



NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

**AN ALTERNATIVE OPTIMIZATION MODEL AND
ROBUST EXPERIMENTAL DESIGN FOR THE
ASSIGNMENT SCHEDULING CAPABILITY FOR
UNMANNED AERIAL VEHICLES (ASC-U) SIMULATION**

by

Derek M. Oliver

June 2007

Thesis Advisor:
Second Reader:

Susan M. Sanchez
Darryl K. Ahner

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
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 June 2007	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE An Alternative Optimization Model and Robust Experimental Design for the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) Simulation			5. FUNDING NUMBERS	
6. AUTHOR(S) Derek M. Oliver				
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 A	
13. ABSTRACT (maximum 200 words) The Modeling, Virtual Environments, and Simulations Institute (MOVES) and the United States Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) at the Naval Postgraduate School, Monterey, California, developed the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation to aid in the analysis of future U.S. Army Unmanned Aerial Vehicle (UAV) requirements. TRAC selected ASC-U to provide insight into the programmatic decisions addressed in the U.S. Army UAV-Mix Analysis that directly affects future development and fielding of UAVs to include the Future Combat System. ASC-U employs a discrete event simulation coupled with the optimization of a linear objective function. At regular intervals, ASC-U obtains an optimal solution to an assignment problem that assigns UAVs to mission requirements that are available or will be available at some time in the future. This thesis presents an alternative optimization model, explores 23 simulation factors, and provides sensitivity analysis for how UAV coverage may degrade in the presence of adverse random events. Integer programming, experimental design, and an innovative Optimized Flexible Latin Hypercube (OFLH) design are used to evaluate a representative sample from an Army 2018 scenario. The conclusions suggest the following: the alternative optimization model developed in this thesis can successfully maximize ASC-U value without the use of a heuristic; Smaller Optimization Intervals do not guarantee higher total value when the heuristics are included; to maximize total value, Early Return should be set to FALSE and Secondary Areas should be set to TRUE; an OFLH is valuable for robust analysis of simulation models containing many factors; and as the model factors change over predefined ranges, the solution quality is consistent.				
14. SUBJECT TERMS Integer Programming, Latin Hypercube Designs, Design of Experiments, Linear Regression Models, Simulation Analysis, Unmanned Aerial Vehicles, Discrete Event Simulation, Future Combat Systems, UAV, Optimization, Optimized Flexible Latin Hypercubes, Nearly Orthogonal Latin Hypercubes, Stochastic Interactive Simulation			15. NUMBER OF PAGES 117	
			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	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**AN ALTERNATIVE OPTIMIZATION MODEL AND ROBUST
EXPERIMENTAL DESIGN FOR THE ASSIGNMENT SCHEDULING
CAPABILITY FOR UNMANNED AERIAL VEHICLES (ASC-U) SIMULATION**

Derek M. Oliver
Major, United States Air Force
B.S., United States Air Force Academy, 1992

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

**NAVAL POSTGRADUATE SCHOOL
June 2007**

Author: Derek M. Oliver

Approved by: Susan M. Sanchez
Thesis Advisor

Darryl K. Ahner
Second Reader

James N. Eagle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Modeling, Virtual Environments, and Simulations Institute (MOVES) and the United States Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) at the Naval Postgraduate School, Monterey, California, developed the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation to aid in the analysis of future U.S. Army Unmanned Aerial Vehicle (UAV) requirements. TRAC selected ASC-U to provide insight into the programmatic decisions addressed in the U.S. Army UAV-Mix Analysis that directly affects future development and fielding of UAVs to include the Future Combat System. ASC-U employs a discrete event simulation coupled with the optimization of a linear objective function. At regular intervals, ASC-U obtains an optimal solution to an assignment problem that assigns UAVs to mission requirements that are available or will be available at some time in the future.

This thesis presents an alternative optimization model, explores 23 simulation factors, and provides sensitivity analysis for how UAV coverage may degrade in the presence of adverse random events. Integer programming, experimental design, and an innovative Optimized Flexible Latin Hypercube (OFLH) design are used to evaluate a representative sample from an Army 2018 scenario. The conclusions suggest the following: the alternative optimization model developed in this thesis can successfully maximize ASC-U value without the use of a heuristic; smaller optimization intervals do not guarantee higher total value when the heuristics are included; to maximize total value, Early Return should be set to FALSE and Secondary Areas should be set to TRUE; an OFLH is valuable for robust analysis of simulation models containing many factors; and as the model factors change over predefined ranges, the solution quality is consistent.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	BACKGROUND	1
B.	PURPOSE.....	4
C.	RESEARCH QUESTIONS.....	4
D.	RESEARCH SCOPE.....	5
E.	THESIS ORGANIZATION.....	5
II.	FUTURE COMBAT SYSTEM AND UNMANNED AERIAL SYSTEMS.....	7
A.	SMALL UNMANNED AERIAL VEHICLE (SUAV).....	8
B.	TACTICAL UNMANNED AERIAL VEHICLE (TUAV).....	10
C.	EXTENDED RANGE MULTI-PURPOSE (ERMP) UNMANNED AERIAL VEHICLE (UAV)	11
D.	CLASS IV UNMANNED AERIAL VEHICLE (UAV).....	13
E.	GROUND CONTROL STATIONS (GCSS) AND MISSION PAYLOADS.....	14
1.	Launch and Recovery Site (LRS).....	14
2.	Ground Control Station (GCS).....	15
3.	Remote Video Terminal	15
4.	Mission Payloads.....	16
III.	SIMULATION DESCRIPTION.....	17
A.	UNMANNED AERIAL VEHICLE (UAV) ALLOCATION PROBLEM	17
B.	TECHNICAL APPROACH.....	19
C.	CONSTRAINTS, LIMITATIONS, AND ASSUMPTIONS	24
1.	Constraints.....	24
2.	Limitations.....	25
3.	Assumptions.....	26
D.	ASC-U INPUT	27
1.	Meta Data Worksheet.....	28
2.	Air Frame Worksheet.....	28
E.	ASC-U OUTPUT.....	30
IV.	AN ALTERNATIVE OPTIMIZATION MODEL	33
A.	INTEGER PROGRAMMING.....	33
B.	UNMANNED AERIAL VEHICLE (UAV) ALLOCATION MODEL	34
1.	Indices	35
2.	Subsets.....	35
3.	Parameters.....	35
4.	Decision Variable	35
5.	Constraints.....	36
6.	Objective Function.....	37
C.	PRELIMINARY RESULTS	38

V.	DATA ANALYSIS	43
A.	DESIGN OF EXPERIMENT (DOE)	43
1.	Factor Selection	44
2.	Latin Hypercube Designs	46
B.	MEASURES OF EFFECTIVENESS (MOES)	48
C.	LINEAR REGRESSION MODELS	49
D.	TOTAL VALUE MODEL	51
1.	Main Effects Model	52
2.	First Order Model with Interactions.....	53
3.	Full Quadratic Model	54
4.	Final Total Value Model.....	55
5.	Block Analysis	60
6.	Significant Factors and Interpretation of Regression Coefficients	63
7.	Significant Interactions	68
E.	SENSITIVITY ANALYSIS.....	71
VI.	CONCLUSIONS	79
A.	IMMEDIATE IMPACT	80
B.	FOLLOW-ON RESEARCH.....	81
APPENDIX A.	OPTIMIZED FLEXIBLE LATIN HYPERCUBE DESIGN.....	83
APPENDIX B.	GENERAL ALGEBRAIC MODELING SYSTEM (GAMS) CODE	85
APPENDIX C.	VISUAL BASIC FOR APPLICATIONS SCRIPT FOR THE EXPERIMENTAL DESIGN INTERFACE	91
	LIST OF REFERENCES.....	95
	INITIAL DISTRIBUTION LIST	97

LIST OF FIGURES

Figure 1.	SUAV RQ-11 Raven (from)	9
Figure 2.	RQ-7B TUAV (from)	10
Figure 3.	ERMP UAV (from).....	11
Figure 4.	Fire Scout, RQ-8A (from).....	13
Figure 5.	Study Analysis Process Design (after).....	18
Figure 6.	Assignments Dynamics Example (after)	20
Figure 7.	ASC-U Dynamic Cueing and Transition (after)	21
Figure 8.	ASC-U Experimental Design Interface	46
Figure 9.	Scatterplot of Total Value by Design Point	51
Figure 10.	Actual vs. Predicted Total Value for Main Effects.....	52
Figure 11.	Actual vs. Predicted Total Value for First Order with Interactions.....	54
Figure 12.	Actual vs. Predicted Total Value for Full Quadratic Model	55
Figure 13.	Statistical Output for Final Total Value Model	56
Figure 14.	Actual by Predicted Plot for Final Total Value Model	57
Figure 15.	Residual by Predicted Plot for Total Value Residuals.....	57
Figure 16.	Histogram, Outlier Box Plot, and Normal Plot for Total Value Residuals.....	58
Figure 17.	ASC-U Mean Total Value vs. Grouped Optimization Intervals.....	60
Figure 18.	Histograms and Smooth Curve Fit for OFLH Block Analysis	62
Figure 19.	Prediction Profiler for Final Total Value Model.....	64
Figure 20.	Total Value Regression Tree After Four Splits	66
Figure 21.	Interaction Plot for Optimization Interval and Secondary Areas.....	69
Figure 22.	Relationship between x , $F(x)$, $G(F(x))$ (after)	73
Figure 23.	Effect of Stratification in Latin Hypercube Sampling (after)	74
Figure 24.	Stochastic Spreadsheet Model (Snapshot)	76
Figure 25.	Coverage Probability by Design Point.....	78

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1.	SUAV RQ-11 Raven Characteristics and Performance Data (from).....	9
Table 2.	Shadow RQ-7A/B Characteristics and Performance Data (after).....	11
Table 3.	Predator MQ-9A Characteristics and Performance Data (after).....	12
Table 4.	Fire Scout RQ-8B Characteristics and Performance Data (from)	14
Table 5.	ASC-U Input Worksheets (after)	27
Table 6.	Measures of Performance Tables (after).....	30
Table 7.	Table of Test Values for UAV Allocation Model	38
Table 8.	Display of GAMS Output for Subsets of the UAV Allocation Model	39
Table 9.	GAMS Solution for UAV Allocation Model.....	40
Table 10.	lp_solve Solution for UAV Allocation Model.....	41
Table 11.	ASC-U Factors and Ranges for DOE	45
Table 12.	Significant Interactions and R-Square Value for Total Value Model.....	55
Table 13.	Designation of OFLH Blocks by Heuristic Value	61
Table 14.	Components of OFLH Block with Highest Mean Total Value	62
Table 15.	Significant Factors and Signs on Regression Coefficients for Final Total Value Model.....	63
Table 16.	Percent Change Comparison for ASC-U Optimization Intervals	71

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I want to thank my thesis advisor, Professor Susan M. Sanchez for taking the time to help me explore the vast expanse of Simulation and Design of Experiments. I also want to thank my Second Reader, Lieutenant Colonel Darryl K. Ahner, U.S. Army, for the assistance he provided in scoping my thesis, helping me to develop the correct research questions, and being available whenever I needed his advice or support. I thank Professor Arnold H. Buss and Mr. John L. Ruck for all of their assistance with command-line runs and interpretation of the nuances of the simulation. My thanks also to Professor Paul Sanchez for helping me process long simulation runs on the computing cluster and for teaching me the utility of UNIX, and to Professor Kevin Wood for opening up the door to an integer mathematical programming solution that had taken me months to attempt to construct on my own. Finally, I want to thank Colonel Alejandro Hernandez, U.S. Army, for giving me the opportunity to employ an innovative experimental design that he developed as part of his doctoral dissertation.

