnitude.

Hypothesis 11 and 12 also used the nonparametric sign test. In Hypothesis 11, the sign test was applied to the evaluated measures' means to compare the predicted results to the actual results. Hypothesis 12 dealt with the question whether an order effect existed, and employed the sign test. In this test, instead of inspecting the architectures to determine which one performed better, the order of the trials were examined to determine if a learning effect

existed. First, a sign test was used on each individual measure to test for the presence of an order effect and its statistical significance. Then the sign test was applied over the results of the ten individual sign tests to test for an overall order effect between the two trials.

C. RESULTS OF ANALYSIS

Two hypotheses were initially investigated to answer the basic research question of whether an Interdependent Architecture requiring more nodal coordination would perform better than an Autonomous Architecture requiring less nodal coordination when performing unrehearsed tasks that were high in their degree of complexity and in their level of surprise. Hypotheses three and four were chosen to further explore the results from A2C2 Experiment 4, whose results suggested that the structurally optimized Interdependent Architecture would outperform the Autonomous Architecture on Multi-node Defensive tasks, that were moderately complex and unpredictable. Hypotheses Five through Ten were tested to further examine the results from A2C2 Experiment 4 where the optimized Autonomous Architecture outperformed the Interdependent Architecture. Hypothesis Eleven compared the predicted results of each measure to the actual results. The twelfth hypothesis was developed to test for the presence of a learning effect and to quantify that effect, if present. In the following paragraphs, the first ten hypotheses are examined in turn using a combination of the parametric paired t test and the nonparametric one-sample Wilcoxon signed rank test. The eleventh and twelfth hypotheses employed only

a nonparametric sign test analysis. A short description accompanies each of the results; however, conclusions that may be drawn from the results are deferred until Chapter V, Conclusions.

1. Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in accuracy on the performance of Unanticipated Tasks.

It was expected before the experiment that the Interdependent Architecture would outperform the Autonomous Architecture in the accomplishment of Unanticipated Tasks because when faced with a new, unexpected and complex task, the Interdependent Architecture would be able to benefit from its advantage of practiced coordination.

The sample means for the Unanticipated Task accuracy scores were used as measures in the analysis. The scores were a 90.65 mean score for the Interdependent Architecture and 86.40 for the Autonomous Architecture. As predicted, the Interdependent Architecture scored better in performing Unanticipated Tasks, but only slightly. A paired t test performed on the Unanticipated Task accuracy scores yielded a p-value of 0.295, which is not significant. The result of the paired t test is presented graphically in Figure 5.

UT Accuracy Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	24	90.65	24.05	4.91
Autonomous	24	86.40	27.88	5.69
Difference	24	4.25	38.23	7.80

95% lower bound for mean difference: -9.12

T-Test of mean difference = 0 (vs > 0): T-Value = 0.55

P-Value = 0.295

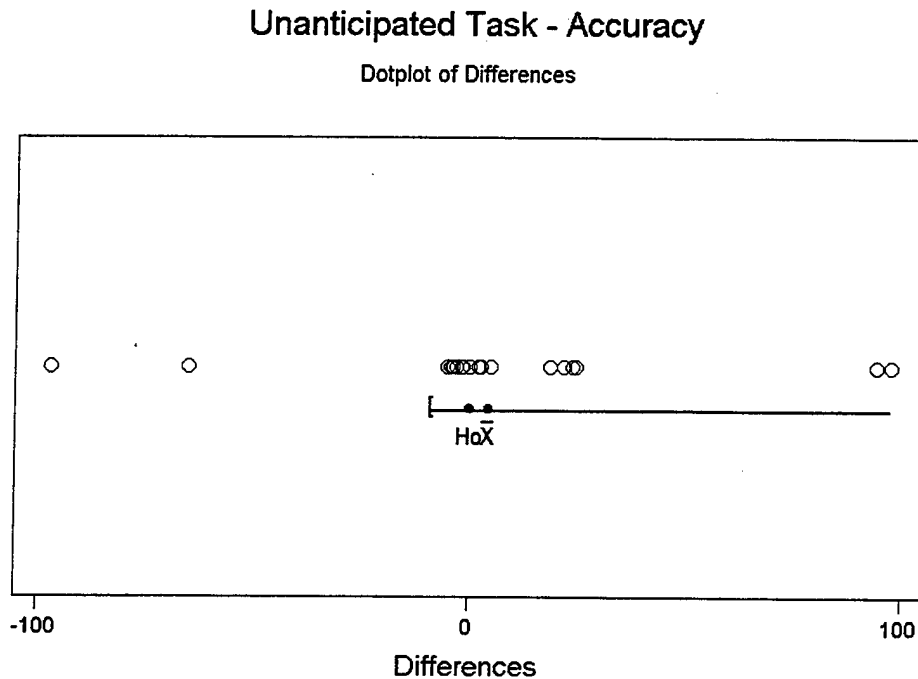


Figure 5. Dotplot of Unanticipated Task Accuracy.

A nonparametric one-sample Wilcoxon signed rank test was then performed on the accuracy scores for each of the Unanticipated Tasks. Unlike the previous test, the one-sample Wilcoxon signed rank test was not based on the mean scores, but rather on the differences of the within-team accuracy performances on each individual task. In these tests, the Autonomous Architecture's value was always subtracted from the Interdependent Architecture's. It was predicted that the Interdependent Architecture would perform better than the Autonomous Architecture, therefore the median of the differences should be a positive value. As expected, the Interdependent Architecture outperformed the Autonomous Architecture as indicated by the

positive estimated median of 0.2175. This result was expected, but the test yielded a statistically insignificant p-value of 0.300.

UT Accuracy Test of median = 0.000000 versus median > 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
UT Accuracy	24	19	108.5	0.300	0.2175

As expected, both test results favored the Interdependent Architecture, but both were statistically insignificant. Thus, there is no strong evidence indicating that a difference exists between the performances of these two architectures on Unanticipated Tasks.

2. Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in latency on the performance of Unanticipated Tasks.

Next, we examined the time elapsed to complete each Unanticipated Task. The lower the latency, the better the team performed. As with accuracy, we expected the latencies for the Interdependent Architecture to be better than those in the Autonomous Architecture. The average latencies were 280.4 seconds for the Interdependent Architecture and 300.4 seconds for the Autonomous Architecture. The difference of 20 seconds favoring the Interdependent Architecture supports the expected outcome, but the results of a paired t test on the latencies yielded a p-value of 0.163, which is not significant. The result of the paired t test is presented graphically in Figure 6.

UT Latency Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	24	280.4	76.3	15.6
Autonomous	24	300.4	72.0	14.7
Difference	24	-19.9	97.5	19.9

95% upper bound for mean difference: 14.2

T-Test of mean difference = 0 (vs < 0): T-Value = -1.00

P-Value = 0.163

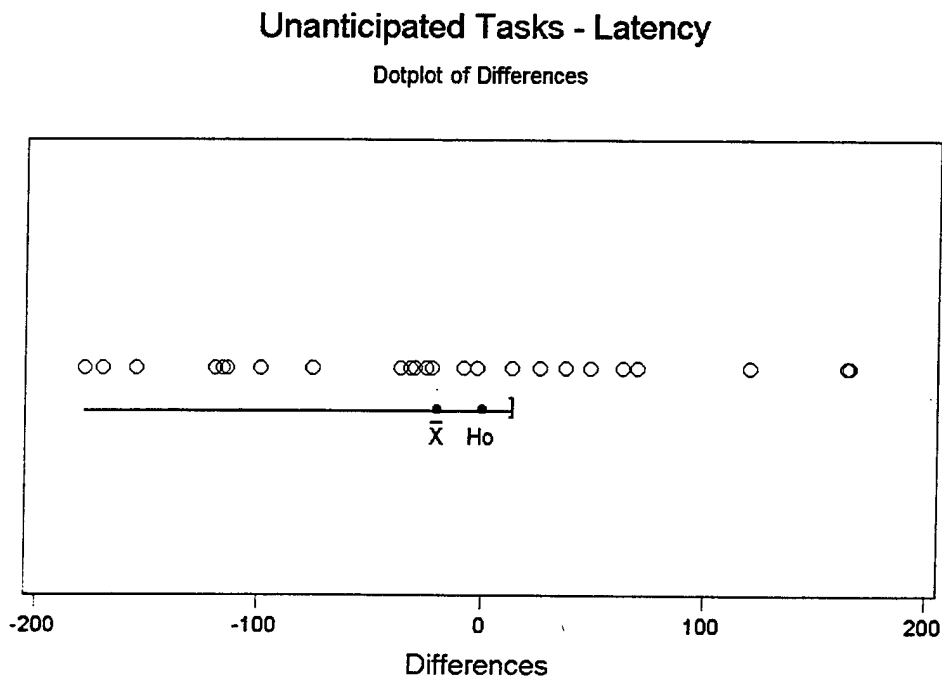


Figure 6. Dotplot of Unanticipated Task Latency.

A nonparametric one-sample Wilcoxon signed rank test was then performed on the latencies of each for the Unanticipated Tasks, which yielded a statistically insignificant p-value of 0.177. As in the previous hypothesis, the Interdependent Architecture was favored. Because lower latencies were favored, the median difference between the architectures should be negative as

the longer Autonomous Architectures latencies are subtracted from the shorter Interdependent Architecture latencies. As expected, the Interdependent Architecture outperformed the Autonomous Architecture as indicated by the negative estimated median of -22.50.

UT Latency Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
UT Latency	24	24	117.0	0.177	-22.50

The two tests on the latencies of Unanticipated Tasks both favored the Interdependent Architecture as expected, but the results were not statistically significant. There is no strong evidence indicating that a difference exists between the latency performances of these two architectures on the Unanticipated Tasks.

3. Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in accuracy on the performance of Multi-node Defensive Tasks.

The next measures examined were the accuracy scores on the Multi-node Defensive Tasks. It was expected that the Interdependent Architecture would perform better than the Autonomous Architecture when performing these tasks because of their similarity to the Unanticipated Tasks. The results from A2C2 Experiment 4 also support this expectation. The Multi-node Tasks are rated as moderately complex due to fact that they cannot be accomplished by a single node. These tasks are also rated as moderately unpredictable because players are unaware of where and when they will appear. Unlike the Unanticipated

Tasks, the Multi-node Defensive Tasks had been previously encountered by the players before the actual trial. They also require fewer nodes are to accomplish. The mean accuracy score for the Interdependent Architecture was 70.50, while the Autonomous Architecture had a mean accuracy score of 82.40. The results favored the Autonomous Architecture, which was counter to what was expected. A paired t test on the accuracy scores yielded an insignificant p-value of 0.104. The result of the paired t test is presented graphically in Figure 7.

MD Accuracy Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	18	70.50	20.29	4.78
Autonomous	18	82.40	20.85	4.91
Difference	18	-11.90	29.37	6.92

95% CI for mean difference: (-26.50, 2.71)

T-Test of mean difference = 0 (vs not = 0): T-Value = -1.72

P-Value = 0.104

Multi-node Defensive Task - Accuracy

Dotplot of Differences

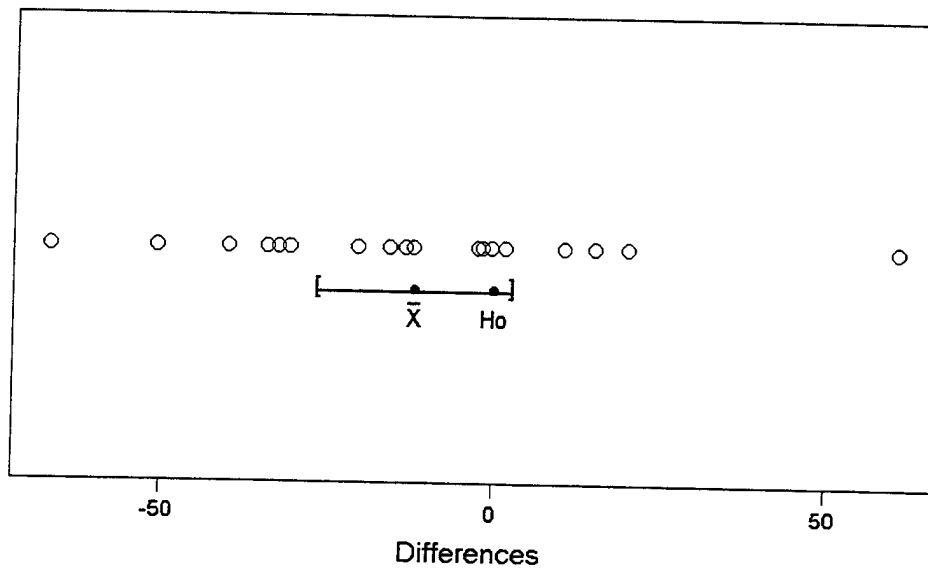


Figure 7. Dotplot of Multi-node Defensive Task Accuracy.

As in the previous analyses, a nonparametric one-sample Wilcoxon signed rank test was then performed on the accuracy scores for each of the 18 Multi-node Defensive Tasks. Because the Interdependent Architecture is predicted to be favored, the differences between architectures should be positive. Contrary to what was expected, the Autonomous Architecture performed much better than the Interdependent Architecture as indicated by the estimated median of -12.45 and a statistically significant p -value of 0.061 .

MD Accu. Test of median = 0.000000 versus median not = 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
MD Accuracy	18	18	42.0	0.061	-12.45

Unexpectedly the results of both the parametric and nonparametric tests favored the Autonomous Architecture. The test involving the nonparametric one-sample Wilcoxon signed rank test also proved to be statistically significant. Thus the evidence show that the Autonomous Architecture outperformed the Interdependent Architecture, given the paired t test p-value of 0.104 coupled with a statistically significant one-sample Wilcoxon signed rank test.

4. Hypothesis: The Interdependent Architecture outperforms the Autonomous Architecture in latency on the performance of Multi-node Defensive Tasks.

The Multi-node Defensive latencies were examined next. As predicted, the Interdependent Architecture scored better with a mean latency of 368.1 seconds, compared to the 390.5 seconds for the Autonomous Architecture. The results of the paired t test on the latencies yielded a statistically insignificant p-value of 0.377. The result of the paired t test is presented graphically in Figure 8.

MD Latency Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	18	368.1	193.4	45.6
Autonomous	18	390.5	263.8	62.2
Difference	18	-22.4	298.7	70.4

95% upper bound for mean difference: 100.1

T-Test of mean difference = 0 (vs < 0): T-Value = -0.32

P-Value = 0.377

Multi-node Defensive Tasks - Latency

Dotplot of Differences

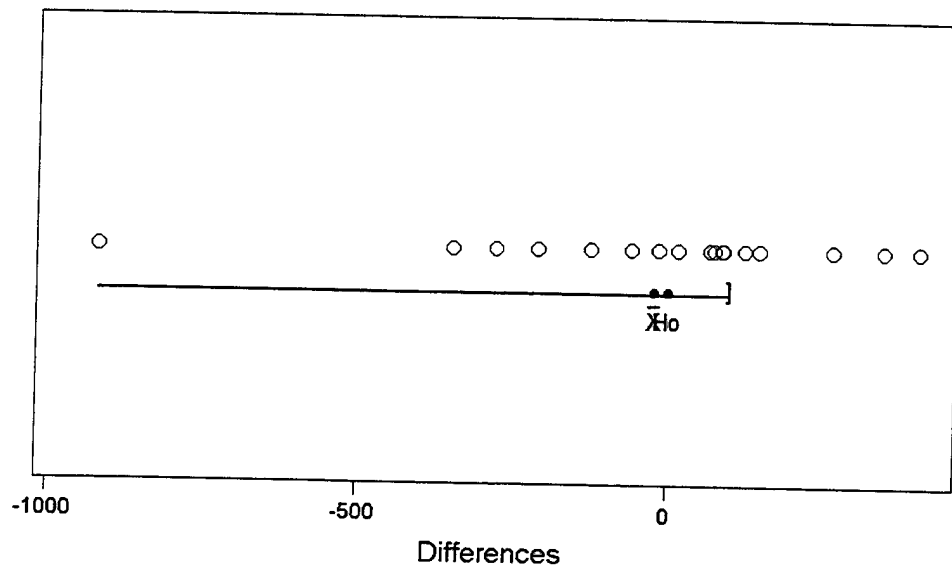


Figure 8. Dotplot of Multi-node Defensive Task Latency.

The nonparametric one-sample Wilcoxon signed rank test also yielded an insignificant result on the latencies for performing the Multi-node Defensive Tasks with a p-value of 0.620. Though statistically insignificant, the Autonomous Architecture outscored the Interdependent Architecture as indicated by the positive estimated median value of 13.47. Because the Interdependent Architecture was predicted to perform faster, the estimated median was expected to be a negative value.

MD Latency Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
MD Latency	18	18	92.0	0.620	13.47

One of the two latency analyses unexpectedly favored the Autonomous Architecture, while the other accuracy score supported the initial expectation. However, neither of the results was statistically significant. Therefore, there is no strong evidence that a difference exists between the two architectures in performance of multi-node defensive tasks with regard to latencies.

5. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in accuracy on the performance of the seven Primary Mission Tasks.

Unlike the two previous categories of measures, Unanticipated Task and Multi-node Defensive Task performance, the performance results on the Primary Mission Tasks were expected to favor the Autonomous Architecture. This architecture was designed through pre-experimental modeling in A2C2 Experiment 4 to be optimized in conducting the seven Primary Mission Tasks, and outperformed the Interdependent Architecture in A2C2 Experiment 4. In A2C2 Experiment 7, the results of Primary Mission Tasks yielded a mean accuracy score of 92.59 for the Interdependent Architecture and 96.78 for the Autonomous Architecture. As expected, the Autonomous Architecture performed better than the Interdependent Architecture. However, the performance of a paired t test on the average scores yielded a p-value of 0.167, which is not significant. The result of the paired t test is presented graphically in Figure 9.

PM Accuracy Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	42	92.59	23.56	3.64
Autonomous	42	96.78	13.04	2.01
Difference	42	-4.19	27.80	4.29

95% upper bound for mean difference: 3.03

T-Test of mean difference = 0 (vs < 0): T-Value = -0.98

P-Value = 0.167

Primary Mission Task - Accuracy

Dotplot of Differences

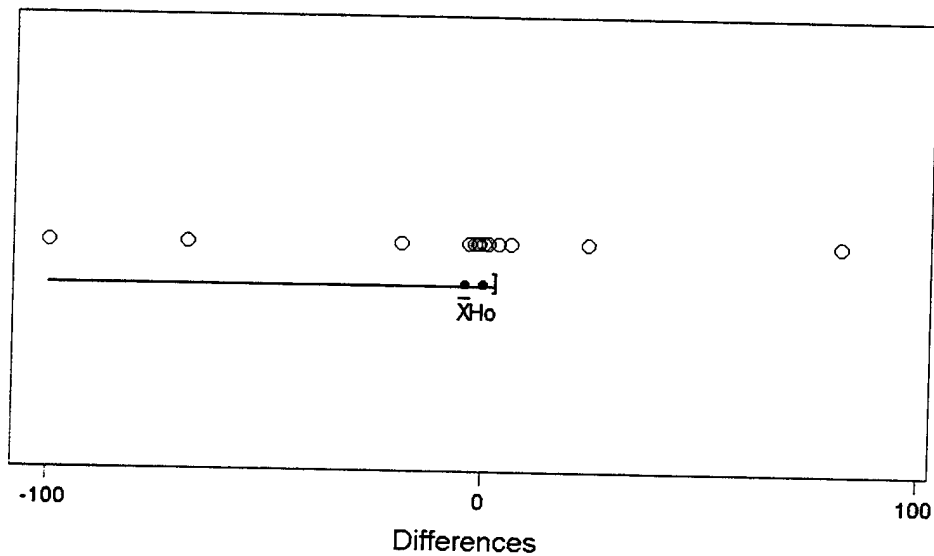


Figure 9. Dotplot of Primary Mission Task Accuracy.