Most of all, I thank God for blessing me with my wife Linda and our children Derek, Jr., Olivia, Rose, and David. Their support has been unwavering throughout my entire Operations Research experience here at the Naval Postgraduate School. They have endured my struggle every single day of this journey. Through the peaks and the valleys, painful headaches, countless late nights, multiple lost weekends, the frustrations of less than stellar results, and the joys of seeing a tremendous amount of hard work, determination, and persistence finally bear fruit. Without them surrounding me, none of this would be possible.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

The United States Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) and the Modeling, Virtual Environments, and Simulations Institute (MOVES) at the Naval Postgraduate School, Monterey, California, developed the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation to conduct operational analysis of future U.S. Army Unmanned Aerial Vehicle (UAV) requirements. TRAC selected ASC-U to provide insight into the programmatic decisions addressed in the U.S. Army UAV-Mix Analysis that directly impacts future UAV system development and fielding of the Future Combat System (FCS).

This thesis presents an alternative optimization model for ASC-U; explores the statistical and practical significance of 23 simulation factors in an innovative experimental design; and analyzes their impact on the ASC-U Total Value Measure of Effectiveness. We focus on the following research questions concerning the allocation of UAVs in the ASC-U simulation.

- Does changing the existing optimization logic (an assignment problem that uses a heuristic to preassign UAV Ground Control Stations to Mission Areas) to a pure binary integer programming model provide an improvement in the overall ASC-U solution?
- Can an integrated experimental design interface within ASC-U provide a more robust analytical process? Which model factors have the greatest impact on the total value derived when allocating UAVs to designated Mission Areas?
- Is there an optimal duration of the ASC-U optimization interval (the amount of time between subsequent reoptimizations)? Does this particular parameter of the ASC-U optimization logic scenario behave as expected when set to values less than one simulation hour?
- Across an empirically determined range on the input factors, does ASC-U provide consistent results?

ASC-U is a discrete event simulation that supports the U.S. Army UAV Mix Analysis. The movement of multiple entities on the modern battlefield complicates military operations. This complexity poses difficulties for UAV allocation and scheduling. ASC-U provides feasible UAV flight schedules that are heavily dependent on accurate input data. The simulation provides a feasible solution to the following problem: given a scenario that specifies the number and types of UAVs, initial UAV locations, and UAV performance characteristics, identify the number of successful missions and a schedule for each UAV. The solution is constrained by Ground Control Station (GCS) locations and capacities, Launch and Recovery Site (LRS) locations and capacities, Remote Viewing Terminal (RVT) requirements, and communication platform footprints and capacities.

The alternative optimization model that we developed for ASC-U takes as input a collection of UAVs, LRSs, Mission Packages, Mission Areas, and GCSs. We further divide these entities into the following subsets: UAVs available at each LRS, Mission Packages available at each LRS, Mission Areas reachable from each LRS, and GCSs assigned to each Mission Area. The data required for the model includes the capacity of each GCS; the value received by a UAV for satisfying a mission requirement over a designated Mission Area; an upper limit on the number of Mission Packages available at each LRS; and an upper limit on the number of UAVs at each LRS.

We used robust experimental design, block analysis, and an innovative Optimized Flexible Latin Hypercube (OFLH) design consisting of 96 design points to evaluate a representative sample derived from an Army 2018 scenario. This robust design process gave us the ability to explore the ASC-U solution quality across a range of model factors. The analysis provides insights that will be useful to future modeling efforts and the operational elements of FCS. The conclusions suggest the following:

- The alternative optimization model developed in this thesis can successfully maximize value obtained by UAV assignment while maintaining the required relationships between LRSs, UAVs, mission packages, GCSs, and mission areas without the use of a heuristic.

- The Optimization Interval alone is not significant when the duration of the interval falls below one simulation hour.
- Incrementally smaller Optimization Intervals do not guarantee that higher Total Value will result when the heuristics Early Return, Secondary Areas, and Appended Areas are included in simulation runs.
- ASC-U run times are long for Optimization Intervals below one simulation hour (ranging from a low of 8 hours to a high of 48 hours per run).
- To maximize Total Value in ASC-U runs, the heuristic Early Return should be set to FALSE, the heuristic Secondary Areas should be set to TRUE, and the heuristic Appended Areas should be set to TRUE.
- ERMP Operating Time, Class IV Operating Time and SUAV Operating Radius are the most significant UAV performance characteristics in this scenario.
- An OFLH design proves tremendously valuable for robust analysis of simulation models containing many (20 or more) factors and significantly decreases runtime.
- The long run times for Optimization Intervals below one simulation hour do not appear to provide a significant improvement in the relative ASC-U solution quality.
- As the model factors change over predefined ranges, the solution quality appears to remain consistent with expected results.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

Approximately one decade after the Cold War, the United States Army realized that its current force structure was not suited to meet its current or future battlefield challenges. Although the Army's "heavy" combat systems have significant combat power, the extensive support requirements and heavy footprint creates a long deployment lead time. The traditionally "light" Army forces have the ability to rapidly deploy, but lack the necessary firepower for sustained combat operations. To address this mismatch between heavy and light forces, the Army's leadership made the decision to radically transform itself into a "Future Force."¹

The Army anticipates that the Future Force will be organized, trained, and equipped for rapid and sustained land combat. This will require a technologically advanced force that is versatile and responsive to the full spectrum of combat operations. This Future Force will have an offensive orientation and rely on revolutionary operational concepts, enabled by advanced equipment and new support practices. This force will also engage in battle very differently than current Army forces—easily transportable, lightweight vehicles will serve as the centerpiece, rather than the traditional heavily armored tanks. These light, agile forces will possess the ability to quickly integrate with existing combat forces in a variety of scenarios, to include joint operations. The ability to quickly adapt to any contingency scenario will prevent future adversaries from seizing and maintaining the initiative. With an agile force, unit commanders must have the authority and high-quality information to make decisions in order to respond to the dynamic environment they face. Thus, to succeed, the Army's transformation cannot rely

¹ Paul L. Francis, "The Army's Future Combat Systems' Features, Risks and Alternatives," Congressional Testimony, United States General Accounting Office, GAO-04-635T, 1 April 2004, p. 3.

solely on weapons and equipment. The transformation must result from the synergy between new tactics, sound doctrine, improved organization, new equipment and an overall culture-shift.²

Of the previously mentioned elements, the core of the Future Combat System (FCS) falls in the “equipment” category, which is where Unmanned Aerial Vehicles (UAVs), UAV organizations, UAV payloads, and ground control/launch and recovery support requirements enter the discussion. The FCS will provide the majority of weapons and sensor platforms that comprise the new “brigade-like” modular units of the Future Force, called Units of Action. Each Unit of Action will be a rapidly deployable combat organization (about the size of an Army brigade), but with the combat power of a larger Army division.³

The preliminary FCS increment will consist of 14 individual systems (categorized as manned ground systems, unmanned ground systems and unmanned aerial vehicles), connected by an advanced network architecture, and operated by “the Soldier.” Hence the FCS designation as a “14+1+1” system. While some systems may take on a more significant role in the network than others may, the network will reside in each system. This resident network will provide information to, and to receive information from, each individual system as required.⁴

The initial FCS requirement included four classes of UAVs (Class I, II, III, and IV), to provide substantial mission area coverage at different operational echelons. The Army has reduced the number of UAVs under the FCS umbrella, leaving the program with only the development of the smallest (Class I) and largest (Class IV) UAVs. The FCS program continues to adjust based on the current operational environment and

² Paul L. Francis, “The Army’s Future Combat Systems’ Features, Risks and Alternatives,” Congressional Testimony, United States General Accounting Office, GAO-04-635T, 1 April 2004, p. 4.

³ Ibid.

⁴ COL Lee Fetterman, United States Army, “14+1+1 Systems Overview,” Future Combat System (Brigade Combat Team) (FCS (BCT)), White Paper, 14 March 2007, p. 2, www.army.mil/fcs, retrieved 11 April 2007.

perceived needs of the Future Force. Additionally, the Army will continue to use the Shadow and Raven UAV fleets and will field the Warrior UAV as planned. These three fleets are not part of FCS.⁵

Within the FCS operational concept, the UAV missions include reconnaissance, target acquisition and designation, mine detection, and wide band communications relay. Originally, these UAVs needed to be designed, developed, and demonstrated prior to the initial FCS production decision, set for November of 2008.

The evolution of the current force UAVs (Small UAV, Tactical UAV, Extended Range/Multiple Purpose UAV), and the supporting Army UAV Mix Analysis, will help lay the foundation for the future programmatic decisions required for FCS UAV acquisition. The results of this iterative process will translate into recommendations for types and quantities of UAVs required by Future Force Units of Action. The Modeling, Virtual Environments, and Simulations Institute (MOVES) and the Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) at the Naval Postgraduate School developed the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation to aid in the analysis of UAV requirements for the modular and FCS force.

ASC-U combines discrete event simulation with the optimization of a linear objective function in order to build UAV schedules. The simulation creates a flight schedule for a particular scenario using a specific UAV inventory. At user-defined intervals, the simulation provides an optimal solution to a UAV assignment problem. The simulation considers those UAVs that are available for launch and assigns them to available missions (reconnaissance, communications, target identification, etc.) or missions that will be available in a future time window. The simulation computes a mission value for each UAV assignment by considering the flight time to each mission area, and the amount of time that the UAV provides coverage over the Mission Area.

⁵ Ashley Roque, "Army Will Cut Future Combat System's Class II and Class III UAVs," *InsideDefense.com*, Inside Washington Publishers, 12 December 2006, retrieved 10 April 2007.

B. PURPOSE

The broad purpose of this thesis is threefold. First, we seek to evolve the optimization logic within ASC-U to realize a marked improvement in solution quality and runtime. Next, our goal is to create an integrated Design of Experiment (DOE) front end for the model as a Proof of Concept. We intend to use the DOE front end to rapidly and accurately populate the model input tables, and set up a robust experimental design to explore the effects of changes to continuous and discrete input factors. Finally, we will evaluate how these adjustments to ASC-U (in both its optimization logic and model input factors) influence the overall Total Value Measure of Effectiveness (MOE).