A nonparametric one-sample Wilcoxon signed rank test was then performed on the Primary Mission Task accuracy scores. Each simulation contained seven Primary Mission Tasks, thus the 12 trials produced 42 pairs of Primary Mission Tasks. Of the 42 pairs of instances performed by each architecture, 19 instances resulted in a tie. The large numbers of ties, where no

differences were observed, explains the resulting estimated median of 0.00. Because the Autonomous Architecture was predicted to outperform the Interdependent architecture, a negative estimated median was expected. The resulting estimated median of 0 favors neither architecture. The one-sample Wilcoxon signed rank test yielded a p-value of 0.297, again indicating no statistical significance in the difference.

PM Accuracy Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
PM Accuracy	42	23	120.0	0.297	0.000E+00

To determine which architecture actually performed better, a nonparametric sign test was performed, whose results favored the Autonomous Architecture. In this test, the Autonomous architecture outperformed the Interdependent architecture 14 to 9. The sample sign test also yielded a p-value of 0.2024

PM Accuracy Sign test of median = 2.0 versus > 2.0

	N	Below	Equal	Above	P	Median
no ties	23	9	0	14	0.2024	3.000

Both parametric and nonparametric tests favored the Autonomous Architecture as expected, but both were statistically insignificant. There is no strong evidence indicating that a difference exists between the performances of

these two architectures with respect to accuracy scores when conducting Primary Mission Tasks.

6. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in latency on the performance of the seven Primary Mission Tasks.

Next, the latencies were examined for the Primary Mission Tasks. The latencies for the Autonomous Architecture were expected to be better, for the same reasons as the accuracy scores. The resulting mean latencies supported this; with a 1065.2 second mean latency for the Interdependent Architecture and 1047.4 seconds for the Autonomous Architecture. Performing a paired t test on the latencies yielded a statistically insignificant p-value of 0.237. The result of the paired t test is presented graphically in Figure 10.

PM Latency Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	42	1077	687	106
Autonomous	42	1047	744	115
Difference	42	29.7	265.8	41.0

95% lower bound for mean difference: -39.4
T-Test of mean difference = 0 (vs > 0): T-Value = 0.72
P-Value = 0.237

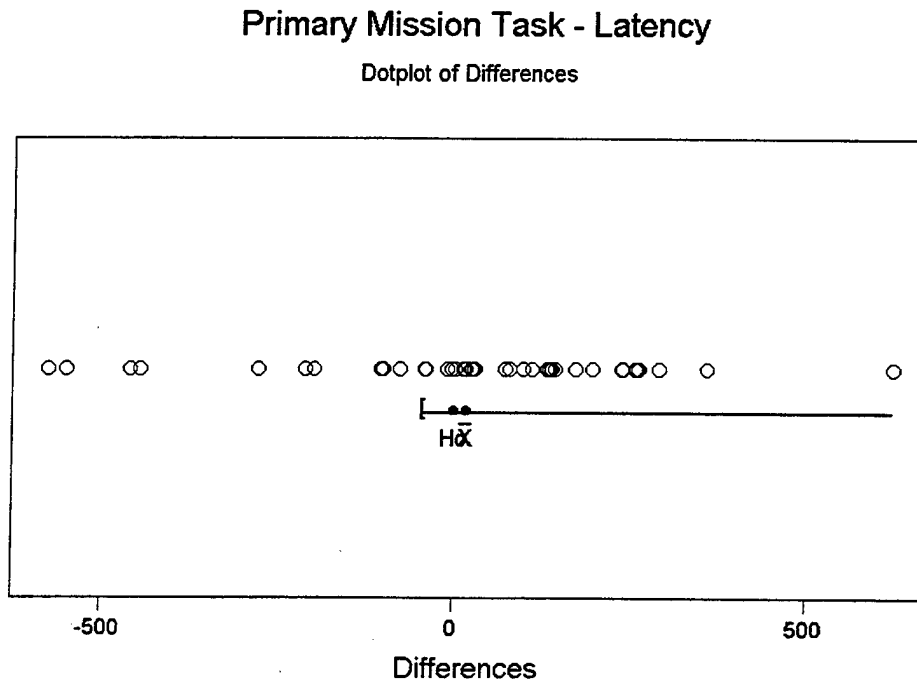


Figure 10. Dotplot of Primary Mission Task Latency.

Each trial contained seven Primary Mission Tasks, thus each architecture performed a total of 42 pairs of Primary Mission Tasks. The performances of the architectures were then compared within each team. A one-sample Wilcoxon signed rank test on the 42 pairs of latencies on the Primary Mission Tasks, produced a result that was both in the expected direction and statistically significant. Because the Autonomous Architecture was predicted to perform better, the estimated median for the latency differences was expected to be positive. In this test, the Autonomous Architecture outperformed the Interdependent Architecture, as indicated by the 33.50 estimated median. The nonparametric one-sample Wilcoxon signed rank test yielded a statistically significant p-value of 0.099.

PM Latency Test of median = 0.000000 versus median > 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
PM Latency	42	42	555.0	0.099	33.50

Analysis of the latency performance on the Primary Mission Tasks indicated that the Autonomous Architecture performed better than the Interdependent Architecture as expected. While the two tests resulted in a favorable result, only the one-sample Wilcoxon signed rank test was statistically significant. The result from the one-sample Wilcoxon signed rank test indicates that the Autonomous Architecture performed better than the Interdependent Architecture in terms of latency on the Primary Mission tasks.

7. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in accuracy on the performance of Single-node Defensive Tasks.

The next set of measures examined were those dealing with the Single-node Defensive Tasks. These tasks resemble the Multi-node Defensive tasks in that they are both rated moderate in predictability, but unlike the Multi-node Tasks, the Single-node Tasks are low in complexity because they require only one node to prosecute these tasks. Based on the results of A2C2 Experiment 4, the Autonomous Architecture was predicted to perform these tasks better. The first two tests examining the Single-node Defensive Tasks involved the accuracy scores. The Autonomous Architecture had a mean accuracy score of 93.15 and the Interdependent Architecture had a mean of 89.21. Performing a paired t test

on the accuracy scores yielded a p-value of 0.060, this is statistically significant.

The result of the paired t test is presented graphically in Figure 11.

SD Accuracy Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	36	89.21	14.06	2.34
Autonomous	36	93.15	11.06	1.84
Difference	36	-3.94	14.88	2.48

95% upper bound for mean difference: 0.25

T-Test of mean difference = 0 (vs < 0): T-Value = -1.59

P-Value = 0.060

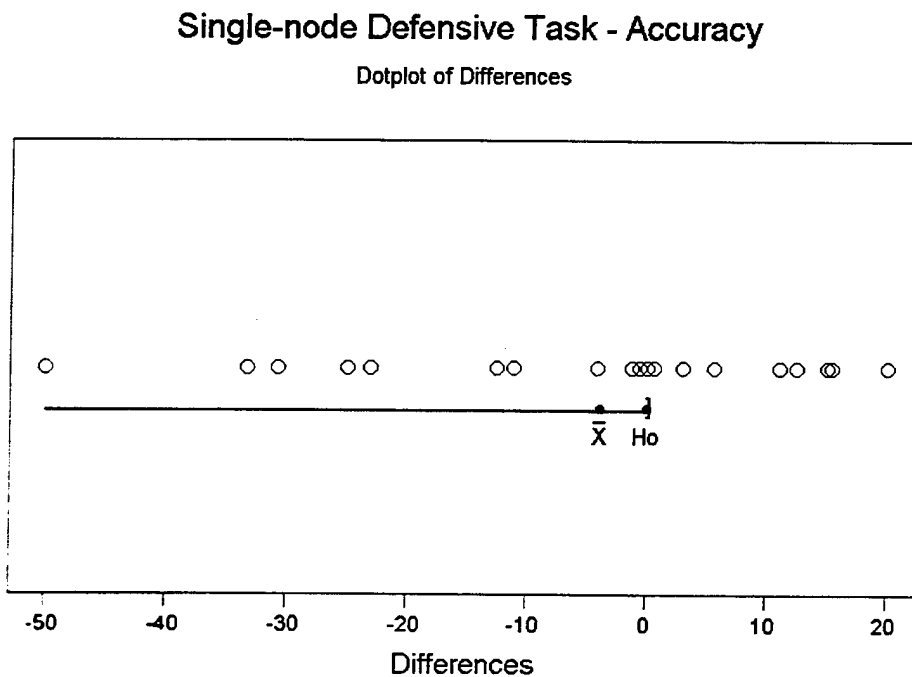


Figure 11. Dotplot of Single-node Defensive Task Accuracy.

A nonparametric one-sample Wilcoxon signed rank test was then performed on the accuracy scores for each of the Single-node Defensive Tasks, which yielded a statistically insignificant p-value of 0.114. As in Hypothesis Five,

this test also produced a result not favoring either architecture as indicated by the estimated median of 0.00.

SD Accu. Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
SD Accuracy	36	19	64.5	0.114	0.000E+00

To determine the actual favored architecture a nonparametric sign test was then performed on the accuracy scores of each of the Single-node Defensive Tasks. This produced a statistically insignificant result favoring the Autonomous Architecture with a score of 11 to 8.

SD Accuracy Sign test of median = 2.000 versus > 2.000

	N	Below	Equal	Above	P	Median
no ties	19	8	0	11	0.3238	3.000

The outcomes of the paired t test and the sign test favored the Autonomous Architecture, and the paired t test was statistically significant. Thus, there is evidence indicating that a difference exists between the performances of these two architectures on Single-node Defensive Tasks.

8. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture in latency on the performance of Single-node Defensive Tasks.

Next, the latencies were examined for the Single-node Defensive Tasks. Analyzing the means first, the Autonomous Architecture had the better mean latency of 178.9 seconds, while the Interdependent Architecture had a mean

latency of 194.9 seconds. Performing a paired t test on the latencies yielded a p-value of 0.154, which was not statistically significant. The result of the paired t test is presented graphically in Figure 12.

SD Latency Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	36	194.9	220.7	36.8
Autonomous	36	178.9	199.1	33.2
Difference	36	16.0	92.8	15.5

95% lower bound for mean difference: -10.1

T-Test of mean difference = 0 (vs > 0): T-Value = 1.04

P-Value = 0.154

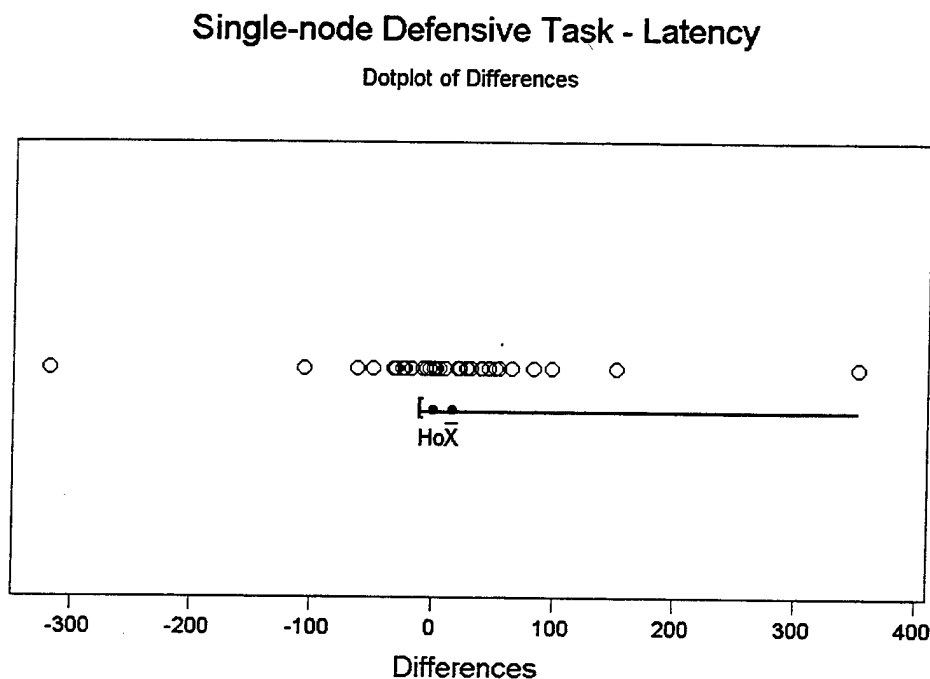


Figure 12. Dotplot of Single-node Defensive Task Latency.

Each scenario contained six Single-node Defensive Tasks, thus the six trials performed on each architecture produced 36 pairs of Single-node

Defensive Task performances. A nonparametric one-sample Wilcoxon signed rank test was then performed on the latencies of each of the 36 pairs Single-node Defensive Tasks, by comparing them by architecture within each team. The Autonomous Architecture was predicted to outperform the Interdependent Architecture in latency, so a positive estimated mean is expected. The one-sample Wilcoxon signed rank test produced a positive estimated mean of 15.69 favoring the Autonomous Architecture and a statistically significant p-value of 0.034.

SD Latency Test of median = 0.000000 versus median > 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
SD Latency	36	36	450.0	0.034	15.69

Similar to the test results in hypothesis seven, the outcomes of both tests favored the Autonomous Architecture, and one test was statistically significant. The result from the one-sample Wilcoxon signed rank test indicates that a difference exists between the performances of these two architectures on Single-Node Defensive Tasks.

9. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture on Mission Scores.

The next measure examined dealt with Mission Scores assigned by DDD-III to the teams at the end of each simulation. The first test examined the difference of the two architectures' Mission Score means. It was expected that the Autonomous Architecture would outperform the Interdependent Architecture

because it was designed to optimize performance when conducting the seven Primary Mission Tasks. As expected, the Autonomous Architecture produced the better mean score of 92.71, compared to 89.83 for the Interdependent Architecture. Though the result met expectations, a paired t test revealed that the difference was not statistically significant, with a p-value of 0.305. The result of the paired t test is presented graphically in Figure 13.

Mission Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	6	89.83	6.11	2.50
Autonomous	6	92.17	5.56	2.27
Difference	6	-2.33	10.52	4.29

95% upper bound for mean difference: 6.32

T-Test of mean difference = 0 (vs < 0): T-Value = -0.54

P-Value = 0.305

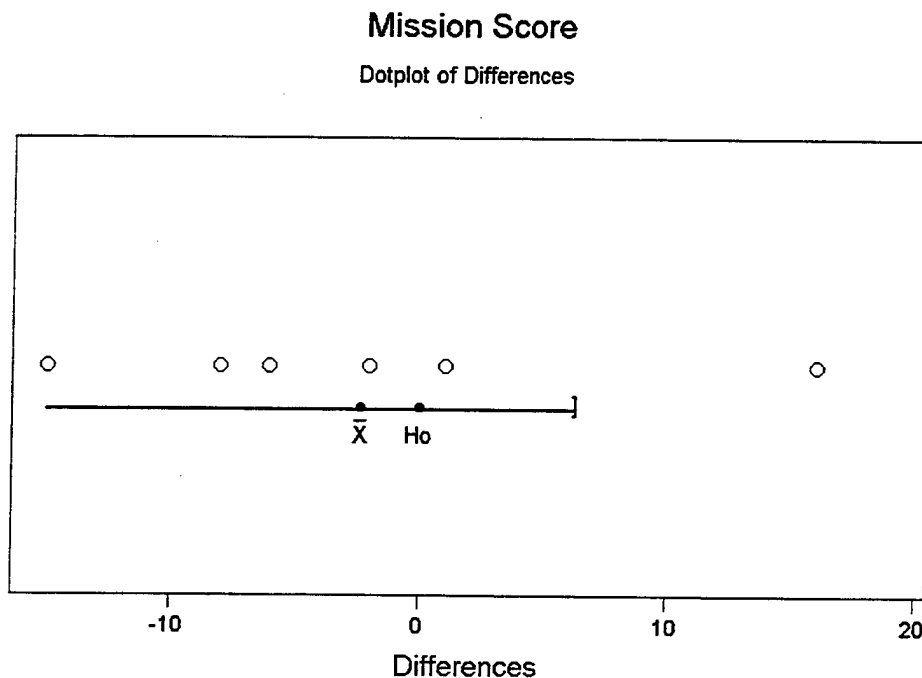


Figure 13. Dotplot of Mission Score.

A nonparametric one-sample Wilcoxon signed rank test was then performed on the six within team differences in Mission Score. Because the Autonomous Architecture was favored, a negative estimated median was expected. The results of the statistical test met expectations and produced a negative estimated median of -3.500, and a p-value of 0.265. The outcome mirrored the paired t test in that both results favored the Autonomous Architecture and were both statistically insignificant. Thus, there is no strong evidence indicating that a difference exists between the performances of these two architectures with respect to Mission Scores.

Mission Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
Mission	6	6	7.0	0.265	-3.500

10. Hypothesis: The Autonomous Architecture outperforms the Interdependent Architecture on Strength Scores.

The second DDD-III generated overall score presented upon the completion of each trial is the Strength Score. While the overall Mission Score reflects offensive performance, the Strength Score is more defensive in nature. The strength score is determined by how well a team manages enemy damage. Once again, the Autonomous Architecture is predicted to prevail over the Interdependent Architecture for the same reasons it was favored in the five previous hypotheses. In fact, the Autonomous Architecture received a mean Strength Score of 89.83, narrowly outperforming the 87.00 mean Strength Score

of the Interdependent Architecture. Further analysis by a paired t test revealed that the slight difference in the score was statistically insignificant. The p-value generated from MINITAB was 0.225. The result of the paired t test is presented graphically in Figure 14.

Strength Paired T for Interdependent - Autonomous

	N	Mean	StDev	SE Mean
Interdependent	6	87.00	6.10	2.49
Autonomous	6	89.83	4.07	1.66
Difference	6	-2.83	8.47	3.46

95% upper bound for mean difference: 4.14

T-Test of mean difference = 0 (vs < 0): T-Value = -0.82

P-Value = 0.225

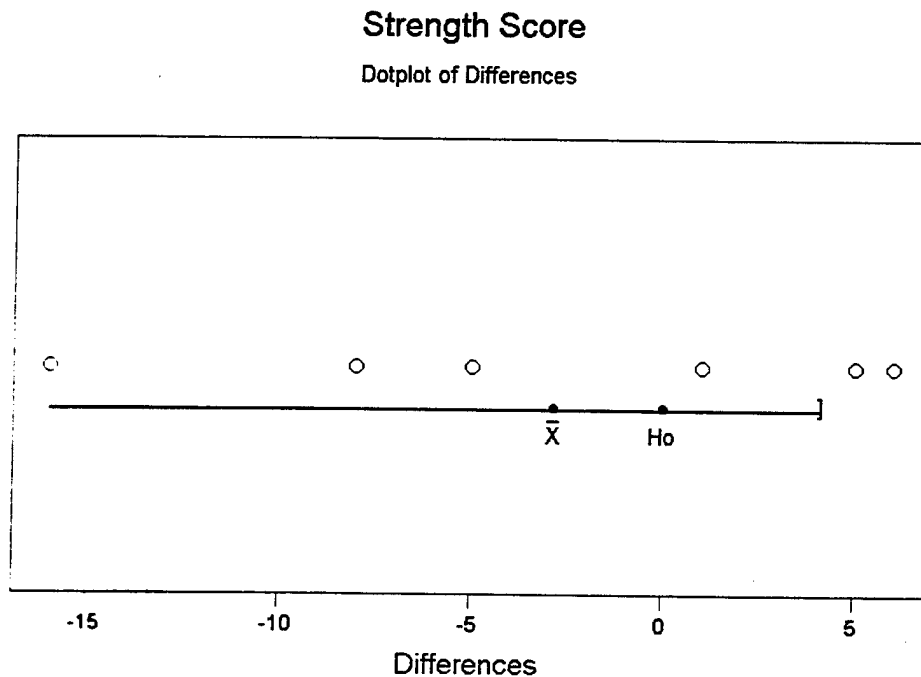


Figure 14. Dotplot of Strength Score.