C. RESEARCH QUESTIONS

The basic research questions addressed by this thesis are:

- Does changing the existing optimization logic (an assignment problem that uses a heuristic to preassign UAV Ground Control Stations to Mission Areas) to an alternative optimization model provide an improvement in the overall ASC-U solution?
- Can an integrated experimental design interface within ASC-U provide a more robust analytical process? Which model factors have the greatest impact on the total value derived when allocating UAVs to designated Mission Areas?
- Is there an optimal duration of the ASC-U optimization interval below one simulation hour (the amount of time between subsequent reoptimizations)? When this parameter of the ASC-U optimization logic is set to values below one simulation hour, are the results as expected?
- Across an empirically determined range on the input factors, does ASC-U provide consistent results?

D. RESEARCH SCOPE

The scope of this thesis is to evaluate and improve a discrete event simulation that optimizes a linear objective function. Given a scenario and an inventory of UAVs, ASC-U builds a schedule to execute UAV sorties. The simulation accounts for UAV capabilities, sensor availability, locations and capacities of Launch and Recovery Sites, GCSs, and designated Mission Areas where sensor requirements exist. This research will also consider how the frequency of the optimization interval and the duration of the decision time window affect the ASC-U output.

E. THESIS ORGANIZATION

Chapter II provides more detail on FCS and the Unmanned Aerial Systems used for the analysis of ASC-U in this thesis. A short description of each UAV is included. Chapter III gives some detail on how ASC-U provides a solution to the UAV assignment problem. Chapter IV will describe a pure binary integer-programming model that eliminates the need for heuristic assignment of GCSs to Mission Areas (currently utilized in the existing ASC-U optimization logic). Chapter V describes the model factors selected for the experiment, development of design points, details on the DOE front end for ASC-U, a multiple regression model for ASC-U total value, and analysis and interpretation of the ASC-U output. Chapter VI summarizes conclusions drawn from the analysis and provides recommendations for follow-on research.

THIS PAGE INTENTIONALLY LEFT BLANK

II. FUTURE COMBAT SYSTEM AND UNMANNED AERIAL SYSTEMS

The Future Combat System (FCS) is the Army's first full spectrum modernization effort in almost 40 years. The critical needs of today's operational forces compelled the Army to accelerate its transformation. The Army is in the middle of a continuous process of transformation that entails a comprehensive restructuring across several domains. The FCS Brigade Combat Team (BCT) represents a significant portion of the overall transformation. The FCS BCT is the material solution for the Future Force. It also represents the Army's primary modernization strategy for approximately the next 20 years. The FCS is the embodiment of the modular force, designed for "full-spectrum" operations. It will provide a communications network for current systems, systems under development, and those systems projected to achieve operational capability at some point in the future. The FCS will adapt to the requirements of traditional warfare as well as more complex, irregular contingencies in urban environments. It will be useful in a joint context across all the military services and provide robust connectivity in a "system of systems" framework. This unprecedented capability will enable greater joint interoperability, improved situational awareness and highly synchronized operations that have been unattainable in the past. When operational, the FCS will provide the Army and the joint community an unmatched ability to see the enemy, rapidly engage, and defeat the enemy on the 21st century battlefield.⁶

In addition to the network and the Soldier, the FCS BCT includes:

- Unattended Ground Sensors
- Non-Line of Sight-Launch Systems
- Unmanned Aerial Vehicles (UAVs)
- Unmanned Ground Vehicles
- Small Unmanned Ground Vehicles

⁶ COL Lee Fetterman, United States Army, "14+1+1 Systems Overview," Future Combat System (Brigade Combat Team) (FCS (BCT)), White Paper, 14 March 2007, p. 2, www.army.mil/fcs, retrieved 11 April 2007.

- Multifunctional Utility/Logistics and Equipment Vehicles
- Manned Ground Vehicles⁷

The UAV capability within the FCS is where commanders can actually realize the vision of “see first, understand first, act first, and finish decisively across the full spectrum of operations.” Unmanned Aerial Systems, defined as UAVs with their requisite Ground Control/Launch and Recovery equipment, give commanders the ability to see and potentially understand the adversary first. This is where ASC-U proves valuable.⁸

The ASC-U discrete event simulation gives the user the capability to model different scenarios with the requisite number of UAVs. The UAVs evaluated in this thesis are those contained in a representative sample taken from an Army 2018 Scenario, and include both Current Force UAVs and an FCS UAV. These UAVs were evaluated based on their characteristics and performance capabilities. The following sections provide a short description of the UAVs and a chart depicting performance characteristics.

A. SMALL UNMANNED AERIAL VEHICLE (SUAV)

In Figure 1, the Raven B system (an enhanced version of the battle-tested Raven A) is a lightweight UAV designed for rapid deployment and high mobility in both military and commercial environments that require low-altitude surveillance, reconnaissance and intelligence.⁹

⁷ COL Lee Fetterman, United States Army, “14+1+1 Systems Overview,” Future Combat System (Brigade Combat Team) (FCS (BCT)), White Paper, 14 March 2007, p. 3, www.army.mil/fcs, retrieved 11 April 2007.

⁸ Bill White, Director, Brigade Combat Team Operations, *Future Combat System Smart Book*, TACOM, 18 September 2006, p. 18.

⁹ Aerovironment Web site, Raven RQ-11B, http://www.avinc.com/publish/2006/10/04/AV_DATASHEET_RAVEN_B_10_6_06.pdf, retrieved 11 April 2007.



Figure 1. SUAV RQ-11 Raven (from)¹⁰

As the most advanced SUAV deployed with the Army, the Raven B can operate autonomously to utilize the system's advanced avionics and precision Global Positioning System (GPS) navigation. With a wingspan of 4.5 feet and weighing 4.2 pounds, the hand-launched Raven B provides day or night aerial observation at line-of-sight ranges up to 10 kilometers. The Raven B delivers near real-time color imagery to its ground control system and remote viewing terminals.¹¹ Table 1 provides characteristics and performance data for the RQ-11B.

RQ-11B			
Length	3.0 ft	Wing Span	4.5 ft
Gross Weight	4.2 lbs	Payload Weight	6.5 oz
Engine	Electric	Max Speed	82 kmph
Endurance	60 – 90 minutes	Operating Radius	10 km
Ceiling	500 ft Above Ground Level	Landing	Deep Stall Vertical
Sensor	Electro Optical/Infrared		

Table 1. SUAV RQ-11 Raven Characteristics and Performance Data (from)¹²

¹⁰ Aerovironment Web site, Raven RQ-11B, http://www.avinc.com/publish/2006/10/04/AV_DATASHEET_RAVEN_B_10_6_06.pdf, retrieved 11 April 2007.

¹¹Ibid.

¹² Office of the Secretary of Defense, "Unmanned Aircraft Systems Roadmap 2005-2030," online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, p. 8, retrieved 1 April 2007.

B. TACTICAL UNMANNED AERIAL VEHICLE (TUAV)

The Army selected the RQ-7 Shadow 200 (TUAV) in Figure 2 in December 1999 to meet the Brigade-level UAV requirement for support to ground maneuver commanders.



Figure 2. RQ-7B TUAV (from)¹³

The RQ-7A is of the same twin-boom pusher layout as several other battlefield UAVs, e.g., the RQ-2 Pioneer and the RQ-5 Hunter. It has rotary engine, and a nonretractable tricycle landing gear for conventional wheeled take-off and landing. The RQ-7A can launch from a catapult and has a tail hook to catch arresting cables for a shorter landing run. A system consists of four RQ-7A UAVs and the associated equipment. The latter includes two Ground Control Stations (GCSs), from where the operators have full control over the UAVs and their sensors.¹⁴

The primary mission payload for the initial (Block 1) RQ-7A production vehicles is an Electro-Optical/Infrared (EO/IR) sensor turret. The improved RQ-7B air vehicle began to roll off AAI Corporation's production line in August 2004. The RQ-7B has larger wings with a more efficient airfoil and increased fuel capacity, allowing an endurance of up to seven hours. Additionally, the vehicle has an enlarged tail, upgraded avionics (including an improved flight controller with an Inertial Measurement Unit

¹³ Andreas Parsch, "Appendix 2: Modern UAVs," *Directory of U.S. Military Rockets and Missiles*, <http://www.designation-systems.net/dusrm/app2/q-7.html>, retrieved 11 April 2007.

¹⁴ Ibid.

(IMU) and increased computing power), and new payload options.¹⁵ Table 2 provides characteristics and performance data for the RQ-7A and RQ-7B.

	RQ-7A	RQ-7B		RQ-7A	RQ-7B
Length	11.2 ft	11.2 ft	Wing Span	12.8 ft	14 ft
Gross Weight	327 lbs	375 lbs	Payload Capacity	60 lbs	60 lbs
Fuel Capacity	51 lbs	73 lbs	Fuel Type	MOGAS	MOGAS
Engine Make	UEL AR-741	UEL AR-741	Power	38 hp	38 hp
Data Link(s)	LOS C2	LOS C2	Frequency	S-band UHF	S-band UHF
	LOS Video	LOS Video		C-band	C-band
Endurance	5 hrs	7 hrs	Max/Loiter Speeds	110/70 kts	105/60 kts
Ceiling	14,000 ft	15,000 ft	Radius	68 NM	68 NM
Takeoff Means	Catapult	Catapult	Landing Means	Arresting Wire	Arresting Wire
Sensor	EO/IR	EO/IR	Sensor Make	Tamam POP 200	Tamam POP 200

Table 2. Shadow RQ-7A/B Characteristics and Performance Data (after)¹⁶

C. EXTENDED RANGE MULTI-PURPOSE (ERMP) UNMANNED AERIAL VEHICLE (UAV)

By 2009, the Army will be able to operate a Medium Altitude Long Endurance (MALE) system, called Extended Range Multi-Purpose (ERMP) UAV, shown in Figure 3.



Figure 3. ERMP UAV (from)¹⁷

¹⁵ Andreas Parsch, "Appendix 2: Modern UAVs," *Directory of U.S. Military Rockets and Missiles*, <http://www.designation-systems.net/dusrm/app2/q-7.html>, retrieved 11 April 2007.

¹⁶ Office of the Secretary of Defense, "Unmanned Aircraft Systems Roadmap 2005-2030," online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, p. 8, retrieved 1 April 2007.

Its primary mission will be to provide persistent intelligence, surveillance and reconnaissance (ISR) in addition to limited direct attack capabilities (via weapons such as the Viper Strike or Hellfire Missile). Warrior, a derivative of the Air Force Predator UAV, will augment and later replace existing systems including the MQ-5B Hunter and Improved-Gnat UAVs. The ERMP will support Army units at division and corps levels. The Army announced the selection of the General Atomics Warrior UAV for the \$1 billion program in August 2005. The ERMP will become operational by 2009. Unlike the current Hunter UAV, the Army plans to field up to 132 General Atomics ERMP UAVs at the division level. Each of the 11 units will contain five ground stations and up to 12 UAVs¹⁸ Table 3 provides characteristics and performance data for the MQ-9A (MALE).