In addition to analyzing the mean scores, a nonparametric one-sample Wilcoxon signed rank test was then performed by comparing the six within team differences in Strength Score. The Autonomous Architecture was projected to outperform the Interdependent Architecture, so a negative estimated median was expected. A negative estimated mean of -2.000 was generated by MINITAB. The sign test on these scores yielded the statistically insignificant p-value of 0.300.

Strength Test of median = 0.000000 versus median < 0.000000

	N	N for Test	Wilcoxon Statistic	P	Estimated Median
Strength	6	6	7.5	0.300	-2.000

One of the two Strength Score analyses favored the predicted Autonomous Architecture, while the other test resulted in a tie. However, neither of the results was statistically significant. Therefore, there is no strong evidence that a difference exists between the two architectures' Strength Scores.

11. Hypothesis: There is a difference in performance between the two architectures across the measures as a whole.

Analysis of several previous hypotheses determined that the differences in architectural performance were either not significant, or marginally significant. The next step was to determine whether there was a trend in relative performance of the two architectures across measures as predicted by theory. The mean Accuracies and Latencies on each task type and the mean Mission

and Strength Scores shown below in Table 4 were compared between architectures using a nonparametric sign test.

Measure	Interdependent	Autonomous
Unanticipated Tasks – Accuracy	90.65 *	86.4
Unanticipated Tasks – Latency	280.4 sec. *	300.4 sec.
Multi-node Defensive Tasks – Accuracy	70.5	82.4 *
Multi-node Defensive Tasks – Latency	368.1 sec. *	390.5 sec.
Primary Mission Tasks – Accuracy	92.59	96.78 *
Primary Mission Tasks – Latency	1065.2 sec.	1047.4 sec. *
Single-node Defensive Tasks – Accuracy	89.21	93.15 *
Single-node Defensive Tasks – Latency	194.9 sec.	178.9 sec. *
Mission Score	89.83	92.17 *
Strength Score	87	89.83 *

* Indicates the better score of each measure

Table 4. Architecture Performance Comparison.

For each measure, if the difference between means was as predicted by the theory the pair received a score of +1. If the difference between means was opposite the prediction, the pair received a score of -1. There were no ties. The null hypothesis that there was no difference between the two architectures across the measures as a whole (median=0) was tested against the alternative that there was a difference as predicted by theory (median>0). The results are significant (P=0.0107).

Sign test of median = 0.00000 versus > 0.00000

	N	Below	Equal	Above	P	Median
score	10	1	0	9	0.0107	1.000

The previous results assumed that the individual measures were independent, which may not be true in the case of Accuracy scores and the overall Mission Score. Repeating the analysis without the overall Mission Score results in a p-value of 0.0195, which is still significant. There is a trend in relative performance of the two architectures across measures as predicted by theory.

Sign test of median = 0.00000 versus > 0.00000

	N	Below	Equal	Above	P	Median
wo Missi	9	1	0	8	0.0195	1.000

Note that excluding the overall Strength Score from the analysis as well still leads to a significant result (P=0.0352).

Sign test of median = 0.00000 versus > 0.00000

	N	Below	Equal	Above	P	Median
wo Stre	8	1	0	7	0.0352	1.000

The results for Hypothesis Eleven favored the theoretically predicted result, and were statistically significant. Therefore, there is strong evidence that a difference in performance exists between the two architectures when analyzing all measures as whole. Specifically, using the sign test, there is significant support for the theoretically-based expectation that the Autonomous Architecture would be the better performing structure for predictable and moderately predictable tasks; while the Interdependent Architecture would be the better performing architecture for the complex highly unpredictable tasks.

12. Hypothesis: Performance on the second trial is better than the performance on the first trial.

The last hypothesis involves the level of improvement the teams experienced as they transitioned from one trial to the next. Table 5 demonstrates that, of the ten measures examined, only two showed no performance improvement on the second trial. For the Multi-node Defensive Task accuracy, the first trial outperformed the second trial, and for the Mission Score a tie occurred between the two trials. In both these instances, the results of the sign test on the individual measures were statistically insignificant. Of the eight measures that experienced an improvement during the second trial, four were statistically significant with p-values of less than 0.10.

Measure	Better Trial	P-value	Significant
Unanticipated Task – Accuracy	2 nd	.0096	Yes
Unanticipated Task – Latency	2 nd	.1537	No
Multi-node – Accuracy	1 st	.7597	No
Multi-node – Latency	2 nd	.0001	Yes
Primary Mission – Accuracy	2 nd	.3388	No
Primary Mission – Latency	2 nd	.0005	Yes
Single-node – Accuracy	2 nd	.1796	No
Single-node – Latency	2 nd	.0326	Yes
Mission Score	Tie	.6553	No
Strength Score	2 nd	.3438	No

Table 5. Order Effect.

A sign test was then performed over the ten measures to determine if a significant difference existed between the first and second trials. In order to

input this data into the MINTAB statistical analysis tool, numerical values of one, two and three were given to each of the ten measures. Outcomes favoring the first trial were given the value of three, and those favoring the second trial were assigned the value of one. Ties between the two trials were given the value of two. Of the ten instances, eight favored the second trial, one favored the first trial, and one resulted in a tie. The results of the MINTAB analysis generated a statistically significant p-value of 0.0195.

Sign test of median = 2.000 versus < 2.000

	N	Below	Equal	Above	P	Median
Order	10	8	1	1	0.0195	3.000

Strong evidence exists that there was a difference in performance between the first and second trials due to the statistically significant result of the sign test favoring the second trial.

V. CONCLUSIONS

Due to the relatively small sample sizes and the exploratory nature of the research, a probability of Type I error (α) of 0.1 was chosen as the criterion for rejecting all hypotheses tested. Besides presenting the reject/fail to reject results, p-values are also included to report the actual significance observed.

A. HYPOTHESIS RESULTS - INTERPRETATIONS

1. Hypothesis Number One

The first hypothesis compared the architectures' Unanticipated Task accuracy scores. A paired t test on the null hypothesis was used to examine the mean scores and an one-sample Wilcoxon signed rank test was used to analyze the individual scores. As expected, both test results favored the Interdependent Architecture. The paired t test produced a p-value of 0.295 and the one-sample Wilcoxon signed rank test produced a p-value of 0.300. Since both tests were statistically insignificant, we can not conclude that there was a difference in Unanticipated Task Accuracy scores between the two architectures. The null hypothesis that there is no difference in the accuracy scores of the Autonomous Architecture and the Interdependent Architecture on the performance of Unanticipated Tasks is therefore not rejected.

2. Hypothesis Number Two

The second hypothesis compared the architectures' Unanticipated Task latencies. A paired t test on the null hypothesis was used to examine the mean

scores and one-sample Wilcoxon signed rank test was used to analyze the individual scores. As expected, the Interdependent Architecture performed better, but both results were statistically insignificant. Based on the paired t test ($P=0.163$) and the one-sample Wilcoxon signed rank test ($P=0.177$), we cannot conclude that there was a difference in Unanticipated Task latencies between the two architectures. Therefore, the null hypothesis that there is no difference in the latencies of the Autonomous Architecture and the Interdependent Architecture on the performance of Unanticipated Tasks is not rejected.

3. Hypothesis Number Three

The third hypothesis compared the architectures' Multi-node Defensive Task accuracy scores. A paired t test on the null hypothesis was used to examine the mean scores and one-sample Wilcoxon signed rank test was used to analyze the individual scores. Surprisingly, the Autonomous Architecture unexpectedly outperformed the Interdependent Architecture in the conduct of these tasks in both tests. The one-sample Wilcoxon signed rank test ($P=0.061$) proved to be statistically significant. This unexpected outcome may be explained by the fact that the test subjects were exposed to these tasks twice before the first trial run and five times before the second trial run. Though the Multi-node Defensive Tasks appeared unpredictably, their numerous appearances allowed the test subjects to anticipate their appearances. The players' constant exposure to these tasks may have relegated these tasks to resemble the Primary Mission Tasks in predictability. Because the paired t test ($P=0.104$) did not meet

the rejection criteria of a p-value less than 0.10, and because the one-sample Wilcoxon signed rank test ($P=0.061$) did meet the rejection criteria, a probability plot of the accuracy score differences was created to check for a normal distribution. A goodness of fit value of 1.0 or less was designated as the criteria for a normal distribution. Figure 15 indicates a goodness of fit value of 0.805, resulting in a normal distribution.

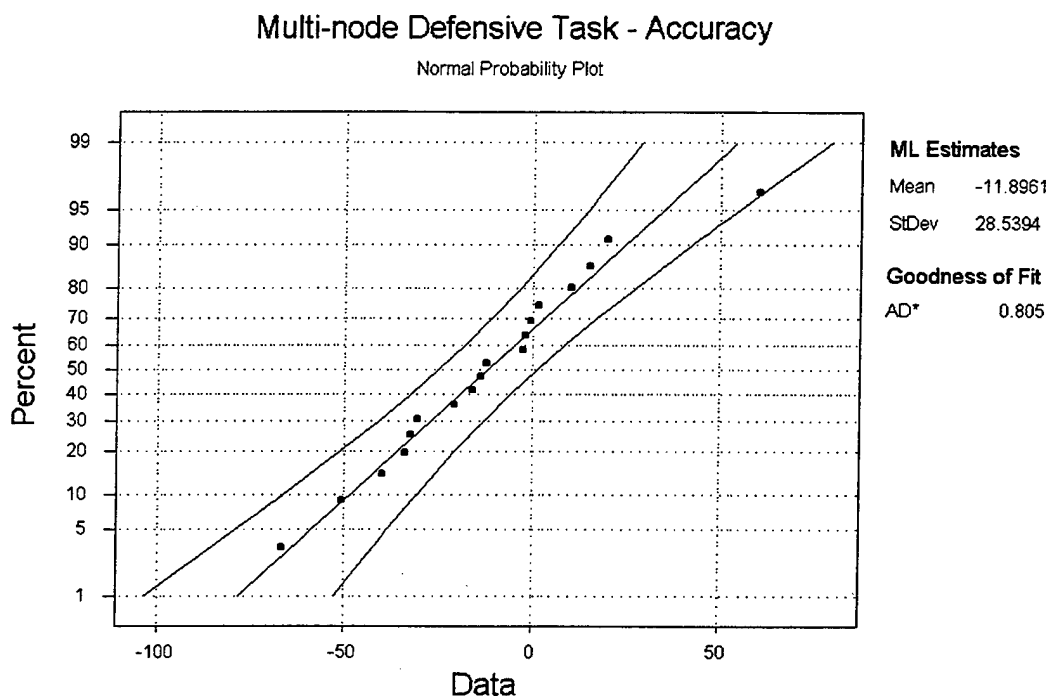


Figure 15. Multi-node Defensive Task Probability Plot.

One of the assumptions of using a paired t test is that the data follows a normal distribution, therefore the result of the paired t test was preferred over the one-sample Wilcoxon signed rank test to test the hypothesis. Based on the statistically insignificant p-value of the paired t test ($P=0.104$) we can conclude that there was no difference in Multi-node Defensive scores between the two

architectures. Therefore, the null hypothesis that there is no difference in the accuracy scores of the Autonomous Architecture and the Interdependent Architecture on the performance of Multi-node Defensive Tasks is not rejected.

4. Hypothesis Number Four

The fourth hypothesis compared the architectures' Multi-node Defensive Task latencies. A paired t test on the null hypothesis was used to examine the mean scores and a one-sample Wilcoxon signed rank test was used to analyze the individual scores. As in Hypothesis Number Three and for the same reasons, the Interdependent Architecture was expected to prevail in these tasks. The results of each test favored a different architecture. The Interdependent Architecture was superior in the paired t test and the Autonomous Architecture performed better in the one-sample Wilcoxon signed rank test. The result of the paired t-test was the only test out of the four tests on Multi-node Defensive tasks that met our expectations. This supports the speculation that these task were encountered so often, that they no longer were moderately unpredictable. Because the paired t test ($P=0.377$) and the one-sample Wilcoxon signed rank test ($P=0.620$) were statistically insignificant and because each test favored a different architecture, we cannot conclude that there was a difference in Multi-node Defensive Task latencies between the two architectures. Therefore, the null hypothesis that there is no difference in the latencies of the Autonomous Architecture and the Interdependent Architecture on the performance of Multi-node Defensive Tasks is not rejected.

5. Hypothesis Number Five

The fifth hypothesis compared the architectures' Primary Mission task accuracy scores. This hypothesis is the first of the remaining hypotheses that expected the Autonomous Architecture to outperform the Interdependent Architecture. As stated previously, the Autonomous Architecture, derived from A2C2 Experiment 4, was designed to optimize performance in the conduct of Primary Mission tasks. Based on the results of A2C2 Experiment 4 it was expected that the Autonomous Architecture would prevail in the following hypotheses. A paired t test on the null hypothesis was used to examine the mean scores and a one-sample Wilcoxon signed rank test was used to analyze the individual scores. As expected the test results from this hypothesis favored the Autonomous Architecture. Due to the statistically insignificant results of the paired t test ($P=0.167$) and one-sample Wilcoxon signed rank test ($P=0.297$), we cannot conclude that there was a difference in Primary Mission Task accuracy scores between the two architectures. Therefore, the null hypothesis that there is no difference in the accuracy scores of the Autonomous Architecture and the Interdependent Architecture on the performance of Primary Mission Tasks is not rejected.

6. Hypothesis Number Six

The sixth hypothesis compared the architectures' Primary Mission Task latencies. The Autonomous Architecture was favored on both tests. A paired t test on the null hypothesis was then used to examine the mean scores and a

one-sample Wilcoxon signed rank test was used to analyze the individual scores. Unsurprisingly, both tests met expectations. The one-sample Wilcoxon signed rank test also resulted in a statistically significant p-value of 0.099, while the paired t test produced a statistically insignificant p-value of 0.237. As in the results of Hypothesis Three, a probability plot of the latency differences was constructed to check for a normal distribution. From the probability plot's goodness of fit value of 1.426 in Figure 16, it was determined that the Primary Mission Task data did not follow a normal distribution.

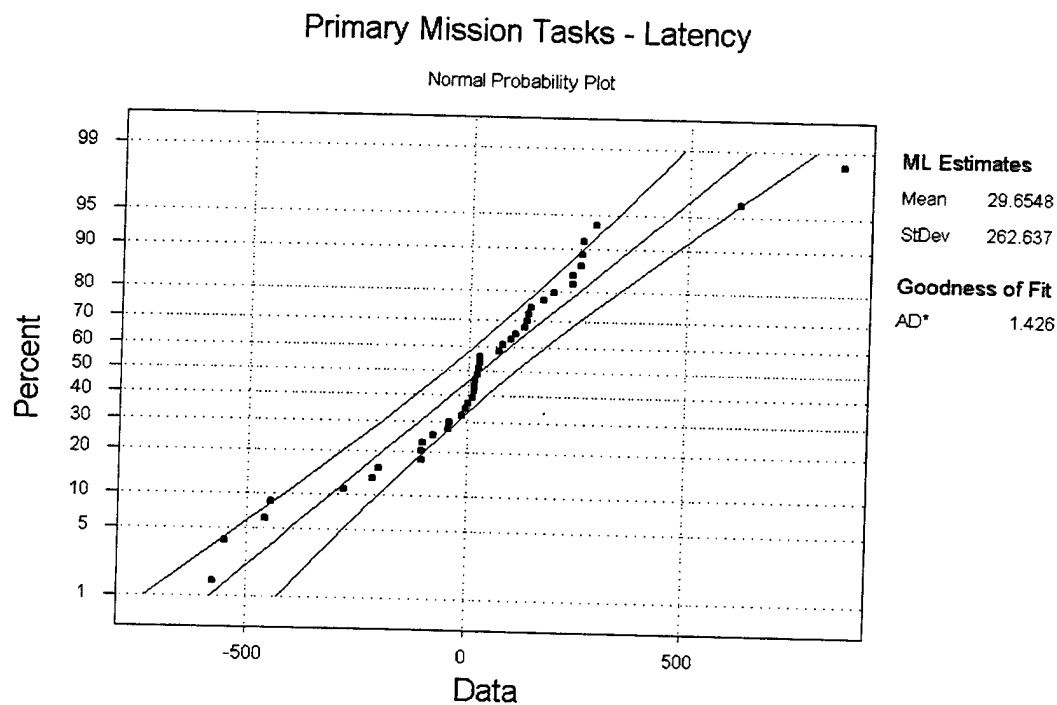


Figure 16. Primary Mission Tasks Probability Plot.

Because the nonparametric one-sample Wilcoxon signed rank test is distribution-free test, it was used as the test for Hypothesis Six. Based on the statistically significant one-sample Wilcoxon signed rank test ($P=0.099$), we can

conclude that there was a difference in Primary Mission Task latencies between the two architectures. Therefore, the null hypothesis that there is no difference in the latencies of the Autonomous Architecture and the Interdependent Architecture on the performance of Primary Mission Tasks is rejected. Rejection of this null hypothesis indicates that the Primary Mission Tasks in terms of latency were better handled by the Autonomous Architecture, whose design, requiring less inter-nodal coordination, was optimized to conduct these types of tasks. The results of Hypothesis Number Six support initial predictions and the results of A2C2 Experiment 4 regarding Primary Mission Tasks.

7. Hypothesis Number Seven

The seventh hypothesis compared the architectures' Single-node Defensive Task accuracy scores. From the results of A2C2 Experiment 4, it was expected that the Autonomous Architecture would prevail in these tasks. A paired t test on the null hypothesis was used to examine the mean scores and one-sample Wilcoxon signed rank test was used to analyze the individual scores. As expected, both scores favored the Autonomous Architecture. Based on the statistically significant paired t test ($P=0.060$) we can conclude that there was a difference in Single-node Defensive Task accuracy scores between the two architectures. The null hypothesis that there is no difference in the accuracy scores of the Autonomous Architecture and the Interdependent Architecture on the performance of Single-node Defensive Tasks is therefore rejected.

8. Hypothesis Number Eight

The eighth hypothesis compared the architectures' Single-node Defensive Task latencies. Like in the previous hypothesis, the Autonomous Architecture is favored on these tests. A paired t test on the null hypothesis was used to examine the mean scores and one-sample Wilcoxon signed rank test was used to analyze the individual scores. The Autonomous Architecture performed better on both tests. The paired t test produced a p-value of 0.154 and the one-sample Wilcoxon signed rank test produced a significant p-value of 0.034. Once again, a probability plot of the latency differences was created to check for a normal distribution. The probability plot's goodness of fit value of 2.648 in Figure 17 indicates that the distribution is non-normal.