MQ-9A			
Length	36 ft	Wing Span	66 ft
Gross Weight	10,500 lbs	Payload Capacity	750 lbs*
Fuel Capacity	4,000 lbs	Fuel Type	JP
Engine Make	Honeywell TPE 331-10	Power	900 shp
Data Link(s)	BLOS	Frequency	Ku-band
	LOS		C-band
Endurance	30 hrs/clean 16-20 hrs/external stores	Max/Loiter Speeds	225/TBD kts
Takeoff Means	Runway	Landing Means	Runway
Ceiling	50,000 ft	Radius	2,000 NM
Sensor	EO/IR	Sensor Make	MTS-B
	SAR/MTI	Weapons	Four, 500-lb class or 8-10, 250-lb class
*Up to 3,000 lbs total externally on wing hardpoints.			

Table 3. Predator MQ-9A Characteristics and Performance Data (after)¹⁹

¹⁷ Defense Update, "Extended Range Multi-Purpose UAV," *International Online Defense Magazine*, Issue 2, 2005, <http://defense-update.com/products/e/ermpUAV.htm>, retrieved 12 April 2005.

¹⁸ Defense Update, "Extended Range Multi-Purpose UAV," *International Online Defense Magazine*, Issue 2, 2005, <http://defense-update.com/products/e/ermpUAV.htm>, retrieved 12 April 2005.

¹⁹ Office of the Secretary of Defense, "Unmanned Aircraft Systems Roadmap 2005-2030," online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, p. 9, retrieved 1 April 2007.

D. CLASS IV UNMANNED AERIAL VEHICLE (UAV)

The Fire Scout Class IV UAV, shown in Figure 4, provides unprecedented situational awareness, precision targeting and engagement, communications relay and unmanned logistics delivery for FCS and the Army's Future Force.²⁰



Figure 4. Fire Scout, RQ-8A (from)²¹

The Fire Scout program is currently in Engineering, Manufacturing, and Development. Five RQ-8A UAVs and four GCSs are now in developmental testing. Over 100 successful test flights have been accomplished demonstrating autonomous flight, tactical data link operations, multimission payload performance, and GCS operations. The Fire Scout system supports multidimensional air to ground operations and manned/unmanned teaming and connectivity to Army and Joint Reconnaissance, Surveillance, and Target Acquisition (RSTA) and communications systems. The Fire Scout will have the ability to autonomously take off and land at unprepared and unimproved landing zones in close proximity to Corps, Division and Brigade Command Posts and Tactical Operations Centers. Modular mission packages facilitate continued growth into new payloads.

²⁰ Andreas Parsch, "Appendix 2: Modern UAVs," *Directory of U.S. Military Rockets and Missiles*, <http://www.designation-systems.net/dusrm/app2/q-8.html>, retrieved 12 April 2007.

²¹ Andreas Parsch, "Appendix 2: Modern UAVs," *Directory of U.S. Military Rockets and Missiles*, <http://www.designation-systems.net/dusrm/app2/q-8.html>, retrieved 12 April 2007.

In 2003, the Army selected the four-bladed RQ-8B model as its Class IV UAV for FCS. Planned delivery for the first two prototypes is in 2006.²² Table 4 provides characteristics and performance data for the Fire Scout, RQ-8B.

RQ-8B			
Length	22.9 ft	Wing Span	27.5 ft
Gross Weight	3150 lbs	Payload Capacity	600 lbs
Fuel Capacity	1,288 lbs	Fuel Type	JP-5/JP-8
Engine Make	Rolls Royce 250-C20W	Power	420 hp
Data Link(s)	LOS C2	Frequency	Ku-band UHF
	LOS Video		Ku-band
Endurance	6+ hrs	Max/Loiter Speeds	125/0 kts
Ceiling	20,000 ft	Radius	150 NM
Takeoff Means	Vertical	Landing Means	Hover
Sensor	EO/IR/LDRF	Sensor Make	FSI Brite Star

Table 4. Fire Scout RQ-8B Characteristics and Performance Data (from)²³

E. GROUND CONTROL STATIONS (GCSs) AND MISSION PAYLOADS

Several different control stations and devices are required to launch, guide, and recover UAVs. These different stations also allow for the collection, evaluation and dissemination of information provided by the various mission payloads on the respective air vehicles. This section provides a short overview of the various GCSs and mission payloads available for the UAVs described in this thesis.

1. Launch and Recovery Site (LRS)

This term identifies the geographic location from which a particular UAV launches to and returns from mission assignments. ASC-U models the location of each LRS in a given scenario with a grid coordinate system. During the scenario, the location

²² Office of the Secretary of Defense, "Unmanned Aircraft Systems Roadmap 2005-2030," online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, p. 9, retrieved 1 April 2007.

²³ Office of the Secretary of Defense, "Unmanned Aircraft Systems Roadmap 2005-2030," online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, p. 9, retrieved 1 April 2007.

of an LRS may change depending on the mission requirement of the sending unit. Each LRS may be able to support a different number of UAVs. Once launched, flight control for the UAV is passed to a GCS.²⁴

2. Ground Control Station (GCS)

GCSs contain the electronic components required to maintain effective control of UAVs throughout their missions. The station typically consists of a rectangular shelter mounted on a vehicle such as the High Mobility Medium Weight Vehicle (HMMWV). The shelter houses pilot stations and monitors in order to display video, static imagery and sensor data provided by the mission payloads on the UAV. While ASC-U has the capability to model a wide range of GCS configurations, it currently accounts for two types: the One System Ground Control Station (OSGCS), and the Dismounted Controller Device (DCD). These GCSs have the ability to control different types of UAV. They are also limited in the number of platforms that they can control at one time.²⁵

3. Remote Video Terminal

The Remote Video Terminal is a scaled down of a GCS specifically designed to receive telemetry and video data directly from the sending UAV. The Army plans to field the terminals to command posts and TOCs. Intelligence analysts and fire support officers within the TOC can download captured images with telemetry data for further analysis and execution.

An extension of the OSGCS called the One System Remote Viewing Terminal (OSRVT) is a mobile, man-portable terminal planned for the FCS family of systems. This device will highlight an overlay containing icons for identified adversaries to improve target acquisition and identification.

²⁴ Christopher J. Nannini, MAJ, USA “Analysis of the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) Simulation Tool,” Master’s Thesis, Naval Postgraduate School, Monterey, CA, June 2006, p. 20.

²⁵ Christopher J. Nannini, MAJ, USA, “Analysis of the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) Simulation Tool,” Master’s Thesis, Naval Postgraduate School, Monterey, CA, June 2006, p. 23.

4. Mission Payloads

UAVs in the current inventory carry a variety of payloads. ASC-U simulates UAVs with many different types of mission packages depending on the intended mission. These mission packages may include:

- Chemical, Biological, Radiological, Nuclear, Explosive (CBRNE)
- Communications Intelligence (COMINT)
- Electro-Optical/Infrared (EO/IR)
- Electronic Intelligence (ELINT)
- Electronic Warfare (EW)
- Foliage Penetrating (FOPEN) Radar
- GPS Designator
- Laser Designator (LD)
- Laser Rangefinder (LR)
- Light Detection and Ranging (LIDAR) Sensor Technology
- Meteorological Sensors
- Signals Intelligence (SIGINT)
- Synthetic Aperture Radar/Moving Target Indicator (SAR/MTI)
- Weapon Systems²⁶

This chapter provided an overview of the UAVs considered in this thesis and their associated control stations required for launch, guidance, and recovery considered in this thesis. Chapter III introduces ASC-U's approach for solving the dynamic UAV allocation problem.

²⁶ Christopher J. Nannini, MAJ, USA, "Analysis of the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) Simulation Tool," Master's Thesis, Naval Postgraduate School, Monterey, CA, June 2006, p. 24.

III. SIMULATION DESCRIPTION

A. UNMANNED AERIAL VEHICLE (UAV) ALLOCATION PROBLEM

ASC-U is a discrete event simulation that supports the United States Army UAV Mix Analysis. The movement of multiple entities on the modern battlefield complicates military operations. This complexity poses difficulties for UAV allocation and scheduling; however, ASC-U provides feasible UAV flight schedules that are heavily dependent on accurate input data.²⁷

The simulation provides a feasible solution to the following problem: given a scenario that specifies the number and types of UAVs, initial UAV locations, and UAV performance characteristics, identify the number of successful missions and a schedule for each UAV. The solution is constrained by Ground Control Station (GCS) locations and capacities, Launch and Recovery Site (LRS) locations and capacities, Remote Viewing Terminal requirements, and communication platform footprints and capacities. The simulation operates within the Study Analysis Process shown in Figure 5.²⁸

²⁷ Darryl. K. Ahner, Arnold. H. Buss, and John. L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 2.

²⁸ Ibid.

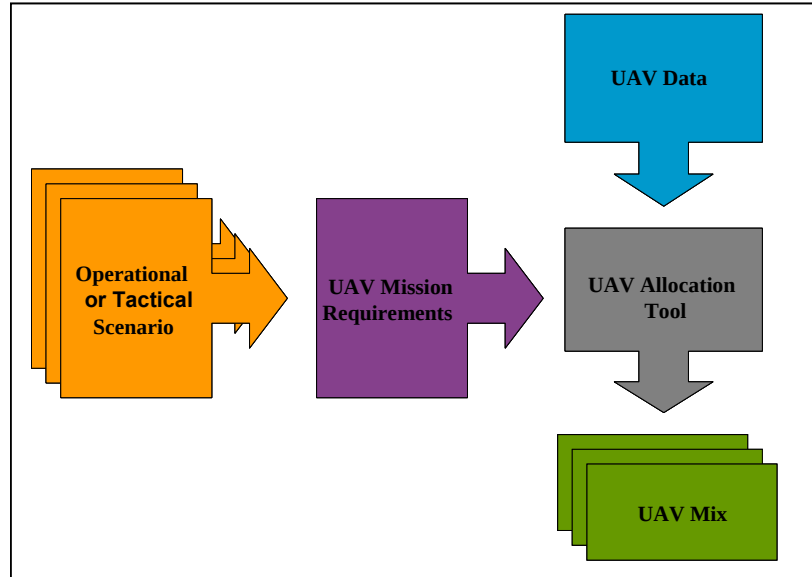


Figure 5. Study Analysis Process Design (after)²⁹

Each scenario includes a set of mission requirements. These mission requirements include geographically fixed sets of sensor, weapon, or communications requirements. The location of the actual requirements does not change during the simulation, but each requirement may have a different time window for which it is available for service by a UAV. The UAV Allocation Tool uses UAV system performance and location data to find feasible UAV schedules. The tool also attempts to maximize value gained from mission requirements as possible within the constraints of the UAV system. A UAV system is defined by a UAV platform, a ground control station (GCSs), and a LRS. These entities may move throughout the scenario. Different scenarios and combinations of UAVs are considered by the application of the UAV Allocation Tool to each scenario.³⁰

ASC-U uses the Dynamic Allocation of Fires and Sensors (DAFS) framework to capture system dynamics. Unlike DAFS, ASC-U is a completely deterministic simulation. For a given set of input data, ASC-U will produce the same UAV flight schedules and Measures of Effectiveness (MOEs). ASC-U takes advantage of the

²⁹ Darryl. K. Ahner, Arnold. H. Buss, and John. L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 2.