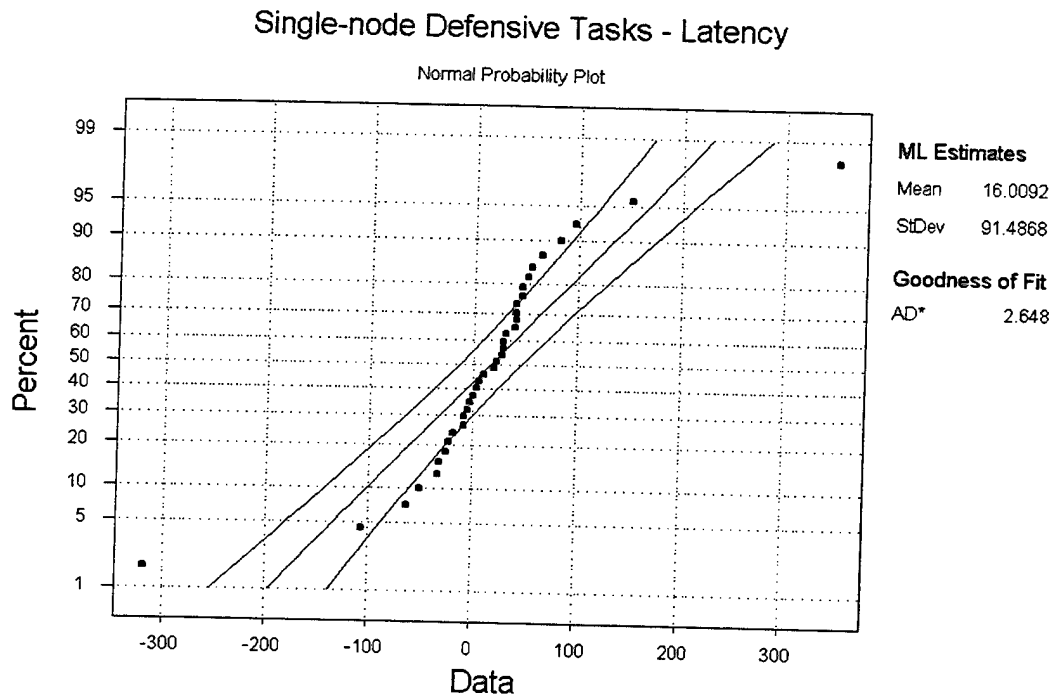


Figure 17. Single-node Defensive Task Probability Plot.

Because the Single-node Defensive Task's latency distribution was not normal, the one-sample Wilcoxon signed rank test was chosen over the paired t test. Based on the statistically significant one-sample Wilcoxon signed rank test ($P=0.034$), we can conclude that there was a difference in Single-node Defensive Task latencies between the two architectures. Therefore, the null hypothesis that there is no difference in the latencies of the Autonomous Architecture and the Interdependent Architecture on the performance of Single-node Defensive Tasks is rejected. Rejection of this null hypothesis indicates the Autonomous Architecture, whose optimized design required less inter-nodal coordination, outperformed in terms of latency the Interdependent Architecture on simple and moderately unpredictable Single-node Defensive Tasks. The results of Hypothesis Number Eight support initial predictions and results of A2C2 Experiment 4 regarding Single-node Defensive Tasks.

9. Hypothesis Number Nine

The ninth hypothesis compared the architectures' Mission Scores. This is the first of two hypotheses that deal with the overall scores generated by DDD-III at the end of each simulation. The Mission Score was determined by how well the teams prosecuted their tasks. Because the majority of these tasks, 13 of the 20, were expected to be performed best by the Autonomous Architecture, it is expected that the higher Mission Scores will be associated with the Autonomous Architecture. Data from A2C2 Experiment 4 supports this expectation. A paired t test on the null hypothesis was used to examine the mean scores and one-

sample Wilcoxon signed rank test was used to analyze the individual scores. The tests produced two statistically insignificant results favoring the Autonomous Architecture. Based on the statistical insignificance of the paired t test ($P=0.610$) and the one-sample Wilcoxon signed rank test ($P=0.344$), we cannot conclude that there was a difference in Mission Scores between the two architectures. The null hypothesis that there is no difference in the Mission Scores of the Autonomous Architecture and the Interdependent Architecture is therefore not rejected.

10. Hypothesis Number Ten

The tenth hypothesis compared the architectures' Strength Scores. The Strength Score is determined by how well a team defends itself from enemy forces. Of the nine defensive tasks, six belonged to the simple and predictable Single-node Defensive tasks. These tasks also appeared much more repeatedly than the other tasks. Because of the Autonomous Architecture was predicted to do better in these tasks, it is reasonable to expect that the better Strength Scores would be coupled with the Autonomous Architecture. As with the Mission Scores, the results from A2C2 Experiment 4 suggests that Strength Scores were better in the Autonomous Architecture. A paired t test on the null hypothesis was used to examine the mean scores and a one-sample Wilcoxon signed rank test was used to analyze the individual scores. The outcomes of the tests were two statistically insignificant results. Both results favored the Autonomous Architecture. Based on the paired t test ($P=0.225$) and the one-sample Wilcoxon

signed rank test ($P=0.300$), we cannot conclude that there was a difference in Strength Scores between the two architectures. Therefore, the null hypothesis that there is no difference in the Strength Scores of the Autonomous Architecture and the Interdependent Architecture is not rejected.

11. Hypothesis Number Eleven

The eleventh hypothesis compared the performance of the two architectures across the range of all ten measures. The performance difference between the architectures was established from the predicted results, because the Interdependent Architecture was expected to be favored on the performance of Unanticipated Tasks and Multi-node Defensive Tasks, while the opposite architecture was expected for the remaining measures. Unlike the previous hypotheses, this hypothesis only employed the use of the sign test. This test compared the predicted and actual results to determine how many outcomes supported the predicted results. The sign test on the null hypothesis produced a statistically significant p-value of 0.0107 in favor of the predicted outcomes. Repeating the analysis by first excluding the overall Mission Score and next the Strength Score resulted in statistically significant p-values of 0.0195 and 0.0352, indicating a trend in relative performance of the two architectures across measures as predicted by theory.

Based on the statistically significant p-values of the initial sign test ($P=0.0107$) and the additional sign tests excluding Mission and Strength Scores, we can conclude that there was a difference in performance between the

Interdependent and Autonomous Architectures. Therefore, the null hypothesis is rejected in favor of the alternative.

This difference in performance between the architectures is visible among the different task types. The results of Hypothesis 11 made it clear that the more unpredictable highly complex Unanticipated Tasks were better performed by the Interdependent Architecture and the very predictable highly complex Primary Mission Tasks were best performed by the Autonomous Architecture. However, the findings on moderately predictable task types, Single-node and Multi-node Defensive Tasks, were mixed. The Autonomous Architecture clearly performed relatively better on low complexity Single-node Defensive Tasks, yet the results on the moderately complex Multi-node Defensive Tasks were divided between the two architectures. The results of Hypothesis 11 support the theory on predictable and unpredictable tasks, but not the results from A2C2 Experiment 4 on complex moderately unpredictable tasks.

12. Hypothesis Number Twelve

The twelfth hypothesis compared the performance of the tasks by the order in which the trials were administered. The aim of this hypothesis was to determine if an order effect existed in the performance of the trials. Experience and learning gained from the first trial should improve performance on the second trials. This hypothesis employed the use of two sign tests. The first test determined whether an order effect existed in each particular task, and the second sign test on the null hypothesis was used to determine if the order effect

was present across the range of measures. The outcome of the first sign test produced one result favoring the first trial, one tie, and the remaining eight results favoring the second trial. Of these results, four were statistically significant. These results were then subjected to a second sign test that produced a statistically significant p-value of 0.020. Based on this p-value we can conclude that an order effect definitely existed between the first and second trials each team performed. Therefore, we reject the null hypothesis that the performance is the same across the order of trial runs.

B. EXPERIMENT SUMMARY

Overall, the results of the hypothesis testing were mixed. Of the hypotheses predicted to favor the Interdependent Architecture, none of the null hypotheses regarding Unanticipated Tasks or Multi-node Defensive Tasks were rejected.

Some evidence did exist to support the results of A2C2 Experiment 4 where the optimized Autonomous Architecture outperformed the Interdependent Architecture on complex and predictable Primary Mission Tasks, simple and somewhat unpredictable Single-node Defensive Tasks, and on Mission and Strength Scores. Out of the six null hypotheses tested that favored the Autonomous Architecture, three were rejected.

Though only three of the ten null hypotheses regarding the individual evaluated measures were rejected, the differences in performance between the architectures may have existed, but were not captured. One of the causes of

this may have been due to a ceiling effect on the task accuracy scores. Task scores ranged from 0 to 100, yet 74 percent of the scores were above 90 and of those scores, 67 percent were perfect scores of 100. The mean for all 240 task scores was 89.67 with a standard deviation of 20.17. The possible ceiling effect is illustrated in Figure 18.

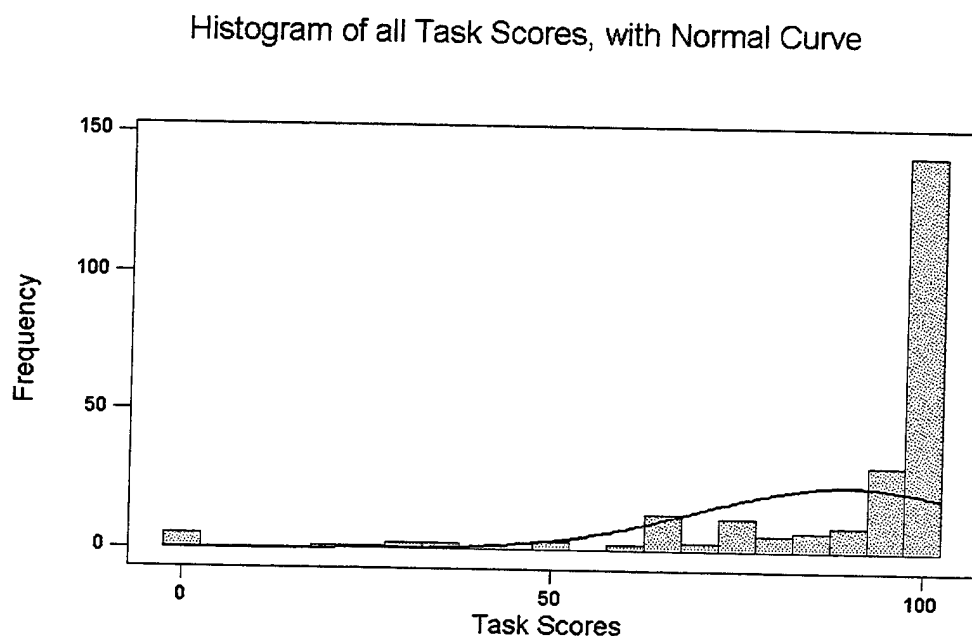


Figure 18. Task Score Histogram.

The best examples of a possible of a ceiling effect are illustrated in the statistical data on three Primary Mission Tasks listed below. All three tasks had mean accuracy scores greater than 98 and had standard deviations of less than 1.7, as shown in Table 6. As a result of this, statistical significance was not achieved on the tests dealing with Primary Mission Task accuracy scores.

Task	Sample	Mean	St. Dev.	Var.
Take the Seaport	12	99.36	0.98	0.95
Take North Beach	12	98.48	1.63	2.65
Take South Beach	12	99.56	1.06	1.13

Table 6. Ceiling Effect on Primary Mission Tasks.

The statistical data solely on the Unanticipated Tasks also showed signs of a possible ceiling effect. Of the 48 scores, 75 percent had scores greater than 95 and 83 percent had accuracy scores greater than 90. The 48 Unanticipated Tasks had a mean accuracy of 88.52 and a standard deviation of 25.84. The histogram in Figure 19 lends evidence to a possible ceiling effect on Unanticipated Tasks.

Histogram of Unanticipated Task Scores, with Normal Curve

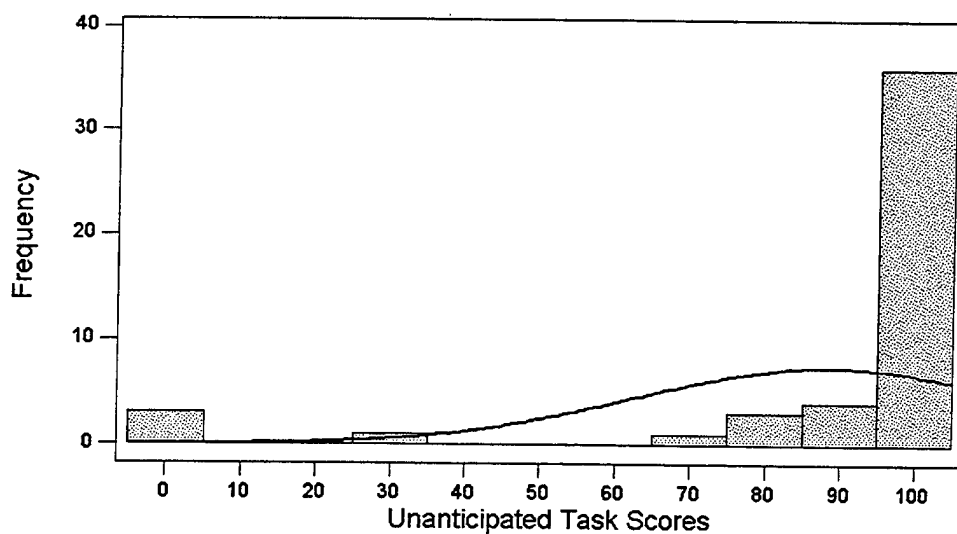


Figure 19. Unanticipated Task Score Histogram.

Although the majority of the test results were statistically insignificant, there are interesting trends within the results that support our initial predictions. The theoretically and experimentally derived predictions were that the Interdependent Architecture would perform better on Unanticipated Tasks, and that the Autonomous Architecture would perform better on the Primary Mission Tasks as examined in Hypothesis 11. Though statistically insignificant, both of the tests dealing with the Unanticipated Tasks favored the Interdependent Architecture, and all the tests where the Autonomous Architecture was predicted to outperform the Interdependent Architecture were in the predicted direction. The differences between the architectures may have been present, but the chosen measures may not have been sensitive enough to capture these differences. Due to the small sample size and the ceiling effect, significant differences were not observed on many of the individual task types. However, the differences almost always went in the direction predicted before the experiment as shown in Table 7. Hypothesis 11 showed that this trend exists and is significant.

On the extreme ends of task predictability, Hypothesis 11 showed that the highly unpredictable tasks were better performed by the Interdependent Architecture and the very predictable tasks were best performed by the Autonomous Architecture. However, the findings on moderately predictable tasks were mixed. The trends of Hypothesis 11 support the results from A2C2 Experiment 4 on predictable tasks and the theory regarding unpredictable tasks,

but there is no strong evidence to support the theory and results from A2C2 Experiment 4 on complex moderately unpredictable tasks, which initially drove the experiment.

Measure	Predictability	Architecture	
		Predicted	Actual Result
Unanticipated Tasks – Accuracy	High	Interdependent	Interdependent
Unanticipated Tasks – Latency	High	Interdependent	Interdependent
Multi-node Def. Tasks – Accuracy	Moderate	Interdependent	Autonomous
Multi-node Def. Tasks – Latency	Moderate	Interdependent	Interdependent
Primary Mission Tasks – Accuracy	Low	Autonomous	Autonomous
Primary Mission Tasks – Latency	Low	Autonomous	Autonomous
Single-node Def. Tasks – Accuracy	Moderate	Autonomous	Autonomous
Single-node Def. Tasks – Latency	Moderate	Autonomous	Autonomous
Mission Score	N/A	Autonomous	Autonomous
Strength Score	N/A	Autonomous	Autonomous

Table 7. Architecture Prediction Results.

THIS PAGE INTENTIONALLY LEFT BLANK

VI. LESSONS LEARNED AND AREAS FOR FUTURE EXPERIMENTATION

This chapter covers the lessons learned during the planning and execution phases of the experiment, as well as recommendations for future follow-on experiments.

A. EXPERIMENT PREPARATION

1. Simulation Modification

It is critical that a fully functioning simulation exists far ahead of the execution of the experiment. As the DDD-III simulation evolved from being a Unix-based program to a Linux based program, it was important to maintain the features and functionality contained in the earlier Unix version of the program. During the course of the pilot trials, it was soon discovered that many enhancements made to the Unix version for previous A2C2 experiments were not transferred to the newer Linux version. Moreover, the newly installed Linux version failed to properly respond to certain commands, and had a tendency to "crash" uncontrollably. Rectifying the glitches took a great deal of time and effort. The time used to tackle the software problems hindered work on the experiment design. Instead of creating software changes to better capture the desired measures, the staff focused on producing a working simulation. With the help of an Aptima technician, who was flown in, a working version of the simulation was created before the start of the experiment.

One of the problems experienced was that of a ceiling effect. To eliminate the ceiling effect, modifications to DDD-III are necessary to design more sensitivity into the calculation of task accuracy, Mission, and Strength Scores. The task accuracy scoring system was almost binomial, in that an incomplete task received a score of zero, and a accomplished task typically received a score of 80 or higher. As shown in the data, this situation makes it difficult to determine the differences between the two architectures.

Another method of countering the ceiling effect is to increase the level of difficulty of the simulation, so that it is more difficult for a team to complete all the tasks in the scenario. This can be accomplished by designing more tasks into the simulation and increasing the pace of the simulation. The maximum amount of time allotted to complete the simulation could also be decreased. The time would have to be standard throughout the teams. In A2C2 Experiment 7, simulation end times were anchored to when the last Primary Mission Task was accomplished.

Though the learning effect was ameliorated through counterbalancing, it still contributed to the ceiling effect. The learning continued to improve scores. A method to tackle this problem would be to design the scenario of the training runs to be distinct from the trial runs, and to also design them to be distinct from one another. By practicing with the same scenarios in the actual trials, teams became proficient in performance by anticipating the chain of events. When the time came to conduct the second trial, the teams had run five similar scenarios.

The current trial run scenarios do not need to be modified, as long as they differ from the training or practice scenarios.

2. Test Subject Preparation

The emphasis of the subject training should be on the individual players, vice the team for several reasons. Training the subjects as individuals would allow experimenters to gauge the skill level of each player. This would be accomplished by having the players perform a test of basic core tasks required for simulation play. Potential subjects would then be required to pass the test before taking part in the trials. This allows subjects competent in the tasks to reduce the amount of time required for instruction, and allows trainers to focus their efforts on players requiring additional attention. Adoption of this process would reduce the overall man-hour training time and ensures that all the subjects have acquired the same minimal level of competency. A web-based self-paced tutorial on DDD-III operation could also be created to reduce the amount of scheduled training time.

3. Experiment Documents

All documents necessary for the conduct of the experiment should be made available to the entire experiment staff for easy access. These documents should be in electronic format, if possible. Having these documents readily available facilitates the easy modification or reproduction of any required material. It is not acceptable to delay the experiment because the required material could not be found or accessed. Because the documents were not

centralized, several versions of a document existed among the experiment staff. These documents should be placed on designated media, such a ZIP disc, or placed in a network account with the proper login information available to the entire staff.

4. Data Recording Equipment

During experiment 7, the audio and video data were recorded on two different types of media. The audio was recorded on cassette tapes and the video data was recorded on VHS tapes. Recording the data solely on VHS tapes would eliminate a step in the data collection process, but more importantly make the data more valuable. Instead of separately reviewing the two types of data, a researcher could simultaneously watch the action taking place and listen to the communications being conducted, without the having to synchronize the audio tape with the VHS tape. This could be achieved by patching the audio portion into the "audio in" jacks of the VCR.

Before the experiment begins all data recording media should be properly labeled by the team conducting the experiment, the type of architecture being run, and whether this was the first or second trial run. This avoids any confusion as to what data is stored on a piece of media. It also avoids the time spent reviewing the data to attempt to determine which team and architecture it belonged to.

5. Information Distribution

During the two weeks allocated for the experiment, experiment training and trials were scheduled in place of class time in other courses. Because of this, test subjects could conceivably have no contact with other experiment participants until their next scheduled evolution. The lack of contact among the participants had the potential for test subjects to not remember experiment evolutions they were scheduled for, and to not be made aware of changes in the schedule. To solve this problem, a website could be created to disseminate pertinent experiment information to staff and test subjects online. Experiment participants would be required to access this website daily.

B. FUTURE EXPERIMENTATION

1. Continuation of Experiment 7

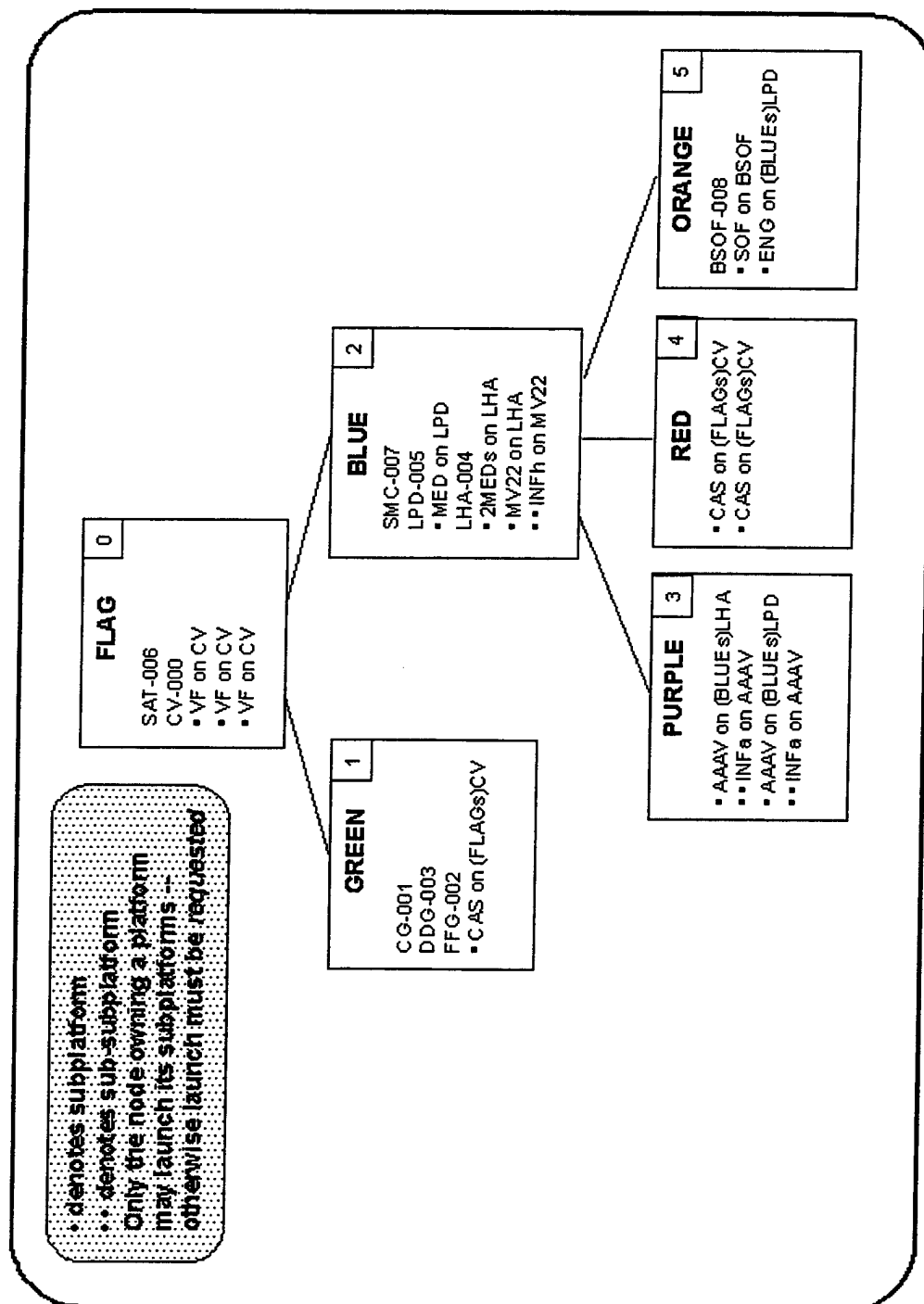
The command structure in A2C2 Experiment 7 was essentially flat. Each node, irrespective of which architecture it was located in, operated in an independent manner. Though the coordination of several nodes was needed to accomplish a task, the decision making process to initiate actions against a task could have been performed at any node in the organization. None of the nodes were subordinate to other nodes, nor were they required to report to other nodes to initiate actions. This was due in part to the lack of an enforced hierarchy typically found in military command and control schemes. Centralized command and control structures inherently require more vertical coordination due to multiple numbers of decision makers in different levels of the command and

control chain needed to initiate actions. The Interdependent Architecture was structurally designed to require more horizontal coordination, but operated without the need for vertical coordination due to the lack of a hierarchy. Further experiments investigating whether a decentralized command structure would perform better than a centralized command structure in performing complex and unpredictable tasks would be worthwhile, especially in the current environment where technological advancements are encouraging a trend towards greater decentralization of command and control architectures.

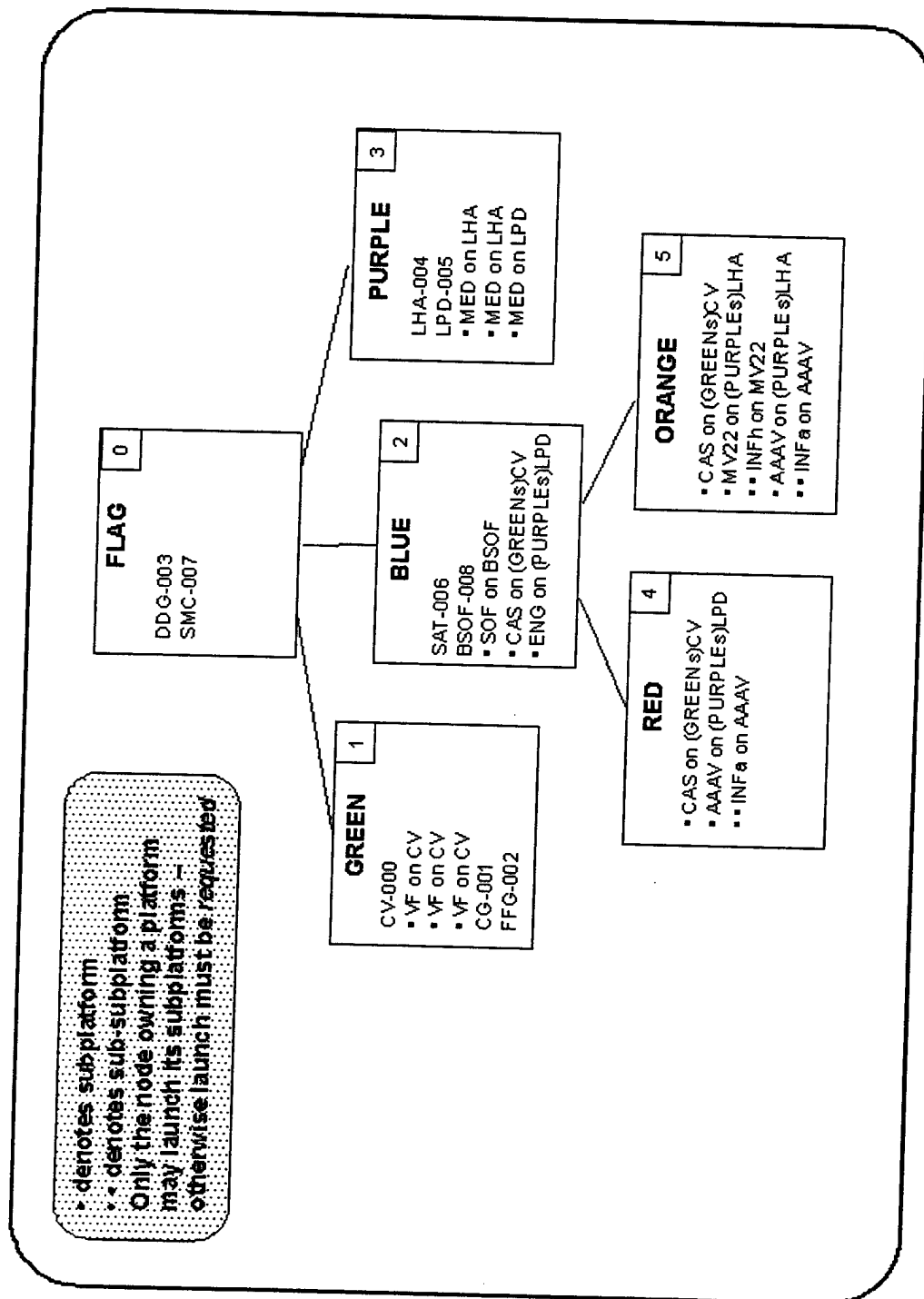
APPENDIX A. ARCHITECTURES

Appendix A contains the architectures used for A2C2 Experiment 7. The Interdependent and Autonomous architectures were used during the practice and trial runs, while the training architecture was used only during the DDD tutorial.

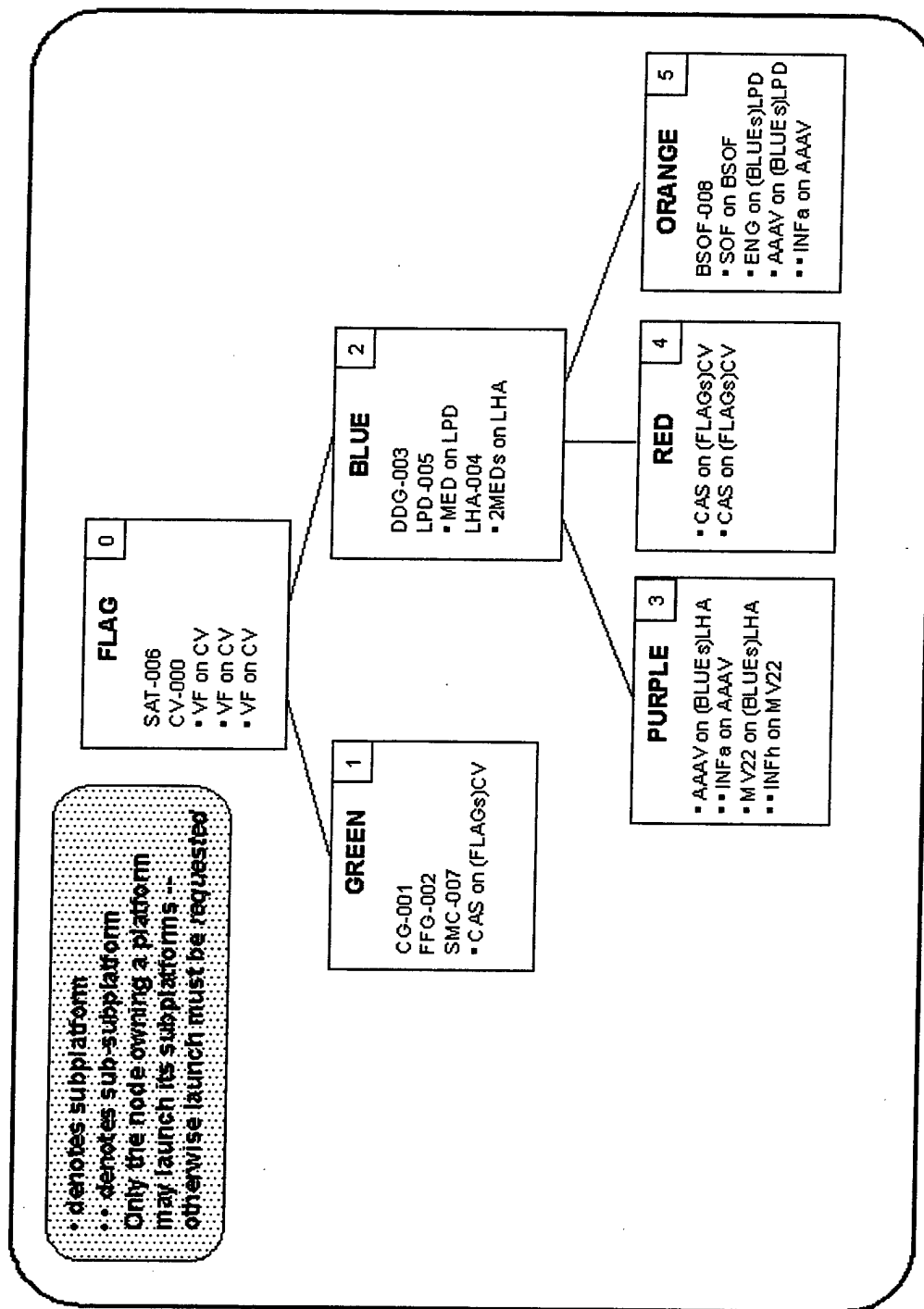
Interdependent Architecture



Autonomous Architecture



Training Architecture



THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. UNANTICIPATED TASK DESCRIPTION

Class Sea 1

Suitable asset packages for this task:

Option 1: {SMC + 2VF + (DDG or CG)}

Option 2: {SMC + DDG + CG}

Cover Stories:

Option 1: Fishing trawler laying mines in shipping lane. Establish air superiority to protect SMC. Destroy enemy trawler and clear the mines.

Option 2: US listening ship has strayed into minefield and is under enemy air and patrol boat attack. Clear mines and assist.

Class Sea 2

Suitable asset packages for this task:

Option 1: {MED + VF + CAS}

Cover Stories:

Option 1: F-14 crew is downed in hostile waters with casualties reported. Protect SAR operation from enemy air and surface actions.

Option 2: SEAL direct action on an oil platform has gone awry. Heavy casualties from ongoing enemy air attack. Send Medivac and destroy oil platform.

Class Ground 1

Suitable asset packages for this task:

Option 1: {INFh + CAS + DDG + SAT}

Option 2: {INFh + 2CAS + SAT}

Cover Stories:

Option 1: Enemy artillery seen digging in with delivery capability to reach ARG. Seek and destroy with infantry supported by precision CAS.

Option 2: Armored convoy with enemy leader spotted in area. Use CAS to immobilize convoy and infantry to capture leader before he can flee.

Class Ground 2

Suitable asset packages for this task:

Option 1: {SOF + MED + (CG or DDG)}

Option 2: {INFh + MED + (CG or DDG)}

Cover Stories:

Option 1: US hostages spotted with reported casualties. Perform a direct action at hostage site and evacuate hostages to CG or DDG.

Option 2: Heavy fighting reported at US embassy. Protect and evacuate all personnel to CG or DDG.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. DDD-III TUTORIAL

The DDD tutorial is the handout used by the trainers and subjects during the DDD training session.

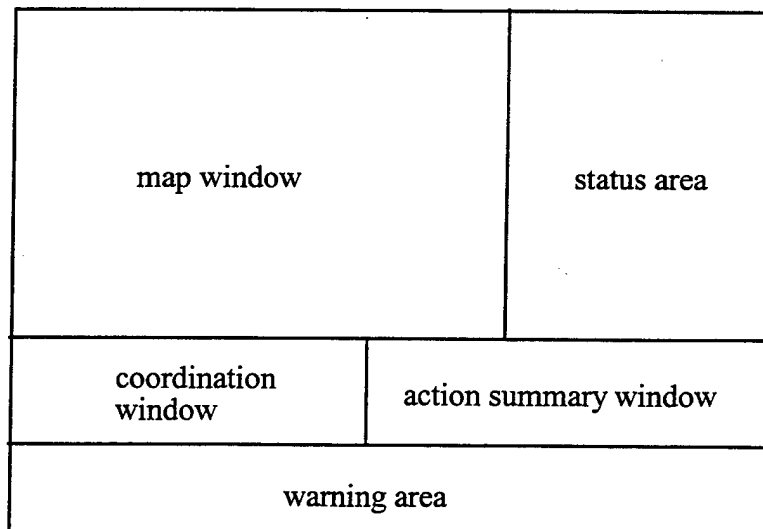
A2C2 Experiment 7 DDD Training Instructions

This handout is organized as follows:

1. DDD Screen Layout
2. Map Icons
3. Friendly Force Actions
4. Task Actions

1. DDD Screen Layout

The screen is partitioned into 5 work areas: map window; status area; coordination window; action summary window; and a warning message area.



a. Status Area.

(1) **Color** of the assets you own/control is the color of the stick man figure. Also listed is the name of the station you are playing (i.e., FLAG, BLUE, etc.). Except for FLAG who corresponds to BLACK, the station name is the same as the color of the assets that the station controls.

- (2) **Time Bar.** When an asset (platform or sub-platform) is selected to perform an action (e.g., launch, attack), a countdown arrow will appear on this bar showing the amount of time remaining to complete this action and which assets are involved. The asset(s) cannot perform any other action until this action is completed.
- (3) **Mission and Strength Counters.** Displays feedback on how well the entire team is doing on the scenario. The mission counter starts at zero and increments as mission tasks are accomplished. The strength counter starts at 100% and decrements any time an action is less than perfect (e.g., attacking with less than the required resources, attacking a neutral, an enemy penetrating a defend zone, etc.)
- (4) **Start/Refresh button.** The Start button is used only at the beginning of a scenario to start the station playing. Once the scenario has begun, the button changes to Refresh. Left clicking on the Refresh button redraws the map and eliminates any undesired traces which may appear.
- (5) **Zoom In.** Allows the user to zoom in for a more detailed look at a particular section of the map. To zoom in, left click on the "Zoom In" button. Then move the cursor over to the map. Click and hold the left mouse button and drag the cursor from upper left to lower right to create a rectangle over the area to be zoomed in. Let go the mouse button.
- (6) **Zoom Out.** Left clicking on this button returns the map to its last view.
- (7) **Cancel.** Left clicking the Cancel button allows the user to abort any about-to-be-issued command to an asset (such as move, pursue or attack).

b. Map window.

On the map, land areas have a beige tint; sea areas are white. Various icons appear on the map that represent friendly and/or enemy forces. Terrain features such as roads are also shown. **Areas within heavy red borders are to defend against possible enemy penetrators/attackers.**

c. Coordination window.

Displays incoming messages that may require some action to be taken by your station, e.g., a request to launch a subplatform owned by someone else but located on a platform that you own. Important messages will also appear in a pop-up window for you to acknowledge (along with an accompanying "beep").

d. Action Summary window.

Summaries of messages or actions performed by your station will appear in this window along with some messages about the status of other friendly platforms.

e. Warning Area.

Displays warning and error messages. A beep will occur along with a warning or error message if any action performed by your station is not allowed (i.e. attempting to attack the enemy when your unit is out of range). Several of these messages will also appear in a "pop-up" window for you to acknowledge.