³⁰ Ibid., p. 3.

strengths of both simulation and optimization operations research techniques, which creates a model with very unique capabilities. Specifically, ASC-U uses optimization over a fixed time horizon to obtain locally optimal UAV-to-mission area assignments. It simulates the flight of UAVs to their respective mission areas and adjudicates the actual value received by a UAV covering a requirement. Using simulation, UAVs can launch at arbitrary times. This prevents the artifact of discrete time steps that typically occurs in multi-period optimization formulations. Within ASC-U, UAVs are not restricted to launch times that are a multiple of an arbitrary time step. Therefore, UAVs can launch whenever it is most advantageous to do so which creates more realistic scheduling results.³¹

B. TECHNICAL APPROACH

ASC-U periodically optimizes the UAV assignments based on a decision time window in the future. There are two parameters of the ASC-U optimization logic.

- 1) The duration of the decision time window (amount of future time considered in each decision).
- 2) The time between subsequent optimizations.

The user sets these two parameters in the ASC-U input files.

When an optimization event occurs, ASC-U assigns currently available UAVs to missions that are open within a fixed time horizon in the future. This UAV assignment is made to maximize the total value obtained for covering a mission requirement, which has an open period in the fixed time horizon. This time horizon is the same for homogeneous UAVs, but differs for those that are heterogeneous. This difference is a result of the fact that ASC-U considers the maximum cycle time for each UAV type. A UAV's maximum cycle time is the sum of its maximum endurance and recovery time. The maximum cycle time is multiplied by a constant between 1.0 and 1.5, which is determined experimentally.³²

³¹ Darryl. K. Ahner, Arnold. H. Buss, and John. L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 2.

³² *Ibid.*, p. 4.

For example, consider two UAVs assigned to four missions as shown in Figure 6. ASC-U performs optimal UAV assignments at predetermined time intervals. [1] At time t_0 , assume that two UAVs are available, UAV1 and UAV2. The model considers all available UAVs with GCS control capacity and all missions with value in the fixed time horizon window (missions 1, 2, 3, but not 4 at time t_0). [2] A UAV is assigned to a mission and can service any sensor requirement for that mission if the UAV is carrying the proper sensor. Assume that the optimal assignment is for UAV1 to cover mission 1 and UAV2 to cover mission 2. ASC-U will launch UAV1 immediately to arrive at mission 1 as soon as possible. [3] UAV2 does not launch, but is scheduled for launch to arrive just in time for the start of mission 2. At time t_1 , UAVs that are scheduled for launch, but have not been launched are “unassigned” and considered for assignment (UAV2 is unassigned). [4] UAV2 is assigned to mission 2 again³³ and scheduled to launch just in time. UAV assignments do not change after they launch.³⁴ This entire process is depicted in the Figure 6.

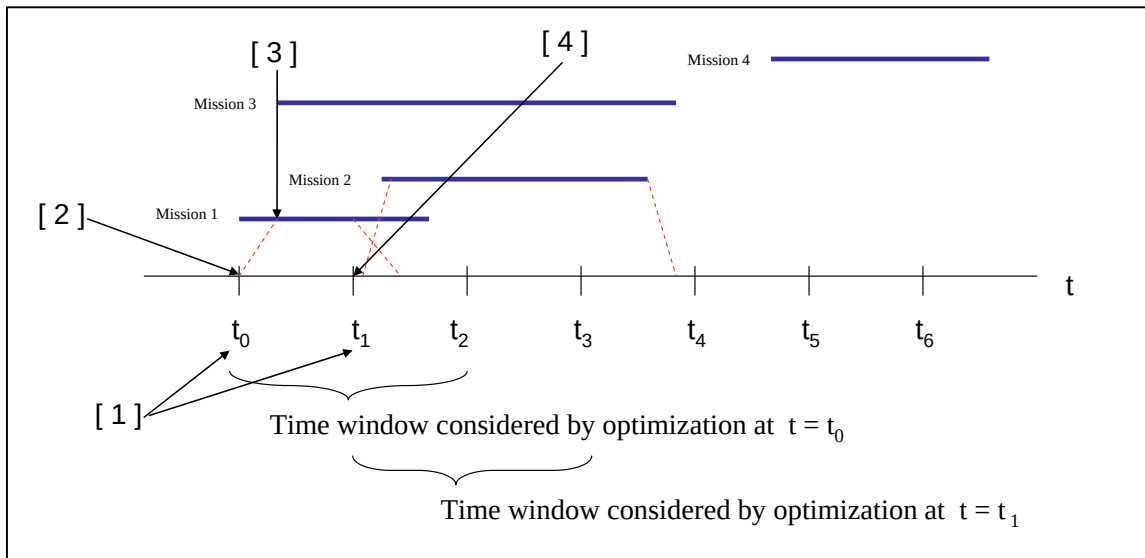


Figure 6. Assignments Dynamics Example (after)³⁵

³³Darryl. K. Ahner, Arnold. H. Buss, and John. L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 2.

³⁴ Ibid., p. 5.

³⁵ Ibid.

As time advances in the simulation, each UAV has a flight schedule built that reflects the optimal assignment for available UAVs. These flight schedules cover a fixed, future time horizon. A user defined value rate determines the value of each mission covered by a UAV. The user defined value rate depends on 1) the force echelon requiring the mission; or 2) the priority given to a mission; or 3) a combination of both. A UAV receives a total value for covering a mission through a three-step process. First, the simulation makes a time-on-station calculation for the UAV. Next, the UAV time-on-station is multiplied by the user defined value rate, as long as the time window for the mission requirement has not closed. Finally, if the time window for the mission requirement closes prior to UAV arrival, only the user defined value rate is gained (no time-on-station multiplier in this case). The sum of these values gives an indication of the potential value of each mission if a UAV was assigned. If a mission area is not within the control footprint of a GCS with available capacity the value is set to zero.³⁶

Figure 7 shows the various stages that ASC-U uses to build UAV flight schedules.

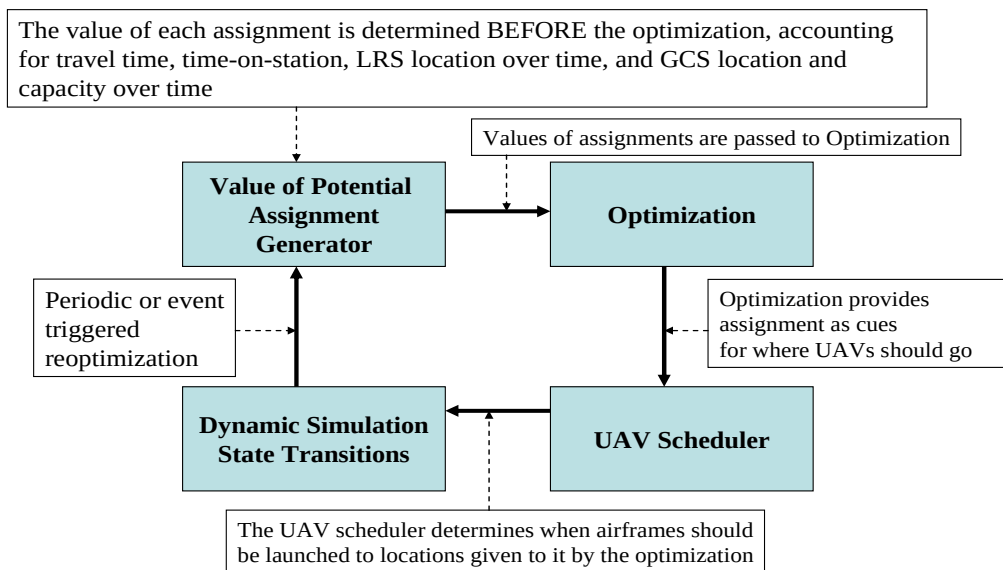


Figure 7. ASC-U Dynamic Cueing and Transition (after)³⁷

³⁶ Darryl. K. Ahner, Arnold. H. Buss, and John. L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 2.

³⁷ Ibid., p. 6.

Given a current system state, in **Dynamic Simulation State Transitions**, a periodic optimization event is added to the simulation event list. This optimization event uses the **Value of Potential Assignment Generator** to calculate the value received within the UAV-type time windows for every possible assignment for each UAV. This value accounts for movement of simulation entities and GCS range constraints. These assignment values, along with GCS capacity constraints, flow to the **Optimization**. The Optimization computes the maximum value for this allocation problem and passes the assignments to the **UAV scheduler**. The UAV Scheduler launches UAVs to arrive just in time if the difference between the mission start time and the transit time to the mission area is greater than the current time. The next optimization event is scheduled to occur after a delay of the optimization time interval. The state transitions take place until the event occurs, so the model will generally have a very different state than that of the previous optimization event.³⁸

The following optimization problem must be solved at each optimization event. Let A be the set of all mission areas with at least one emission active within the global time horizon.

Let L be the set of all active LRSs

Let G be the set of all GCSs

Let G_L be the set of all GCSs assigned to LRS L

Let c_g be the number of UAVs GCS g can control

Let I_l be the subset of all UAVs at LRS $l \in L$ determined as follows:

For each LRS $l \in L$, get n UAVs, where n is the minimum of: the number of ready UAVs, the LRS launch limit on the number of UAVs airborne, or the number of UAVs controllable by GCSs assigned to LRS l ($\sum \{ c_g \mid g \in G_L \}$)

(Note that n could be set to all UAVs that are ready without changing the answer.

³⁸ Darryl K. Ahner, Arnold. H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 6.

Restricting it by remaining LRS and GCS capacity reduces the size of the problem. All UAVs at an LRS have the same capabilities and are of the same type.)

Let J_l be the subset of all sensor packages currently located at LRS $l \in L$

Let $Y_{ga} = 1$ if mission area a is assigned to GCS g , 0 otherwise (by a heuristic discussed below)

Let c_{ja} be the reward for a UAV with sensor package j being assigned to mission area a from the soonest possible arrival time of the UAV at the area to the end of the time horizon, $t + \Delta t^i$, for UAV i

Let $X_{ja} = 1$ if a UAV with sensor package j is assigned to mission area a , 0 otherwise

Then the optimization formulation is:³⁹

$$\max \sum_{ja} c_{ja} X_{ja} \quad (1)$$

$$\text{s.t. } \sum_j X_{ja} \leq 1 \quad \forall a \in A \quad (2)$$

$$\sum_a X_{ja} \leq 1 \quad \forall j \in J \quad (3)$$

$$\sum_{ja} Y_{ga} X_{ja} \leq c_g \quad \forall g \in G \quad (4)$$

$$\sum_{j \in J_l} X_{ja} \leq |I_l| \quad \forall l \in L \quad (5)$$

$$X_{ja} \in \{0,1\}$$

$$Y_{ga} \in \{0,1\} \text{ determined by a heuristic}$$

- (1) Maximize the value of Mission Areas covered.
- (2) Assign only 1 UAV per Mission Area.
- (3) Assign only 1 Mission Area per UAV.
- (4) Do not exceed the GCS control limit.
- (5) The number of sensors assigned cannot exceed the number of UAVs available to carry them.

³⁹ Darryl K. Ahner, Arnold. H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 7.

The following heuristic determines the value of Y_{ga} used in the ASC-U optimization logic (assignment of Mission Areas to GCSs):

For each LRS

For each Mission Area a

Let N_a be the number of GCSs g that are in range of a UAV assigned to Mission Area a

For each Mission Area a sorted by N_a

For each GCS g , sorted by $\sum_a Y_{ga} \quad \forall g$ (so far)

If a is in range of g set $Y_{ga} = 1$, next Mission Area⁴⁰

C. CONSTRAINTS, LIMITATIONS, AND ASSUMPTIONS

In this section, constraints limitations and assumptions made during ASC-U development are identified. Constraints represent uncontrollable conditions that influenced the development of the simulation. Limitations define conditions that the ASC-U developers defined to help scope the problem. Assumptions are choices made by the developers to simplify the problem into a manageable set of components in order to generate a solution.⁴¹

1. Constraints

Two conditions served as constraints during the development of ASC-U: problem type and time. This type of problem combines vehicle assignment and routing. In the mathematical programming lexicon, these types of problems have been identified as “Non-Deterministic Polynomial Hard” or “NP-Hard,” which hinders an analyst’s ability to find a solution in a polynomial number of computational steps. The difficulty lies in the process of enumerating all the UAV mission assignments in order to develop the optimal schedule in a reasonable amount of time.⁴²

⁴⁰ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 7.

⁴¹ Ibid., p. 9.

⁴² Ibid.

The developers were also limited by the amount of time available to build the simulation. Once developed, ASC-U proved valuable to the Army's UAV Mix Analysis. The simulation can easily accept new problem characteristics and provides flexibility for future studies.⁴³

2. Limitations

The following is a list of limitations from the ASC-U Users/Analyst Manual.⁴⁴

- ASC-U represents mission areas as a single Cartesian coordinate for the entire mission window. If more than one UAV is required to perform a mission, the user must provide sensor requirements at two different/adjacent coordinates. The use of single coordinates for mission areas account for the first order effects of assignments over space and time and GCS capacity and range constraints.
- ASC-U does not account for tactics, techniques, and procedures concerning the employment of UAVs as they serve mission areas.
- The UAV remains at the mission area until either all sensor requirements that the UAV can perform end or the UAV must return to its LRS so that it does not exceed its operating time.
- ASC-U does not model passing UAV control between two GCSs. A single GCS controls a UAV throughout its entire mission.
- The user defines the contents of a sensor package for each UAV type prior to running the simulation. Sensor packages cannot be reconfigured during the simulation.
- The user provides LRS and GCS locations for particular times throughout the length of the scenario. LRSs and GCSs travel at a uniform rate between the locations during the prescribed times.

⁴³ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 7.

⁴⁴ *Ibid.*, p. 10.

3. Assumptions

The following is a list of assumptions from the ASC-U Users/Analyst Manual. The developers note that some limitations may be considered as assumptions in order to highlight their impact on the solution.⁴⁵

- The overall mission value for the scenario is the sum of the sensor requirement values for served mission areas making up the solution.
- UAV airspeed and operating radius are constant.
- UAV control and data link requirements are satisfied.
- Transition times between flights are constant.
- Transition times represent recovery of UAVs and include swapping payloads.
- The simulation does not consider attrition of or damage to UAVs. Additionally, maintenance requirements are not considered.
- Army Airspace Command and Control (A2C2) requirements are satisfied.
- The locations of the GCSs and LRSs are extracted from a Map Exercise (MAPEX) or a simulation running on an approved scenario and are known throughout the scenario.
- LRSs and GCSs travel at a uniform rate between locations from start to end times.
- GCSs often operate in pairs, employing a “leap frog” movement where one controls UAVs from a forward position while the other GCS moves to its next position. ASC-U replicates this method of employment by representing two GCSs by a single GCS that operates continuously with uniform movement between positions. This modeling technique allows for uninterrupted GCS capacity and control between positions.
- LRSs are within range of at least one GCS for launch and recovery operations.

⁴⁵ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 10..

- Terrain features do not influence UAV mission accomplishment. ASC-U does not model terrain features within the simulation.
- Atmospheric conditions do not influence UAV mission accomplishment. ASC-U does not model atmospheric conditions within the simulation.⁴⁶

D. ASC-U INPUT

The mission requirements and UAV performance characteristics are contained in either a spreadsheet or a database. The user loads the spreadsheet or database into ASC-U prior to each simulation run. The input file contains several worksheets or multiple tables; each provides data for a specific component of the mission requirements. Table 5 displays the worksheets included in the spreadsheet file.

Worksheet Title
MetaData
Mission
AirFrame
MissionPackageType
MissionPackageLocation
GCSInput
LRSInput
RVTInput
SuperMissionData

Table 5. ASC-U Input Worksheets (after)⁴⁷

The following sections provide a description of the worksheet components used for analysis in this thesis. For a complete description of each input worksheet used within ASC-U, see the most recent version of the *ASC-U Users/Analysts Manual*.⁴⁸

⁴⁶ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 10.

⁴⁷ Ibid., p. 12.

⁴⁸ Ibid.

1. Meta Data Worksheet

The MetaData Worksheet contains six elements: Scenario Length, Optimization Interval, Report Interval, Early Return, Secondary Areas, and Append Areas. In this thesis, we examined the Optimization Interval, Early Return, Secondary Areas and Appended Areas.

- **Scenario Length** indicates the length of the scenario in simulation hours (defaults to 360.0, if MetaData worksheet is not present).
- **Optimization interval** is the time (simulation hours) between optimizations that derive the UAV allocations at that moment in time (optional: defaults to 1.0).
- **Report Interval** indicates the phase of the scenario for which measures of performance will be reported. Measures of Performance for the overall scenario length are also provided in addition to these interval reports (optional: defaults to 4.0).
- **Early Return, Secondary Areas** and **Append Areas** are binary heuristics that give UAVs in the simulation the opportunity to gain additional mission value if conditions allow. (Early Return and Append Areas are mutually exclusive.)⁴⁹

2. Air Frame Worksheet

The Air Frame worksheet contains all of the performance characteristics for each UAV in the scenario. Each row in the worksheet represents one UAV. For this research, the value in each row was varied under the following column headings: transition time, maximum speed,⁵⁰ operating radius, operating time and time horizon. The following lists describe the column headings in the Air Frame worksheet.⁵¹

⁴⁹ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 14.

⁵⁰ Ibid.

⁵¹ Ibid., pp. 14-15.

- **Type** is the class of UAV and may be used to ensure that a proper LRS is used for take off and landing.
- **Unit** is the owning military unit of the UAV listed in the same row and under the column Type.
- **Compatible GCS** indicates the type of GCS from the GCS worksheet that is able to control this UAV. (Not used in the current version, but should not be deleted.)
- **Starting LRS** is the name from the LRSInput Worksheet of the LRS from which the UAV originates.
- **Ending LRS** is the name from the LRSInput Worksheet of the LRS to which the UAV lands. (Should be the same as the Starting LRS.)
- **Transition time** is the time, upon landing, that is required before the UAV is available to launch.
- **Max speed** is the travel rate of the UAV between locations.
- **Operating radius** is the maximum distance the UAV can be from a controlling GCS for effective guidance.
- **Name** is a unique identifier for the UAV.
- **StartAvail** is the time the UAV is initially available for launch, usually H-hour.
- **EndAvail** is the time the UAV is last available in the scenario.
- **Operating time** is the maximum time the UAV may be away from an LRS.
- **Time horizon** is the time window used in the rolling algorithm of the optimization. Time horizon is usually 1.0 to 1.5 times the operating time plus transition time. This value is critical in determining which missions will be considered in the look-ahead time window to launch a UAV if GCS capacity is available.⁵²

⁵² Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 15.

E. ASC-U OUTPUT

ASC-U stores the simulations results in a database file. The output reports are captured in 14 database tables. Table 6 provides a short description for each Measure of Effectiveness (MOE). For this research, the data required is contained in the Coverage and CoverageByType database tables. The Measures of Performance are provided by output reports shown in Table 6.

Table Name	Description
Coverage	Amount of time each mission is covered by a UAV
CoverageByType	Amount of time each mission is covered by each different type of UAV
CoverageBy Delay	The delay from the time a mission opens until it is first covered by a UAV
GCSLoadOverTime	The time average number of UAVs using each GCS in each time period
MissionAssignment	A schedule of which UAV is covering which missions
MissionPackageUtilization	The time average utilization rate of each type of MissionPackage at each LRS
RunInformation	Information about the model version and data used to create the output tables
RVTCoverage	Not currently used
Schedule	A schedule of the UAV sorties
TerseRVTCoverage	A summary of the coverage of each mission by each RVT
UAVReadyTimeOverTime	For each UAV, the amount of time it is ready for launch in each time period
UAVUtilization	UAV utilization data
UAVUtilizationOverTime	For each UAV, the proportion of ready time that the UAV was airborne in each time period

Table 6. Measures of Performance Tables (after)⁵³

This chapter provided some insight on the UAV allocation problem and some details on the ASC-U discrete event simulation. The descriptions are limited to those aspects of the model that pertain to this thesis. ASC-U is a dynamic model and the

⁵³ Darryl K. Ahner, Arnold H. Buss, and John L. Ruck, *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006, p. 19.

developers are regularly working to improve the fidelity and utility of the simulation. This thesis uses ASC-U version 1.4.8 for observation and analysis. Chapter IV will introduce an alternative optimization model for the ASC-U optimization logic.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. AN ALTERNATIVE OPTIMIZATION MODEL

The preceding chapter introduced the current optimization formulation for the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) simulation. The current optimization logic relies on a heuristic to assign Mission Areas to Ground Control Stations (GCSs). This thesis develops an alternative optimization model for ASC-U that does not rely on a heuristic. Specifically, this mathematical programming model is a Binary Integer Program that seeks to maximize a linear objective function.

A. INTEGER PROGRAMMING

Integer Programming is a mathematical programming technique used to solve optimization problems with discrete or integer variables. Such variables are used to model indivisibilities, and binary (0 or 1) variables are used to represent on/off decisions to purchase, invest, select, and so on. Problems that lend themselves to integer solutions include developing train or aircraft schedules, planning work in a production line or maintenance team, or planning nationally or regionally the daily or weekly production of electricity.⁵⁴

More specifically, a linear program is defined by the following:

$$\max \{cx : Ax \leq b, x \geq 0\},$$

where A is in m by n matrix, c is an n -dimensional row vector, b is an m -dimensional column vector, and x is an n -dimensional column vector of variables or unknowns. Now, we include the restriction that certain variables must take on integer values.