2. Map Symbolology

The following describes the tasks (i.e., "things to do") in the scenarios. Tasks include taking a hill, attacking an enemy aircraft, clearing a mine, etc. Each specific task has a unique resource requirement which should be met (or exceeded) by the combined capabilities of the attacking assets in order for the attack to be 100% successful. For a specific task, using any one of the recommended asset packages will result in total success

a. Mission tasks

Hill (GHL): The hill is commanding terrain overlooking the North Beach (Beach-A). It is not accessible by road which means that **the only way of accomplishing this mission task is by heliborn infantry**, (launched from their MV22) **in a coordinated attack with other assets.**

Take_beach (GTKA and GTKB): To take a beach, a company of amphibious assault infantry must first be launched/landed onto the beach from their AAV. Then, the beach can be 'attacked' with some combination of infantry, CAS and/or NSFS. It is desirable to take the north beach (GTKA) **before** the south beach (GTKB).

Airport (GAP): The airfield must be taken prior to the port to allow for follow on introduction of friendly forces.

Seaport (GSP): Taking the seaport marks the end of the mission.

Bridge task (GBR): Two bridges are located along roads leading to the west.

1. SOF forces must detect (unknown) ground traffic (**G??**) along these roads.
2. ID of this traffic as either neutral (**GNU**) or an enemy lead-vehicle for a missile convoy (**GTL**) can only be done by the SAT.
3. Once identified, the lead vehicle must be attacked. Next, there is a limited time to destroy the corresponding bridge on that road using the correct combination of assets (to prevent the convoy from crossing it and setting up a missile launcher).

4. Failure to destroy the lead vehicle and bridge in a timely manner will result in missile attacks upon the ARG.
5. Destroying the wrong bridge will result in a penalty (loss) to team strength.

Medevac (GMV): A medevac task *may* appear after assets engage in any ground action (except when dealing with GAT, GFG, GTL or GSWG). A medevac task is accomplished by 'attacking' its icon with a medivac helicopter (MED) within 4-5mins of its appearance. **(NOTE: Medevac helos have a limited endurance of only 8 minutes once they are launched from the ARG ships -- including time to return to base).**

b. Enemy threats/actions

The following section describes enemy tasks that may or may not appear in a scenario. The text also gives the friendly weapon of choice to use against it.

General Symbology: The first letter of a task icon's label will have a letter designation indicating type: "A" denotes air threat; "G" denotes ground threat; and "S" denotes sea threat. Whenever a ? appears in an icon's label/name, the task is a possible threat and must first be identified (typically by SAT or SOF).

- An unidentified task cannot be attacked.
- If identified as a friendly or non-hostile, a task's icon will automatically change to a "smiley-face".

Artillery (GAT): Enemy artillery may pop up at random times. The pieces are stored in reinforced concrete bunkers with the ammunition stored in deep underground bunkers.

- Once detected, a GAT takes approximately 5 minutes to set up before it is able to fire. They target the beaches, hill, and airport.
- Enemy artillery may be suppressed via Naval Surface Fire Support (NSFS), or Close Air Support (CAS). Sufficient NSFS can be provided by the DDG or CG. (The FFG does not have enough armor capability versus these bunkers).
- Once the artillery pieces begin to move toward you, which simulates firing, you will be unable to attack them.

Mines: The enemy has the capability of deploying both land and sea mines. If encountered and moved through by friendly forces, the team's strength will be reduced.

- Seamines (SMS) may only be cleared by the mine clearing ship (SMC).
- Ground_mines (GMN) may only be cleared by the engineering platoon (ENG) sub-platform located on the LPD.
- If an INF unit detects a minefield it will stop and send a warning message to the station that is controlling it. Moving the INF towards the mines

before they have been cleared will cause the mines to detonate, reducing team strength.

Frog Missile sites (GFG): These sites are capable of launching short-range missiles containing chemical munitions.

- The launchers take approximately 5 minutes to set up.
- Suppression must be done through the use of CAS launched from the aircraft carrier or NSFS.

Possible Silkworm Missile Sites (GSW?): The enemy has placed silkworm missile sites (as well as decoys) in three outlined residential areas along the coast.

- The appearance of a potential silkworm site (GWG?) requires visual/image identification via either SAT or SOF prior to attacking the site.
- A decoy site (GDEC) must not be attacked; a confirmed silkworm site (GSWG) should be attacked before it launches its missile.
- The site can only be destroyed by using CAS with precision guided munitions. This requires a coordinated attack with SAT or SOF for precision designation.

Possible SAM Site (ASA?): The enemy has placed Surface-to-Air Missile sites (as well as decoys) around the seaport and airport. The real sites must be identified and destroyed before air support or heli-borne forces can safely be brought into the area.

- The appearance of a potential SAM site (ASA?) requires visual/image identification via either SAT or SOF prior to attacking the site.
- A decoy site (ADEC) must not be attacked; a confirmed SAM site (ASAM) should be attacked.
- The site should be attacked with CAS guided munitions to avoid collateral damage; this requires coordinating precision designation from either the SAT or SOF.
- If any air asset gets too close to a SAM site, a warning will be displayed and the asset will stop. If the air asset again moves toward the SAM site before it has been cleared, the SAM will detonate, simulating an attack on the air asset, and reducing team strength.

Submarines (SSS): The enemy submarines are diesel-powered submarines capable of working the shallow areas near the shore as well as deeper ocean. They can only be destroyed using the FFG platform.

Possible Enemy Ships (S??): The only ships the enemy possesses are fast patrol boats (SPB) that are camouflaged as commercial/neutral shipping (SNU).

- The surface ships require ID by SAT or by very close inspection by a sea platform before they can be attacked.

- Ships can be destroyed by using either the CG, DDG, FFG or CAS aircraft.
- *Enemy* patrol boats are capable of attacking the CV or the ARG; they are also trying to bring in reinforcements at two small coastal areas (1 north, 1 south).
- A neutral/friendly ship must not be attacked.

Helicopter (AHH): The enemy possesses Hind that can attack the beach staging areas and the Hill. The friendly assets capable of destroying them are the CG, DDG and fighters (VF) from the carrier.

Possible Enemy Aircraft (A??): Enemy aircraft (AAS) are intermingled with heavy neutral and commair/neutral traffic (ANU). The enemy air may launch attacks against the CV, ARG or the Hill. Unknown aircraft can be identified by surface ship platforms and/or air platforms. Enemy aircraft may be destroyed by using either the CG, DDG, or fighter aircraft (VF) from the carrier; neutrals must not be attacked.

Tanks (GTNK): Enemy tanks may be encountered anywhere along the roads leading to both the airport and the seaport.

- The tanks can only be 'seen' when within the detection range of a friendly ground asset. If the asset moves out of range the tank icon will disappear.
- Tanks can be destroyed by 2CAS aircraft, or a combination of 1CAS and INF.

c. Friendly (i.e., own) Forces/Assets

Platform: A square icon with the letter A, S or G inside represents a friendly platform. The letter denotes type of medium in which the platform operates (Air, Sea or Ground). Ground assets cannot go on sea, sea assets cannot go on land. The icon's label gives the platform's name (i.e. DDG-003). Platform icons are color-coded to show ownership.

Sub-platform: When launched from its parent platform a **sub-platform** will appear as a *circle* with a letter (A, S or G), and have a label giving its name (i.e. ENG-501). Sub-platform icons are also color coded to show player ownership.

Asset Busy: While a platform or sub-platform is performing some action such as attacking, launching a sub-platform; or when a sub-platform is in return mode, its icon will change to a box with a "x" in it. The asset cannot perform any other action while it is busy. The Time Bar in the status area shows how long the asset will be unavailable/busy.

3. Friendly Force Actions

Platforms (carriers, amphibious ships, etc.) are the major friendly forces in the scenarios. *Sub-platforms* are smaller forces such as aircraft, infantry, engineers, helicopters, etc., that are carried by a platform. [Sub-platforms can also have sub-platforms of their own.] The ownership of any sub-platform may or may not be the same as the owner of the platform it is being carried on.

a. Displaying information about an asset

With the arrow cursor placed on a platform or sub-platform icon:

- 1) Click both right and left mouse buttons simultaneously, OR
- 2) Right click and select "info on asset" from the pull down menu. A pop-up window will list the capabilities/resources, ownership, and status of all sub-platforms (if any) located on the platform. This window is also used to launch or request a launch of a sub-platform.

b. Launching a sub-platform from a platform you own

Only you can launch a sub-platform on a platform that you own. However, if you are NOT the owner of the sub-platform then *a specific request to launch it from the owner must first have been received by you*. To launch:

- 1) Bring up the info on asset window as noted above.
- 2) Left click on the right arrow key in the line corresponding to the sub-platform needed
- 3) Left click OK. Launch will begin and be shown on the countdown bar.
- 4) Repeat for each sub-platform to be launched.
- 5) A platform can launch only one sub-platform at a time.

c. Launching a sub-platform you own from a platform you do NOT own

- 1) Bring up the 'info on asset' window for the platform on which your sub-platform is located. Note that heading on number reads 'REQUEST'
- 2) Left click on the right arrow located in the line corresponding to the sub-platform desired
- 3) Left click on OK.

A message will then be sent to the owner of the platform on which your sub-platform is located requesting that it be launched. It is the responsibility of the platform owner to launch your sub-platform as requested. *This "electronic request" is a software requirement.* Verbal requests should also be used to alert the player of each request.

d. Displaying sensor and/or weapons ranges

- 1) Bring up the 'info on asset' window. Note the options listed in the left portion of window.
- 2) Left click to select whether you are interested in seeing the assets capability versus either Air, Sea or Ground **type** tasks.
- 3) Left clicking the **sensor** option will display four concentric range rings around the asset:
 - The outermost/largest (black) ring is the asset's detection range for tasks of the selected type.
 - The light blue ring is the range at which measurements are obtained on a task. Disregard this range ring as it is not applicable to this experiment.
 - The dark blue ring indicates the asset's positive identification range. For some assets this range is zero.
 - A yellow ring (generally closest to the asset) represents its range of vulnerability.
- 4) Left clicking the **weapons** option will display two concentric range rings. The red ring is the asset's effective weapons range. Also shown is the yellow ring that represents the asset's range of vulnerability.
- 5) To show both sensor and weapons ranges left click "both". To turn off the range rings, left click on "none".
- 6) The asset's ranges may be different for different task **classes** (tanks, patrol boats, mines). Select the task class of interest from the list. If a task class is not specifically listed then the range values are the "default" ones for air, sea or ground types, as appropriate.

IMPORTANT: Different task classes may have different ranges at which they can be detected and/or attacked by the asset in question. First select the task type (air, sea, or ground). Within each type there may be different ranges for different task/target classes.

e. Moving a platform

- 1) Place the cursor on the asset icon and hold down the right mouse button. A menu will appear.
- 2) Select "move". The cursor will change to a cross-hair.
- 3) Position the cross-hair at the place you wish the asset to move to and single click with the left mouse button. The asset will then move to this position. Once it arrives, it will stop until another command to move is given. Any moving object has a line (velocity vector) extending from it. This vector provides an indication of speed and direction of movement.
- 4) A moving asset may be stopped at any time by issuing a "stop" command from the menu.
- 5) Ground assets (INF) are confined to move on road segments or wholly within defend zones (e.g., hill area, beach area, airport, etc.). SOF have All-Terrain style vehicles that are capable of off-road movement—the only ground asset that has this capability.

6) *Take care not to move an asset icon directly on top of another task or platform icon!*

f. Pursuing a task/object

- 1) Place the cursor on the asset icon and hold down the right mouse button. A menu will appear..
- 2) Select "pursue". The cursor will change to a finger.
- 3) Place the finger on the task you desire the asset to pursue and left click. Your asset will then move to intercept and stay with the task until further directed.
- 4) An asset cannot pursue a task if that task is outside of the asset's detection range.

g. Attacking a task

- 1) Place the cursor on the asset icon and hold down the right mouse button. A menu will appear.
- 2) Select "Attack". The cursor will change to an X.
- 3) Place the X on the task to be attacked and left click. If the asset is in range to perform this attack (and has capability to attack), a window will appear that shows the *resources available* on the asset(s) selected to perform the attack and the *resources required* to successfully attack the task.
- 4) Click OK to initiate the attack if you so choose.

Coordinated Attacks

If a single asset does not have enough capabilities to successfully prosecute a task, a coordinated attack involving several assets may be required. IMPORTANT: A coordinated attack will work only if the task is within attack range of all participating assets.

h. Coordinated Attacks using Two or More of your own Platforms

- 1) Hold down the **shift** key on the keyboard and left click on all of the assets to be included in the attack.
- 2) Release the **shift** key and right click on one of the selected assets. The menu will pop up.
- 3) Select "attack". The cursor will change to an X.
- 4) Place the X on the task to be attacked and left click. The attack window that appears will list all of the assets that have been combined for the attack.
- 5) Click OK to initiate the attack if you so choose.

i. Coordinated Attacks among Two or More Players

A simultaneous attack by two or more players may be needed to bring sufficient combat power to bear. These should be coordinated using the voice net. Procedures for multi-

player attacks are the same as for individual attacks. However, when the attack window that lists resources being brought to bear vs. resources required is displayed, **wait!** A verbal countdown should then be initiated by one of the participants. All players contributing to the attack should click OK simultaneously.

4. Task Actions

a. Obtaining latest known information about a task

With the arrow cursor placed on the task icon:

- 1) Click both right and left mouse buttons simultaneously, OR
- 2) Right click and select "info on task" from the pull down menu. A window appears which provides latest information about the task (identified class, neutral/enemy status, attributes, resources required, etc.).
- 3) If the task has been identified a "HELP" button may be clicked to provide the user with any database information relevant to attacking the task.

b. Requesting information about a task from another player

- 1) Right mouse click on the task icon and select "request info" from the pull-down menu.
- 2) A window will open up.
- 3) Select one or more players from whom you wish to obtain information on the task.
- 4) Click OK. A message will then be sent to the person(s) notifying them that this information is requested and by whom.

APPENDIX D. MISSION BRIEF

The Mission Brief is a handout to the test subjects which sets the background for the simulation scenario and instructs players on their initial mission objectives.

MISSION BRIEF

SITUATION:

Country Orange has attacked the friendly nation of Country Green, a U.S. ally, and has seized the northern portion of Country Green including the port of Eastport and the nearby international airfield. Country Green's government has requested U.S. assistance in driving Country Orange's forces from Country Green, and the U.S. has agreed. The CINC plans call for an attack from East to West across the northern portion of Green to drive Orange forces from country Green and reestablish Green's sovereignty. The initial objective is to seize, occupy and defend the Country Green port of Eastport and nearby international airfield to facilitate the insertion of follow-on forces. A Joint Task Force (JTF) has been formed to carry out this mission.

MISSION:

The JTF mission is conduct an amphibious operation to seize, occupy and defend the port of Eastport and the international airfield, as the points of entry for the follow on forces.

YOUR ROLE:

The CINC's planning section has designed two different architectures (organizational structures), each intended to accomplish the mission using the JTFs assets in some optimal manner. To help select the best architecture, the CINC has decided to examine both using a human-in-the-loop, war game-like simulation.

Your team has been asked to play both of the two architectures in simulation mode and then report to the CINC which organizational structure appears best suited to perform the mission and why.

Train hard on your assigned architectures and play them as well as you can so you can send an accurate assessment to the CINC. The quality of your evaluations will have a direct impact on the execution of the real mission.

TASKS COMPRISING THE MISSION:

The tasks to accomplish the JTF mission are presented in chronological order.

1. The Amphibious Ready Group, with embarked Marine amphibious forces, will conduct operations over North and South Beaches. The Marine force is comprised of 2 Advanced Amphibious Assault Vehicle (AAAV)-mounted infantry companies, 1 MV22-mounted heliborne infantry company and other assets (e.g., combat engineers and MEDEVAC assets - see Friendly Asset Sheet). Forces will first secure high terrain and then take both beaches via amphibious assault. Care must be taken, as it is likely that the enemy has laid mines in the water, on the beaches, and on the roads. Mine clearing assets may have to be called in.
2. North beach must be taken first. Prior to taking the North beach, heliborne infantry, in conjunction with additional assets (fire support and anti-armor), will seize the hill overlooking North Beach. This will prevent enemy forces from shelling North Beach. The South Beach should be taken as quickly as possible after the North Beach is taken.
3. The road from North Beach leads to the seaport, and the road from South Beach leads to the airfield. Dismounted AAAV infantry will move up the road leading from each beach. The roads must be cleared of mines and enemy resistance. Due to the swampy nature of the terrain, all ground travel must be on the roads with the exception of the SOF forces, which have all-terrain vehicles.
4. A Special Operations Force (inserted prior to the amphibious operation) and satellite assets (for positive hostile identification) must determine which of two roads in the west leads to an underground Orange mobile missile base. This requires detecting and assessing vehicle traffic along both roads to identify the lead vehicle of an enemy advance force. Once the proper road is identified, it must be cut (by blowing up a bridge) to prevent the enemy mobile missile force from getting within range of friendly forces. The bridge on the other road must not be blown up, since it is needed for friendly traffic.
5. Company-sized armored counterattack forces are believed to be at the seaport and airfield. They must be identified and destroyed.
6. Both the seaport and airfield must be captured and held. The holding action is necessary to prevent the enemy from retaking them. The attack on the airfield has priority and should occur first if they cannot be attacked simultaneously.

OTHER TASKS THAT CAN OCCUR THROUGHOUT THE OPERATION:

1. Performance of MEDEVAC missions to remove wounded.
2. Protection of the battle group from hostile submarines, fast patrol boats and aircraft.
3. Suppression of enemy artillery and Frog missile launchers.

5. Destruction (with guided munitions) of detected enemy SAM sites (most likely around seaport and airfield). The SAM sites are likely to be intermingled with dummy SAM sites.
6. Destruction (with guided munitions) of detected enemy Silkworm sites. Because enemy Silkworm sites have been placed in residential neighborhoods, they must be positively confirmed by SOF or Satellite before attacked.
7. Denial of re-supply ports to enemy patrol boats who are attempting to get to port to load and unload supplies.
8. Execution of other tasks that may appear.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX E. PLAYER REFERENCE HANDOUTS

Player reference handouts were documents that aided the players in the conduct of the experiment. They consisted of the Unanticipated Task Instruction Sheet, Taskgraph, Quick Reference Guide, Friendly Order of Battle, Enemy Order of Battle, and the Primary Mission Reference Guide.

A. UNANTICIPATED TASK INSTRUCTION SHEET

The Unanticipated Task Instruction Sheet was given to the players just prior to the commencement of their trial run. Because the players had no knowledge of Unanticipated Tasks until their first trial run, this sheet was necessary for them to understand how to handle these new tasks.