If some, but not all, of the variables are integer, we have a **(Linear) Mixed Integer Program** (MIP), written as:

$$\begin{aligned} &\max cx + hy \\ &s.t. \quad Ax + Gy \leq b \\ &\quad x \geq 0, y \geq 0 \text{ and integer} \end{aligned}$$

⁵⁴ Laurence A. Woolsey, *Integer Programming*, Wiley-Interscience Publication, 1998, p. xiii.

where A is again m by n , G is m by p , h is a p row-vector, and y is a p column-vector of integer variables.⁵⁵

If all variables are integer, we have a **(Linear) Integer Program**, written as:

$$\begin{aligned} \max \quad & cx \\ \text{s.t.} \quad & Ax \leq b \\ & x \geq 0 \text{ and integer} \end{aligned}$$

and if all the variables are restricted to 0 or 1 values, we have a 0-1 or **Binary Integer Program (BIP)** written as:⁵⁶

$$\begin{aligned} \max \quad & cx \\ \text{s.t.} \quad & Ax \leq b \\ & x \in \{0, 1\}^n \end{aligned}$$

B. UNMANNED AERIAL VEHICLE (UAV) ALLOCATION MODEL

The alternative optimization model takes as input a collection of UAVs, Launch and Recovery Sites (LRSs), Mission Packages, Mission Areas, and GCSs. We further divide these entities into the following subsets: UAVs at each LRS, Mission Packages available at each LRS, Mission Areas reachable from each LRS, and GCSs assigned to each Mission Area. The data required for the model includes the capacity of each GCS, the value received by a UAV for satisfying a mission requirement over a designated mission area, an upper limit on the number of mission packages available at each LRS, and an upper limit on the number of UAVs at each LRS. The BIP contains a single variable. This variable is set to a value of 1 if a UAV at a particular LRS with a given Mission Package is assigned to a designated Mission Area and controlled by the proper GCS. Otherwise, the binary variable is set to 0. The entire mathematical programming formulation defines the indices, subsets, parameters, decision variable, constraints, and objective function.

⁵⁵ Laurence A. Woolsey, *Integer Programming*, Wiley-Interscience Publication, 1998, p. 3.

⁵⁶ Ibid.

1. Indices

The indices used to define this model are:

- $u \in U$ UAV u
- $l \in L$ Launch and Recovery Site l (LRS)
- $j \in J$ Mission Package j (msnPkg)
- $a \in A$ Mission Area a (msnArea)
- $g \in G$ Ground Control Station g (GCS)

2. Subsets

The subsets used in this model are:

- U_l Subset of UAVs at LRS l
- J_l Subset of msnPkgs available at LRS l
- A_l Subset of msnAreas reachable from LRS l
- G_a Subset of GCSs assigned to msnArea a

3. Parameters

The parameters used to define the data for this model are:

- c_g Capacity of GCS g
- v_{ulja} Value recieved by UAV u at LRS l with msnPkg j assigned to msnArea a
- n_{lj} Upper limit on the number of msnPkgs j at LRS l
- m_{lu} Upper limit on the number of UAVs u at LRS l

4. Decision Variable

The model uses the following binary variable:

$$x_{uljag} = \begin{cases} 1, & \text{if UAV } u \text{ at LRS } l \text{ with msnPkg } j \text{ is assigned to} \\ & \text{msnArea } a \text{ and controlled by GCS } g \\ 0, & \text{otherwise} \end{cases}$$

The binary decision variable contains an index for each of the key relationships that must be maintained in order for the model to provide an optimal solution.

5. Constraints

The model requires five different constraints.

The following first constraint ensures that: For each LRS l and Mission Package j , the sum over UAVs u in the subset of UAVs at LRS l , the sum over Mission Areas a in the subset of Mission Areas reachable from LRS l , and the sum over GCS g in the subset of GCSs assigned to Mission Area a of the variable x , must not exceed the number of Mission Packages j at LRS l . **(For each type of Mission Package at each LRS, do not employ more Mission Packages than are available.)**

$$\sum_{u \in U_l} \sum_{a \in A_l} \sum_{g \in G_a} x_{uljag} \leq n_{lj} \quad \forall l, j$$

The following second constraint ensures that: For each LRS l , the sum over UAVs u in the subset of UAVs at LRS l , the sum over Mission Packages j in the subset of Mission Packages at LRS l , the sum over Mission Areas a in the subset of Mission Areas reachable from LRS l , and the sum over GCS g in the subset of GCSs assigned to Mission Area a of the variable x , must not exceed the number of UAVs u at LRS l . **(For each LRS, do not launch more UAVs than are available at that LRS.)**

$$\sum_{u \in U_l} \sum_{j \in J_l} \sum_{a \in A_l} \sum_{g \in G_a} x_{uljag} \leq m_{lu} \quad \forall l$$

The following third constraint ensures that: For each Mission Area a , the sum over UAVs u in the subset of UAVs at LRS l , the sum over LRS l in the set of LRSs, the sum over Mission Packages j in the subset of Mission Packages at LRS l , and the sum over GCS g in the subset of GCSs assigned to Mission Area a , of the variable x , must not exceed 1. **(For each Mission Area, do not assign more than one UAV.)**

$$\sum_{u \in U_l} \sum_{l \in L} \sum_{j \in J_l} \sum_{g \in G_a} x_{uljag} \leq 1 \quad \forall a$$

The following fourth constraint ensures that: For each UAV u , the sum over LRS l in the set of LRSs, the sum over Mission Packages j in the subset of Mission Packages at LRS l , the sum over Mission Areas a in the subset of Mission Areas reachable from LRS l , and the sum over GCS g in the subset of GCSs assigned to Mission Area a of the variable x , must not exceed 1. **(For each UAV, do not assign more than one Mission Area.)**

$$\sum_{l \in L} \sum_{j \in J_l} \sum_{a \in A_l} \sum_{g \in G_a} x_{uljag} \leq 1 \quad \forall u$$

The following fifth constraint ensures that: For each GCS g , the sum over UAVs u in the subset of UAVs at LRS l , the sum over LRS l in the set of LRSs, the sum over Mission Packages j in the subset of Mission Packages at LRS l , and the sum over Mission Areas a in the subset of Mission Areas reachable from LRS l , of the variable x , must not exceed the capacity of GCS g . **(For each GCS, do not exceed its capacity.)**

$$\sum_{u \in U_l} \sum_{l \in L} \sum_{j \in J_l} \sum_{a \in A_l} x_{uljag} \leq c_g \quad \forall g$$

6. Objective Function

The objective function for the UAV Allocation Model maximizes, by choice of the binary decision variable x , the sum over UAVs u in the subset of UAVs at LRS l , the sum over LRS l in the set of LRSs, the sum over Mission packages j in the subset of Mission Packages at LRS l , the sum over Mission Areas a in the subset of Mission Areas reachable from LRS l , and the sum over GCS g in the subset of GCS assigned to Mission Area a , of the scalar mission value v multiplied by the variable x . **(Maximize the value that each UAV receives for coverage over a Mission Area.)**

$$\max_{x \in \{0,1\}} \sum_{u \in U_1} \sum_{l \in L} \sum_{j \in J_1} \sum_{a \in A_1} \sum_{g \in G_a} v_{ulja} x_{uljag}$$

The full formulation for the UAV Allocation Model is:

$$\begin{aligned} & \max_{x \in \{0,1\}} \sum_{u \in U_1} \sum_{l \in L} \sum_{j \in J_1} \sum_{a \in A_1} \sum_{g \in G_a} v_{ulja} x_{uljag} \\ & \text{s.t.} \quad \sum_{u \in U_1} \sum_{a \in A_1} \sum_{g \in G_a} x_{uljag} \leq n_{lj} \quad \forall l, j \\ & \quad \sum_{u \in U_1} \sum_{j \in J_1} \sum_{a \in A_1} \sum_{g \in G_a} x_{uljag} \leq m_{lu} \quad \forall l \\ & \quad \sum_{u \in U_1} \sum_{l \in L} \sum_{j \in J_1} \sum_{g \in G_a} x_{uljag} \leq 1 \quad \forall a \\ & \quad \sum_{l \in L} \sum_{j \in J_1} \sum_{a \in A_1} \sum_{g \in G_a} x_{uljag} \leq 1 \quad \forall u \\ & \quad \sum_{u \in U_1} \sum_{l \in L} \sum_{j \in J_1} \sum_{a \in A_1} x_{uljag} \leq c_g \quad \forall g \end{aligned}$$

C. PRELIMINARY RESULTS

In order to test the UAV Allocation Model, we implemented the model using the General Algebraic Modeling System (GAMS) with the following input test data in Table 7.

Entity	Quantity
UAVs	20
LRS	20
Mission Packages	20
Mission Area	20
GCS	20
Value received by UAV u for coverage	Random(20, 250)
Capacity of each GCS	1
Limit on Mission Packages j at LRS l	5
Limit on UAVs u at LRS l	6

Table 7. Table of Test Values for UAV Allocation Model

Table 8 is an output product from GAMS that shows the relationships between the subsets in the UAV Allocation Model. The actual assignment in GAMS is reflected by a “YES” in the appropriate row and column of the table.

---- 140 SET U_l						
	uav1	uav2	uav3	uav4	uav5	uav6
lrs1	YES	YES				
lrs2			YES	YES		
lrs3					YES	YES
+ uav7 uav8 uav9 uav10						
lrs4	YES	YES				
lrs5			YES	YES		
---- 140 SET A_l						
	msnArea1	msnArea2	msnArea3			
lrs1	YES	YES	YES			
lrs2	YES	YES	YES			
lrs3	YES	YES	YES			
lrs4	YES	YES	YES			
lrs5	YES	YES	YES			
---- 140 SET J_l						
	msnPkg1	msnPkg2	msnPkg3	msnPkg4	msnPkg5	msnPkg6
lrs1	YES	YES				
lrs2			YES	YES		
lrs3					YES	YES
+ msnPkg7 msnPkg8 msnPkg9 msnPkg10						
lrs4	YES	YES				
lrs5			YES	YES		
---- 140 SET G_a						
	gcs1	gcs2	gcs3	gcs4	gcs5	
msnArea1	YES					
msnArea2		YES				
msnArea3			YES			
msnArea4				YES		
msnArea5					YES	

Table 8. Display of GAMS Output for Subsets of the UAV Allocation Model

We implemented the model using the XA Solver in GAMS. The optimal integer solution for the model using the limited test data mentioned previously is shown in Table 9.

Solver	MIP Solution	Best Possible	OPTCR	Relative Gap
XA	687.365341	687.365341	0.000000	0.000000

Table 9. GAMS Solution for UAV Allocation Model

The Optimality Criteria (OPTCR) is a relative measure of optimality. This option sets a relative termination tolerance for problems containing discrete variables, which means that the solver will stop and report on the first solution found whose objective value is within $100 \times \text{OPTCR}$ of the best possible solution (i.e., $\text{OPTCR} = 0.05$ requires the solution to be within 5% of optimal).⁵⁷ This output indicates that the UAV Allocation Model certainly provides an optimal solution to a UAV assignment problem without the use of a heuristic. This model also shows promise for scalability and quick solution time for those problems that are more robust than our small test example. (The full GAMS implementation of the model and solution is included in Appendix B).