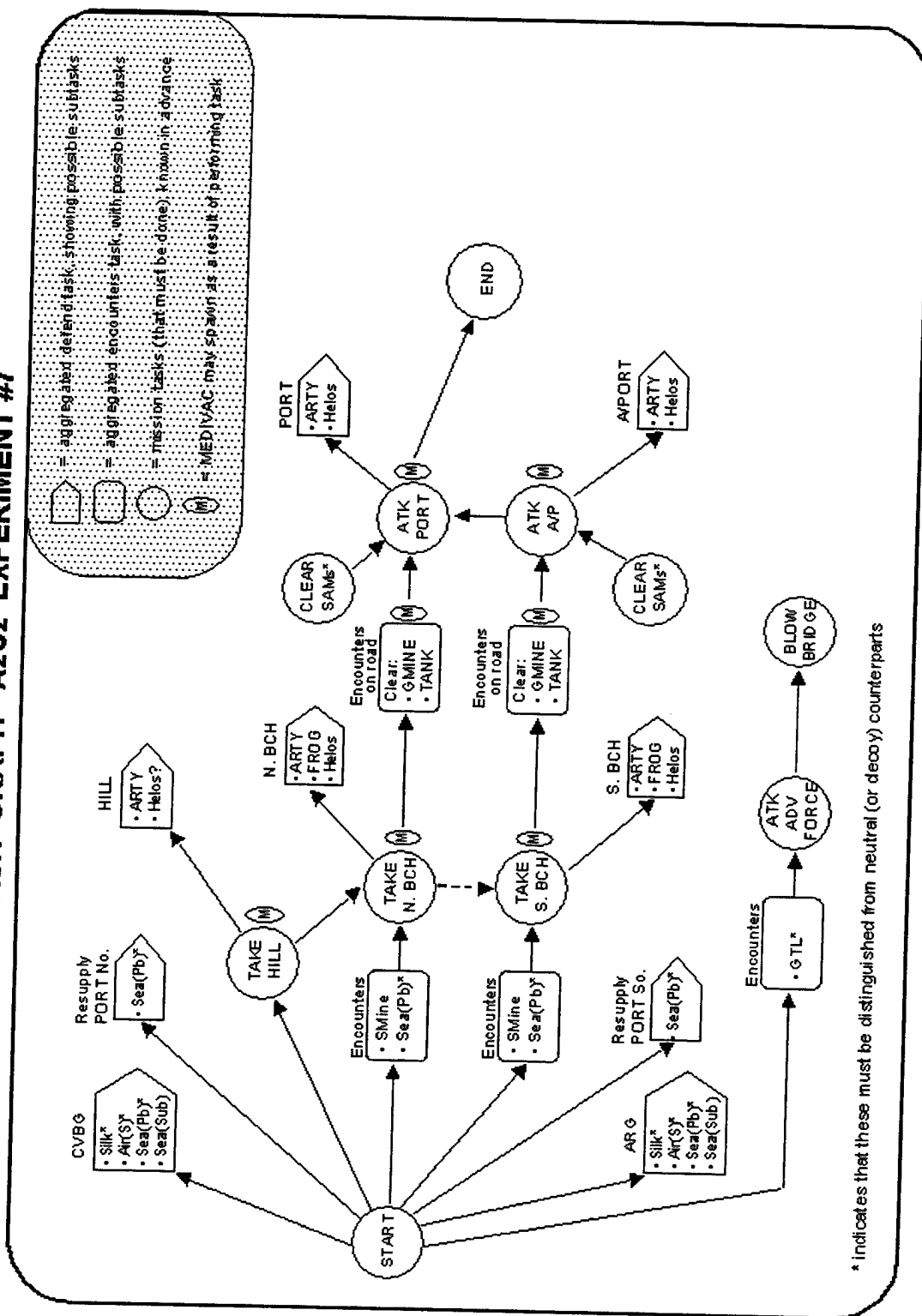
= NEW HIGH PRIORITY TASK REQUIREMENT

This task is equally important to your mission tasks, but does not change the overall, planned mission requirements.

- **This task is time critical.** Each task bearing this new icon will expire seven minutes after it appears.
- To gather INTEL on this task, use SAT to ID it.
- After ID'd, intelligence is available by clicking on New Task icon to get Info on Assets and Help. As with any task, it should be prosecuted as an "attack" on the icon.
- Completion deadline for task is posted as last line of Information on Task Window.

B. TASKGRAPH

TASK GRAPH - A2C2 EXPERIMENT #7



C. QUICK REFERENCE GUIDE

"HELP" BUTTON in TASK INFO WINDOW gives INTEL & ASSET REQMTs

TASKS		SUITABLE FORCE PACKAGES**			
Task name	Symbol	Value	Option 1	Option 2	Option 3
Take hull	GHL	20	INFh+1CAS+DDG	INFh+2CAS	2INFh+1CAS
Take beach	GTR#	20	1INF+1CAS+DDG	1INF+2CAS	2INF+1CAS
Airport	GAP	30	2INF+1CAS		
Seaport	GSP	30	2INF+1CAS	2INF+DDG	2INF+CG
Land vehicle	GTL	15	SOF+1CAS		
Bridge	GBR	15	SOF+1CAS+ENG		
Medivac*	GMV	5	MED		
Sea mines	SMS	10	SMC		
Artillery	GAT	2	DDG	1CAS	
Frogs	GFG	10	DDG	1CAS	
Cnd mines	GMN	5	ENG		
Tank	GTNK	5	1INF+1CAS	2CAS	
Silkworm	GSWG	15	1CAS+SOF	1CAS+SAT	
SAM site	ASAM	10	1CAS+SOF	1CAS+SAT	
Hind helo	AHH	4	1VF	CG	DDG
Hostile a/c	AAS	15	1VF	CG	DDG
Submarine	SSS	25	FFG		
Fast attack	SPB	15	1CAS	CG	DDG or FFG

NOTE: • Items in **bold** need to be positively ID'd (vs. neutrals or decoys)

- GTL can be ID'd by SAT only (@1.5mi)
- GSWG and ASAM can be ID'd by SAT (@2mi) or by SOF (@2mi)
- SPB can be ID'd by SAT (@2mi) or by DDG/FFG/CG (@1.5mi)
- AAS can be ID'd by most ships (@15mi) and a/c (@13mi)

*NOTE: • Attack on any ground (G) target (except GAT, GFG, GTL, GSWG) has possible casualty consequences that may require Medivac.
• Medivacs have a short time window ~ 5mins in which to accomplish

**NOTE: • Any other force packages will result in lower scores or overkill/waste

ASSETS / PLATFORMS		Symbol	Capability
S	destroyer	DDG	
S	frigate	FFG	
S	cruiser	CG	
S	a/c carrier	CV	has VF(3), CAS(3)
S	landing ship	LHA	has AAAV, MV22, MED(2)
S	landing ship	LPD	has AAAV, MED, ENG
A	engineers	ENG	launch from LPD
G	infantry	INFa	launch from AAAV, confined to roads
A	close air	CAS	launch from CV
A	fighters	VF	launch from CV
A	medevac	MED	launch from LHA and/or LPD
S	mine sweeper	SMC	
S	beach lander	AAAV	launch from LHA and/or LPD
A	troop helo	MV22	launch from LHA
A	satellite	SAT	
G	special ops	SOF	launch from BASE, can go off-road
A	SOF's base	BASE	
G	infantry	INFh	must launch from MV22 in order to use

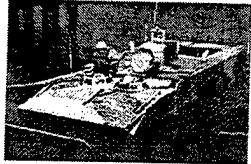
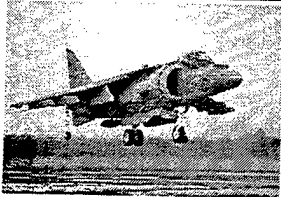
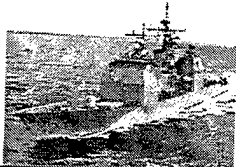
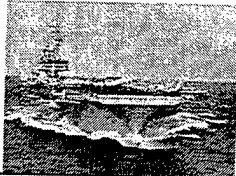
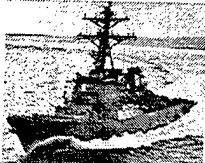
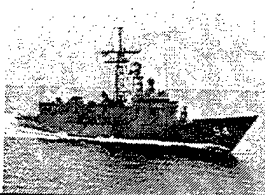
NOTE: • MEDs once launched have < 8mins to complete their mission before automatically returning to LHA/LPD

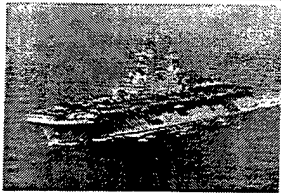
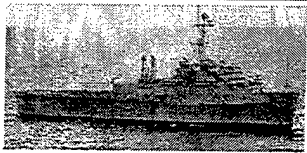
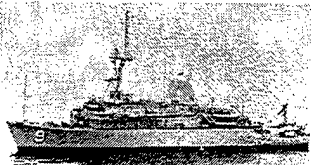
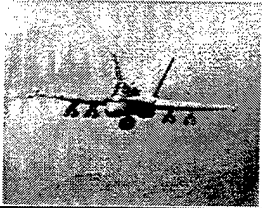
NOTE: • Must launch/unload 1 company of INFa (from AAAVs) at each beach, then move them inland on roads

NOTE: • Asset ranges for specific task classes can be selected from the 'Info on Asset' window

D. FRIENDLY ORDER OF BATTLE

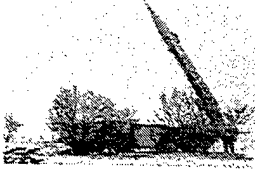
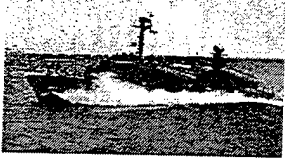
A2C2 Experiment 7 Friendly Order of Battle

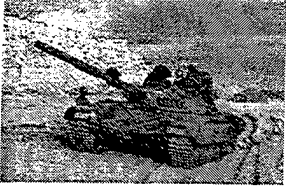
Asset	Picture	Description/Role in DDD
AAAV Advanced amphibious assault vehicle		A vehicle used to carry landing forces ashore
CAS AV-8B Harrier aircraft launched from the CV		Provides Close Air Support for ground forces Capable of vertical and short take off and landing.
CG Guided missile cruiser		A ship equipped with the AEGIS radar system. Primary role is Anti-Air Warfare, defending the carrier.
CV Aircraft carrier		Carries the fighter and CAS assets and the CJTF.
DDG Guided missile destroyer		A ship equipped with two 5"/54 guns. Provides naval surface fire support (NSFS) for the landing forces, and air cover.
ENG Combat engineers		Ground forces which clear landmines, bridges, etc.
FFG Guided missile frigate		Used in Anti-Submarine Warfare (ASW)

INF Infantry companies		Land-based units used to take and hold ground. H(heliborne)- offloaded from MV-22 A(amphibious)- offloaded from AAV
LHA Amphibious assault ship (helicopter carrier)		Used to launch landing forces ashore via AAV and MV-22. Medevac units are also onboard.
LPD Amphibious transport dock		Used to transport and off-load ground forces via AAV. Medevac and engineering units also aboard.
MED Medical evacuation helicopters		Used to perform medical evacuation tasks. Once launched, will return to LHA or LPD in 8 Mins since they have limited endurance time.
MV-22 Tilt-rotor troop transport aircraft		Used to air-transport troops. Capable of forward flight like an airplane and take-offs and landings like a helicopter.
SAT Satellite "beam"		Used by friendly forces in identifying enemy assets: e.g., Silkworm, SAM site, lead vehicle, patrol boats.
SMC Minesweepin g ships		Small ships used in clearing mines at sea.
SOF Special Operations Forces		Elite teams inserted behind enemy lines to observe enemy movement and destroy the bridge. Have on ground precision and off-road travel capability.
VF Fighter aircraft from the CV		Have an air-superiority mission. Charged with defending the carrier battle group.

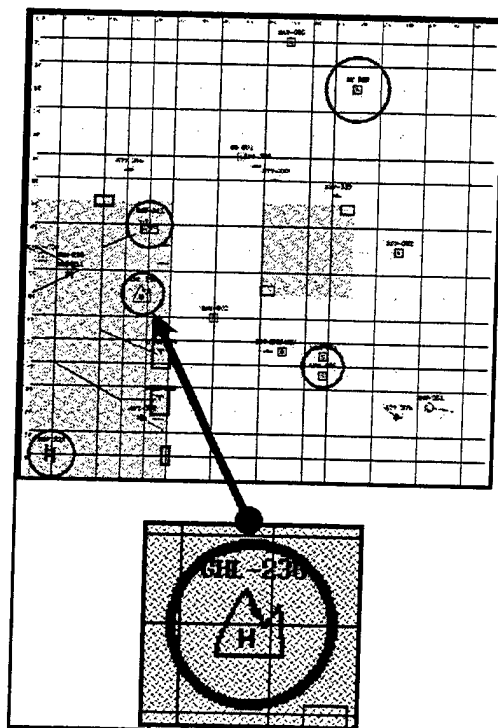
E. ENEMY ORDER OF BATTLE

A2C2 Experiment 7 Enemy Order of Battle

Asset	Picture	Description/Capability
Aircraft Mirage F-1		Enemy has significant air strike capability, and can launch anti-ship missiles from most of its strike aircraft. Countered by assets with significant AAW capability (CG, DDG, VF).
Artillery		Countered by either DDG or 1 CAS.
FROG Scud-like missile launchers		Capable of carrying chemical munitions. Can emerge, set-up, and launch quickly (within 5 minutes) Crews will continue to prepare and launch missiles even if they are being suppressed by NSFS or artillery.
HIND Helicopter		Capable of launching anti-ship missiles. Countered by assets with AAW capability (CG, DDG, and VF)
PBs Fast patrol boats		May fire potent torpedoes. May be loaded with explosives for suicide missions and carry reinforcing troops, supplies. May be camouflaged to resemble commercial craft common in the area and must be identified before they can be engaged.
SAM Surface-to-air missile		Sites must be identified and destroyed before air support or helo-borne forces can be brought in. Once identified, sites must be struck with guided munitions.
Silkworm Silkworm missile		Silkworm launchers placed in residential neighborhoods so must be identified by SAT or SOF before being targeted. Strikes must be made with precision guided munitions (CAS)
Submarines Alpha-class		Pose considerable threat to U.S. Naval fleet. Must be detected and destroyed.

Tanks T-72		May be encountered along the road between airfield and port. Can only be seen when within detection range of friendly ground forces. Can be destroyed by 2 CAS aircraft or a combination of 1 CAS and 1 infantry.
Unknown		Unidentified enemy forces

F. PRIMARY MISSION TASK REFERENCE GUIDE



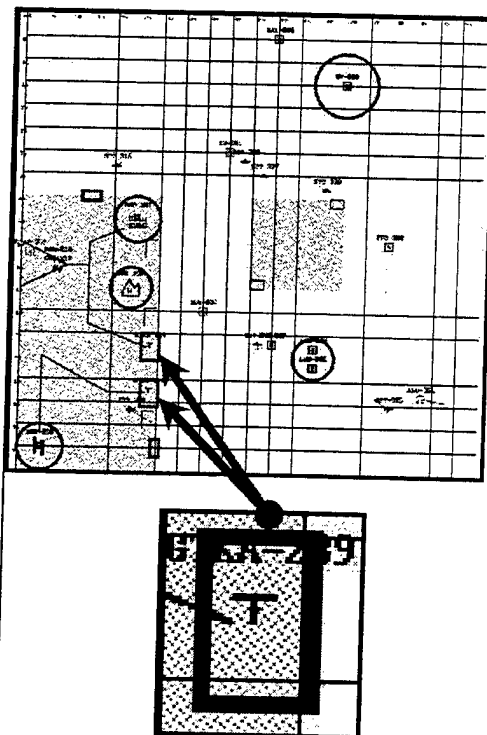
Task 1: The Hill

Prerequisites:

None

Force Packages:

1. (1) INFh + (1) CAS + (1) DDG
2. (1) INFh + (2) CAS



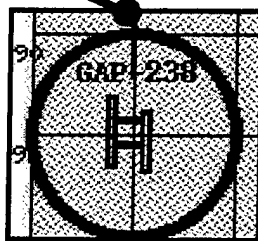
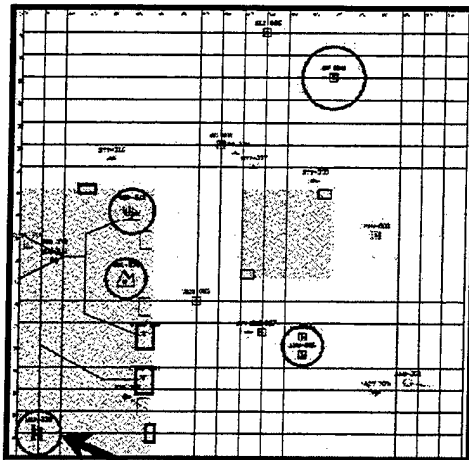
Tasks 2 & 3: The Beaches

Prerequisites:

1. Sea mines
2. Take Hill
3. Must take North Beach before South Beach

Force Packages:

1. (1) INF + (1) CAS + (1) DDG
2. (1) INF + (2) CAS
3. (2) INF + (1) CAS



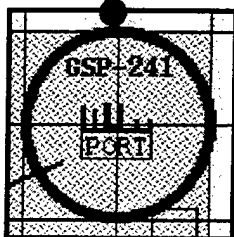
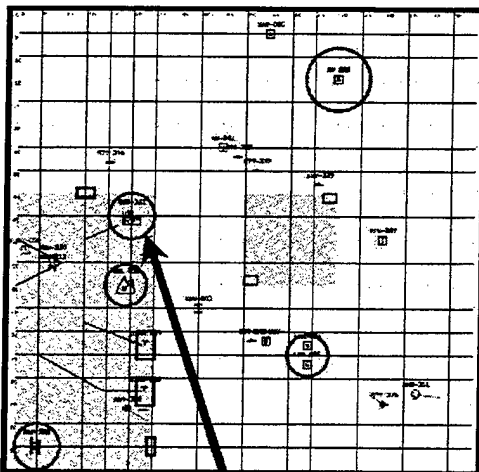
Task 4: The Airport

Prerequisites:

1. Ground Mines
2. SAM Sites
3. Beaches

Force Package:

1. (2) INF + (1) CAS



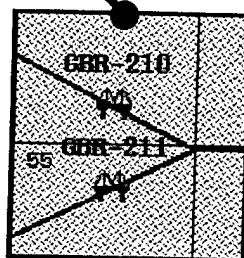
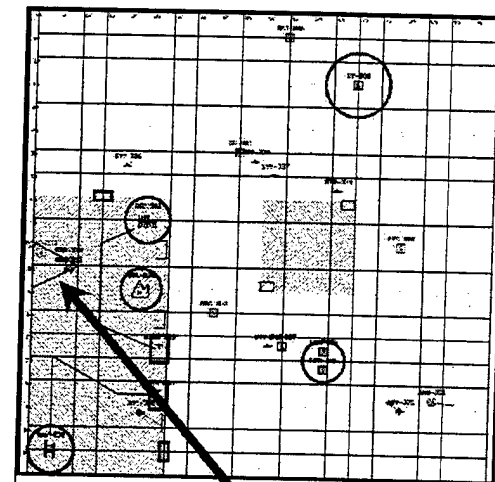
Task 5: The Seaport

Prerequisites:

1. Ground Mines
2. SAM Sites
3. Airport

Force Packages:

1. (2) INF + (1) CAS
2. (2) INF + (1) DDG
3. (2) INF + CG



Tasks 6 & 7: Lead Vehicle & Bridge

1. Detect Hostile Lead Vehicle
2. ID Hostile Lead Vehicle
3. Eliminate Lead Vehicle
(1) CAS + (1) SOF
4. Destroy only the Bridge on the road with the Hostile Lead Vehicle
(1) CAS+ (1) SOF+(1) ENG

APPENDIX F. SAMPLE DEPENDENT VARIABLE FILE, DATA CODING SCHEME, AND DATA TABLE

A. SAMPLE RAW DATA FILE

Text in bold are as they appear in the dependent variable files. Non-bolded text are added information to explain the raw data.

The header below contains the team name (B), the scenario (Interdependent architecture 1), number of players (6 plus one observer station), number of types of tasks encountered (32), and number of friendly zones to protect (9).

Team name: B
 Experiment condition: AC6rn1
 Number of tasks arrived: 151
 Number of DMs: 7
 Number of task classes: 32
 Number of penetration zones: 9

Number of task arrivals by task class describes the number of times a certain task appeared during the simulation.

Number of task arrivals by task class

1	take hill	8	patrol boat
1	airport	3	submarine
1	seaport	0	commercial sea
0	hold hill	16	neutral sea
1	take Nbeach	7	medivac
18	artillery	17	commercial air
9	Frog launcher	0	missile
5	Silkworm-real	1	take Sbeach
4	ground mine	1	lead Vehicle
10	sea mine	7	neutral traffic
13	hostile air	1	wrong bridge
5	hind helo	1	bridge
3	SAM site-real	1	UT1 ship trapped
3	SAM site-fake	1	UT2 oil platform
6	tank	1	UT3 armored convoy
5	silkworm-fake	1	UT4 embassy action

The number of initiated attack by each dm on various task classes indicates the number of times a specific player attacked a certain type of class. Notice the numbered data below is in columns of seven. Each column represents one of the players. The seventh node is the observer station.

Number of initiated attacks by each dm on various task classes														
0	0	0	0	1	0	0	take hill	0	6	0	0	0	0	patrol boat
0	0	1	0	0	0	0	airport	0	2	0	0	0	0	submarine
0	1	0	0	0	0	0	seaport	0	0	0	0	0	0	commercial sea
0	0	0	0	0	0	0	hold hill	0	0	0	0	0	0	neutral sea
0	1	0	0	0	0	0	take Nbeach	0	0	6	0	0	0	medivac
0	3	0	0	15	0	0	artillery	0	0	0	0	0	0	commercial air
0	5	0	0	4	0	0	Frog launcher	0	0	0	0	0	0	missile
1	0	0	0	4	0	0	Silkworm-real	0	0	0	0	1	0	take Sbeach
0	0	0	0	0	4	0	ground mine	0	1	0	0	0	0	lead Vehicle
0	0	9	0	0	0	0	sea mine	0	0	0	0	0	0	neutral traffic
11	1	0	0	0	0	0	hostile air	0	0	0	0	0	0	wrong bridge
5	0	0	0	0	0	0	hind helo	0	1	0	0	0	0	bridge
0	1	0	0	0	1	0	SAM site-real	0	1	0	0	0	0	UT1 ship trapped
0	0	0	0	0	0	0	SAM site-fake	1	0	0	0	0	0	UT2 oil platform
0	0	0	1	3	0	0	tank	0	1	0	0	0	0	UT3armoredconvoy
0	0	0	0	0	0	0	silkworm-fake	0	1	0	0	0	0	UT4embassyaction

The number of assisted attack by each dm on various task classes represents the number of times a player participated in a coordinated attack on a specific type of task.

Number of assisted attacks by each dm on various task classes														
0	0	1	0	0	0	0	take hill	0	0	0	0	0	0	patrol boat
0	0	0	1	1	0	0	airport	0	0	0	0	0	0	submarine
0	0	1	1	0	0	0	seaport	0	0	0	0	0	0	commercial sea
0	0	0	0	0	0	0	hold hill	0	0	0	0	0	0	neutral sea
0	0	0	1	1	0	0	take Nbeach	0	0	0	0	0	0	medivac
0	0	0	0	0	0	0	artillery	0	0	0	0	0	0	commercial air
0	0	0	0	0	0	0	Frog launcher	0	0	0	0	0	0	missile
1	0	0	0	1	0	0	Silkworm-real	0	1	0	1	0	0	take Sbeach
0	0	0	0	0	0	0	ground mine	0	0	0	0	0	1	lead Vehicle
0	0	0	0	0	0	0	sea mine	0	0	0	0	0	0	neutral traffic
0	0	0	0	0	0	0	hostile air	0	0	0	0	0	0	wrong bridge
0	0	0	0	0	0	0	hind helo	0	0	0	0	0	1	bridge
0	0	0	0	0	1	0	SAM site-real	0	0	1	0	0	0	UT1 ship trapped

0 0 0 0 0 0 0	SAM site-fake	0 0 0 0 0 0 0	UT2 oil platform
0 0 0 2 1 0 0	tank	1 0 1 0 1 0 0	UT3armoredconvoy
0 0 0 0 0 0 0	silkworm-fake	0 0 1 0 0 1 0	UT4embassyaction

The avg accuracy of attacks by each dm on various task classes indicates the average amount of time it took a certain player to prosecute a specific task.

A score of 999.00 indicates that no attack was conducted by that player.

Avg accuracy of attacks by each dm on various task classes							
999.00	999.00	999.00	999.00	80.97	999.00	999.00	take hill
999.00	999.00	100.00	999.00	999.00	999.00	999.00	airport
999.00	96.75	999.00	999.00	999.00	999.00	999.00	seaport
999.00	999.00	999.00	999.00	999.00	999.00	999.00	hold hill
999.00	96.75	999.00	999.00	999.00	999.00	999.00	take Nbeach
999.00	100.00	999.00	999.00	100.00	999.00	999.00	artillery
999.00	100.00	999.00	999.00	100.00	999.00	999.00	Frog launcher
96.72	999.00	999.00	999.00	62.50	999.00	999.00	Silkworm-real
999.00	999.00	999.00	999.00	999.00	100.00	999.00	ground mine
999.00	999.00	100.00	999.00	999.00	999.00	999.00	sea mine
100.00	100.00	999.00	999.00	999.00	999.00	999.00	hostile air
100.00	999.00	999.00	999.00	999.00	999.00	999.00	hind helo
999.00	53.12	999.00	999.00	999.00	50.00	999.00	SAM site-real
999.00	999.00	999.00	999.00	999.00	999.00	999.00	SAM site-fake
999.00	999.00	999.00	100.00	99.56	999.00	999.00	tank
999.00	999.00	999.00	999.00	999.00	999.00	999.00	silkworm-fake
999.00	100.00	999.00	999.00	999.00	999.00	999.00	patrol boat
999.00	100.00	999.00	999.00	999.00	999.00	999.00	submarine
999.00	999.00	999.00	999.00	999.00	999.00	999.00	commercial sea
999.00	999.00	999.00	999.00	999.00	999.00	999.00	neutral sea
999.00	999.00	100.00	999.00	999.00	999.00	999.00	medivac
999.00	999.00	999.00	999.00	999.00	999.00	999.00	commercial air
999.00	999.00	999.00	999.00	999.00	999.00	999.00	missile
999.00	999.00	999.00	999.00	100.00	999.00	999.00	take Sbeach
999.00	100.00	999.00	999.00	999.00	999.00	999.00	lead Vehicle
999.00	999.00	999.00	999.00	999.00	999.00	999.00	neutral traffic
999.00	999.00	999.00	999.00	999.00	999.00	999.00	wrong bridge
999.00	98.77	999.00	999.00	999.00	999.00	999.00	bridge
999.00	96.75	999.00	999.00	999.00	999.00	999.00	UT1 ship trapped
27.98	999.00	999.00	999.00	999.00	999.00	999.00	UT2 oil platform
999.00	94.79	999.00	999.00	999.00	999.00	999.00	UT3armoredconvoy
999.00	96.10	999.00	999.00	999.00	999.00	999.00	UT4embassyaction

The number of contacts by each dm on various tasks classes indicates the number of times a player collided with a specific enemy asset and incurred damage from the enemy.

Number of contacts (collisions) by each dm on various task classes

0 0 0 0 0 0 0	take hill	0 1 0 0 0 0 0	patrol boat
0 0 0 0 0 0 0	airport	0 0 0 0 0 0 0	submarine
0 0 0 0 0 0 0	seaport	0 0 0 0 0 0 0	commercial sea
0 0 0 0 0 0 0	hold hill	0 0 0 0 0 0 0	neutral sea
0 0 0 0 0 0 0	take Nbeach	0 0 0 0 0 0 0	medivac
0 0 0 0 0 0 0	artillery	0 0 0 0 0 0 0	commercial air
0 0 0 0 0 0 0	Frog launcher	0 0 0 0 0 0 0	missile
0 0 0 0 0 0 0	Silkworm-real	0 0 0 0 0 0 0	take Sbeach
0 0 0 0 0 0 0	ground mine	0 0 0 0 0 0 0	lead Vehicle
0 0 0 0 0 0 0	sea mine	0 0 0 0 0 0 0	neutral traffic
0 0 0 0 0 0 0	hostile air	0 0 0 0 0 0 0	wrong bridge
0 0 0 0 0 0 0	hind helo	0 0 0 0 0 0 0	bridge
0 0 1 0 0 0 0	SAM site-real	0 0 0 0 0 0 0	UT1 ship trapped
0 0 0 0 0 0 0	SAM site-fake	0 0 0 0 0 0 0	UT2 oil platform
0 0 0 2 0 0 0	tank	0 0 0 0 0 0 0	UT3armoredconvoy
0 0 0 0 0 0 0	silkworm-fake	0 0 0 0 0 0 0	UT4embassyaction

Total Number of contacts (collisions): 4

The number of penetrations on PZ's by task class indicates the number of times an specific enemy asset penetrated one of the nine friendly penetration zones and incurred enemy damage. Notice the nine columns for each of the PZ's.

Number of penetrations on PZ's by task classes

000000000	take hill	000000001	patrol boat
000000000	airport	000000100	submarine
000000000	seaport	000000000	commercial sea
000000000	hold hill	000000000	neutral sea
000000000	take Nbeach	000000000	medivac
000000000	artillery	000000000	commercial air
000000000	Frog launcher	000000000	missile
000000000	Silkworm-real	000000000	take Sbeach
000000000	ground mine	000000000	lead Vehicle
000000000	sea mine	000000000	neutraltraffic
010000000	hostile air	000000000	wrong bridge
000000000	hind helo	000000000	bridge
000000000	SAM site-real	000000000	UT1 ship trapped
000000000	SAM site-fake	000000000	UT2 oil platform
000000000	tank	000000000	UT3 armoredconvoy
000000000	silkworm-fake	000000000	UT4embassy action

Total Number of penetrations: 3

The number of attack on various task classes indicates the number of total attacks performed on a single task type.

Number of attacks on various task classes

1	take hill	0	patrol boat
1	airport	6	submarine
1	seaport	2	commercial sea
1	hold hill	0	neutral sea
0	take Nbeach	0	medivac
1	artillery	6	commercial air
18	Frog launcher	0	missile
9	Silkworm-real	0	take Sbeach
5	ground mine	1	lead Vehicle
4	sea mine	1	neutral traffic
9	hostile air	0	wrong bridge
12	hind helo	0	bridge
5	SAM site-real	1	UT1 ship trapped

2	SAM site-fake	1	UT2 oil platform
0	tank	1	UT3 armored convoy
4	silkworm-fake	1	UT4 embassy action

Total Number of attacks: 93

The average attack latency time on various task classes is average amount of time for a specific type of task to be prosecuted.

Average attack latency time on various task classes			
514.00	take hill	272.17	patrol boat
1463.00	airport	415.50	submarine
2032.50	seaport	999.00	commercial sea
999.00	hold hill	999.00	neutral sea
668.00	take Nbeach	208.33	medivac
46.53	artillery	999.00	commercial air
83.44	Frog launcher	999.00	missile
218.40	Silkworm-real	771.00	take Sbeach
391.75	ground mine	135.50	lead Vehicle
926.39	sea mine	999.00	neutral traffic
70.42	hostile air	999.00	wrong bridge
96.40	Hind helo	1206.00	bridge
197.75	SAM site-real	401.00	UT1 ship trapped
999.00	SAM site-fake	207.50	UT2 oil platform
317.38	tank	200.50	UT3 armored convoy
999.00	silkworm-fake	318.00	UT4 embassy action

B. DATA CODING SCHEME

The following is the data scheme used to distinguish various data in the data table (section C.)

Team	
A	1
B	2
C	3
D	4
E	5
F	6

Architecture	
Interdependent	AI
Autonomous	AA

C. DATA TABLES

ACCURACY SCORES

	AI-1	AI-2	AI-3	AI-4	AI-5	AI-6	AA-1	AA-2	AA-3	AA-4	AA-5	AA-6
UT1-A	0	96.75	96.75	100	96.75	100	96.75	100	100	95.19	75	100
UT2-A	95.66	27.98	95.34	100	97.34	97.78	97.56	93.06	99.77	100	72.78	95.34
UT3-A	98.3	94.79	94.79	100	100	98.95	100	76.09	100	98.05	100	1.6
UT4-A	98.05	96.1	96.1	94.15	100	100	100	96.1	100	0	100	76.2
Silkworm-A	78.7	69.34	96.72	98.05	76.2	99.34	99.34	100	99.34	100	88.48	79.34
SAM-A	65	34.37	33.33	83.3	50	49.17	63.44	66.7	100	22.67	65.84	100
Tank-A	93.78	66.44	83.11	66.2	66	60	83.33	80	68	66.7	100	100
Take Hill-A	31.82	80.97	100	100	96.75	100	100	100	96.75	93.67	96.75	100
Airport-A	100	100	100	98.9	98.9	100	100	100	18	100	100	98.9
Seaport-A	100	96.75	100	100	98.9	98.9	100	98.9	100	98.9	100	100
Take Nbeach-A	96.75	96.75	100	100	97.95	100	100	96.75	96.75	96.75	100	100
Take Sbeach-A	100	100	100	100	97.95	100	100	100	96.75	100	100	100
Lead Vehicle-A	100	100	100	100	0	100	100	100	100	100	100	100
Bridge-A	100	98.77	100	98.77	0	100	100	100	100	100	100	76
Artillery-A	100	100	100	98	100	100	100	94.4	100	95	100	100
Frog Launcher-A	100	100	100	77.8	88.9	100	100	100	100	66.7	100	100
Hostile Air-A	76.92	92.3	100	83.3	69.2	91.7	100	91.7	84.6	84.6	100	92.3
Hind Helo-A	100	100	100	75	100	100	100	100	100	60	100	80
Patrol Boat-A	75	75	100	83.3	50	75	100	75	87.5	87.5	100	87.5
Submarine-A	66.7	66.7	100	100	66.7	100	100	100	100	100	66.7	100
Mission-A	86	91	97	96	81	88	94	93	81	95	96	94
Strength-A	89	85	89	89	76	94	94	93	88	83	92	89

LATENCY SCORES IN SECONDS

	AI-1	AI-2	AI-3	AI-4	AI-5	AI-6	AA-1	AA-2	AA-3	AA-4	AA-5	AA-6
UT1-L	450	401	211.5	260.5	353.5	183.5	256	280.5	234	375	305	354
UT2-L	288	207.5	375.5	247.5	257	356	387	210	305.5	256	287	381
UT3-L	232	200.5	293	170.5	336.5	128	268.5	137	409.5	325.5	171.5	248
UT4-L	302.5	318	296	241.5	365.5	284.5	276.5	304.5	328	420	328	360.5
Silkworm-L	255.25	218.4	289	122.6	310.12	163.7	169	343.8	142.4	181.9	187.2	373.38
SAM-L	622.75	197.75	370.5	339.17	825.25	224.5	217.75	213.5	282.17	1259	479.25	572.33
Tank-L	404.08	317.38	444.2	692.88	564.75	264	338.6	378.12	370.67	678.5	301.17	540.42
Take Hill-L	509	514	510.5	578.5	540.5	284	494	372	374	550.5	252	483
Airport-L	1719.5	1463	1800.5	1866.5	2163.5	1643.5	1464.5	1567	1722	2441	1543	2192.5
Seaport-L	2177.5	2032.5	2370.5	2206.5	2409.5	2030	1941.5	2016	2261	2651.5	2149.5	2488.5
Take Nbeach-L	718.5	668	626.5	634	695	466.5	548.5	655.5	528.5	620	438	678
Take Sbeach-L	803.5	771	774	946	924.5	628	673	813	639.5	917.5	729	906.5
Lead Vehicle-L	160	135.5	161	250.5	1000	144	237.5	146	137.5	291	143	143
Bridge-L	1317	1206	1390.5	1282.5	1500	1214.5	1420	1209	1318	1384	1262.5	1189
Artillery-L	59.45	46.53	71.2	68.84	103.75	41.73	31.27	70.15	31.02	94.76	63.32	60.55
Frog Launcher-L	107.56	83.44	98.22	74.29	126.25	38.75	67.72	115.25	57.56	43.25	98.17	101.83
Hostile Air-L	96.9	70.42	58.38	77.95	94.72	64.64	50.73	65.45	58.5	81.09	85.5	73.75
Hind Helo-L	91.7	96.4	34.6	120.33	90.7	40.25	71	69.5	40.9	55.67	88.2	90.75
Patrol Boat-L	112.36	272.17	221.38	275.1	280.12	247.25	218.81	281.17	70.5	229.57	313.62	150.36
Submarine-L	694	415.5	736.33	649	906.5	349.17	612.17	395.5	684.5	594.33	555.25	668

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

1. Benson, E. L., "Conduct and Assessment of A2C2 Experiment 3 and Guidelines for Future Experimentation", Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1997.
2. Devore, J. L., *Probability and Statistics for Engineering and the Sciences*, Fourth ed., Duxbury Press, 1995.
3. Drake, J. F., "Design and Development of the Scenario for the Second NPS A2C2 Experiment", Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1997.
4. Hocevar, S. P., Kemple, W. G., Kleinman, D. L., Porter, G. R., "Assessments of Simulated Performance of Alternative Architectures for Command and Control: The Role of Coordination", Proc. 1999 Command and Control Research and Technology Symposium, NPS, Monterey, CA, June 1999.
5. Hocevar, S. P., "Autonomous vs. Interdependent Structures: Impact on Unpredicted Tasks in a Simulated Joint Task Force Mission", Proc. 2000 Command and Control Research and Technology Symposium, NPS, Monterey, CA, June 2000.
6. Green, L. R., "The Effectiveness of Tactical Adaptation and Coordination Training on Team Performance in Tactical Scenarios", Master's Thesis, Naval Postgraduate School, Monterey, CA, June 1994.
7. Kleinman, D. L., Young, P. W., "DDD-III Scenario Generator Users Manual DDD-III Version 3.5.2", Instruction Guide, October 1997.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center 2
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, VA 22060-6218

2. Dudley Knox Library 2
Naval Postgraduate School
411 Dyer Road
Monterey, CA 93943-5101

3. Mr. Arison 1
JCS/J6
Pentagon
Washington, D.C. 20350-2000

4. Capt Maslowsky 1
CNO/N62
Pentagon
Washington, D.C. 20350-2000

5. Lt Col Ziegenfuss 1
HQMC C4I Plans and Policy Division
2 Navy Annex
Washington, D.C. 20380-1775

6. HQ AFCIC/XPF 1
Lt Col Jacques
Pentagon
Washington, D.C. 20350-2000

7. Chairman, C4I Academic Group.....1
Code CC/Bo
Naval Postgraduate School
Monterey, CA 93943

8. Professor William G. Kemple 2
Code CC/Ke
Naval Postgraduate School
Monterey, CA 93943

9. Professor Susan P. Hocevar 1
Code SM/Hc
Naval Postgraduate School
Monterey, CA 93943
10. Gerald S. Malecki 1
Office of Naval Research, ONR 342
800 N. Quincy Street
Arlington, VA 22217-5660
11. LT W. L. Pasaraba, USN.....2
1492 N. 17th St.
El Centro, CA 92243