Although GAMS solves the BIP to optimality, ASC-U does not use GAMS for optimization. ASC-U relies on an open source (GNU lesser general public license) Mixed Integer Linear Program (MILP) solver called `lp_solve` for optimization. `lp_solve` is based on the revised simplex method and the Branch-and-Bound method for integers. It solves pure linear, integer/binary, semi-continuous and special ordered sets (SOS) models. The emphasis here is on the word “linear,” so equations must be of the first order. For example, $5x - 3y$ is linear; however, $x \cdot y$ (two variables) is nonlinear and cannot be handled by `lp_solve`. Both the objective function and the constraints have this restriction.⁵⁸ In order to demonstrate that the UAV Allocation Model can satisfactorily

⁵⁷ Richard E. Rosenthal, *GAMS – A User’s Guide*, GAMS Development Corporation, Washington, D.C., 2006, p. 218.

⁵⁸ Kjell Eikland and Peter Notebaert, “Introduction to `lp_solve` 5.5.0.10,” <http://lpsolve.sourceforge.net/5.5/>, retrieved 14 April 2007.

replace the existing ASC-U optimization logic, we solved the BIP using lp_solve as well. The lp_solve result, using the exact same test data that we used for GAMS, is in Table 10.

Model name: 'GAMSMOD' - run #1			
Objective: Minimize(R00000000)			
SUBMITTED			
Model size:	156 constraints,	2321 variables,	2621 non-zeros.
Sets:		0 GUB,	0 SOS.
Using DUAL simplex for phase 1 and PRIMAL simplex for phase 2.			
The primal and dual simplex pricing strategy set to 'Devex'.			
Relaxed solution	-687.3653411 after	8 iter	is B&B base.
Feasible solution (gap 0.0%)	-687.3653411 after	8 iter,	0 nodes
-Optimal solution (gap 0.0%).	-687.3653411 after	8 iter,	0 nodes
Excellent numeric accuracy * = 0			
MEMO: lp_solve version 5.5.0.5 for 32 bit OS, with 64 bit REAL variables.			
In the total iteration count 8, 0 (0.0%) were bound flips.			
There were 0 refactorizations, 0 triggered by time and 0 by density.			
... on average 8.0 major pivots per refactorization.			
The largest [LUSOL v2.2.1.0] fact(B) had 157 NZ entries, 1.0x largest basis.			
The maximum B&B level was 1, 0.0x MIP order, 1 at the optimal solution.			
The constraint matrix inf-norm is 243.62, with a dynamic range of 243.62.			
Time to load data was 0.422 seconds, presolve used 0.015 seconds,			
... 0.032 seconds in simplex solver, in total 0.469 seconds.			

Table 10. lp_solve Solution for UAV Allocation Model

lp_solve produces the same optimal solution for the model as GAMS, with excellent numerical accuracy, and in minimal time (the negative sign on the optimal solution results from the fact that lp_solve converts the maximization problem into a minimization problem prior to solving).

Finally, it is important to note that, although the UAV Allocation Model shows promise for use in ASC-U, the model has not yet been implemented in the ASC-U discrete event simulation.

This chapter introduced an alternative optimization model for ASC-U. Chapter V will discuss our methodology, Design of Experiment (DOE), and analysis of the output from our simulation runs.

V. DATA ANALYSIS

This chapter describes the research methodology and analysis of our data. Section A gives an overview of Design of Experiments (DOEs) for simulation studies. Section B outlines the Measures of Effectives (MOEs) provided in the ASC-U output tables. Section C discusses the analysis technique used in this chapter. Section D presents a regression model for the ASC-U Total Value MOE. In Section E, we provide sensitivity analysis for how UAV coverage may degrade in the presence of adverse random events.

A. DESIGN OF EXPERIMENT (DOE)

When conducting simulation studies, experimenters and analysts often have to make decisions on input values. Typically, there are very large input spaces and a finite amount of computing resources available to perform simulation runs. **Experimental design** provides a structured way for analysts to decide (before making simulation runs) which configurations to simulate so that the desired output is obtained with the fewest possible number of runs. These carefully designed experiments are more efficient than arbitrary “hit-or-miss” techniques that simply use a large number of unsystematic runs.⁵⁹

Although simulation experiments can be viewed as a subset of experimentation in general, there are some peculiarities regarding simulation experiments that distinguish them from traditional physical, industrial, laboratory, or agricultural experimental design found in most DOE literature.

- Simulation experiments offer control over factors that, in reality, are not controllable (like customer arrival rates or system performance improvements). Thus, an experimenter can explore a wider range of possibilities than normally possible in a traditional physical experiment.

⁵⁹ Averill M. Law and W. David Kelton, *Simulation Modeling and Analysis*, 3rd Ed., McGraw-Hill, 2000, p. 623.

- In simulation studies, analysts can control the basic source of variation (through random number generators), unlike physical experiments.⁶⁰
- Randomization of treatments is prudent for physical experiments to guard against systematic variation. In simulation experimentation, this is not required as long as an acceptable pseudo-random number generator is used.

1. Factor Selection

In experimental design terminology, the input parameters are called factors, and the output measures are called responses. There are usually several different parameters or performance measures that are of interest to an experimenter. Factors can be quantitative or qualitative, controllable or uncontrollable.⁶¹ ASC-U relies on multiple input factors to model a given scenario and allocate UAVs to mission areas. The factors we selected for this study fell into two categories: 1) UAV performance characteristics; and 2) model parameters.

Since ASC-U solves a dynamic routing problem to assign UAVs, it is logical to conclude that the UAV characteristics (Transition Time, Maximum Speed, Operating Radius, and Operating Time) would be influential factors.

As mentioned in Chapter III, ASC-U has two optimization parameters: the Optimization Interval (time between subsequent reoptimizations) and the Time Horizon (duration of the decision time window). The optimization interval determines how frequently the simulation should poll UAVs to determine their availability and assign them to open mission areas in respective time horizons (i.e., an optimization interval of 0.5 triggers an optimization event every one-half simulation hour). Because of the importance of these two inputs to the overall solution, we included them both as influential factors. We vary the Optimization Interval from 0.1 to 0.9 for this study.

The last factors we included were three Boolean heuristic settings contained the ASC-U simulation: 1) Early Return (UAV returns home or stays out until operating time

⁶⁰ Averill M. Law and W. David Kelton, *Simulation Modeling and Analysis*, 3rd Ed., McGraw-Hill, 2000, pp. 622-623.

⁶¹ Ibid.

ends); 2) Secondary Areas (UAV flies to gain more value from a secondary mission); and 3) Appended Areas (UAV completes primary mission, and then looks for another mission to gain additional value). Table 11 shows the factors and their base levels used in the DOE. After conferring with our sponsors, we determined that it would be reasonable to compute factor ranges by adding and subtracting 20% of the base value for each.

UAV Type	ASC-U Factor	DOE Identifier	Base Value	Factor Ranges	
				Low	High
ERMP					
	Transition Time (hrs)	TransE	1.5	1.2	1.8
	Max Speed (km/hr)	SpeedE	241	192.8	289.2
	Operating Radius (km)	RadiusE	500	400	600
	Operating Time (hrs)	OperE	36	28.8	43.2
	Time Horizon (hrs)	HorizE	45	36	54
Class IV					
	Transition Time (hrs)	Trans4	1	0.8	1.2
	Max Speed (km/hr)	Speed4	231	184.8	277.2
	Operating Radius (km)	Radius4	75	60	90
	Operating Time (hrs)	Oper4	6	4.8	7.2
	Time Horizon (hrs)	Horiz4	7.5	6	9
SUAV					
	Transition Time (hrs)	TransS	0.16	0.13	0.19
	Max Speed (km/hr)	SpeedS	84	67.2	100.8
	Operating Radius (km)	RadiusS	16	12.8	19.2
	Operating Time (hrs)	OperS	1.33	1.06	1.6
	Time Horizon (hrs)	HorizS	1.66	1.33	1.99
TUAV					
	Transition Time (hrs)	TransT	1	0.8	1.2
	Max Speed (km/hr)	SpeedT	194	155.2	232.8
	Operating Radius (km)	RadiusT	125	100	150
	Operating Time (hrs)	OperT	6	4.8	7.2
	Time Horizon (hrs)	HorizT	7.5	6	9
Heuristics					
	Early Return	EarRet	TRUE	TRUE	FALSE
	Secondary Areas	SecArea	TRUE	TRUE	FALSE
	Appended Areas	AppArea	FALSE	TRUE	FALSE
Optimization	Optimization Interval	OptInt	0.5	0.1	0.9

Table 11. ASC-U Factors and Ranges for DOE

We developed a customized, spreadsheet-based Experimental Design Interface (EDI) for ASC-U to facilitate setting factor ranges and incorporating those values into a robust experimental design. The EDI is shown in Figure 8.

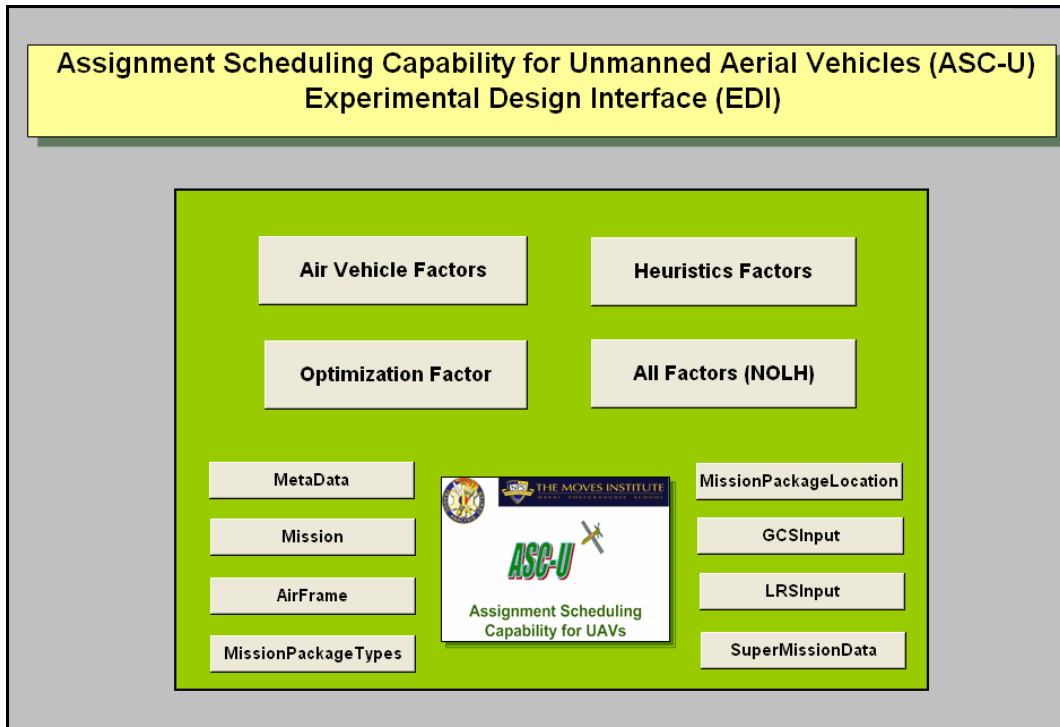


Figure 8. ASC-U Experimental Design Interface

Each button in the spreadsheet interface takes the user to a different worksheet for data entry. For example, all of the values listed in Table 11 were originally input via the “Factor” buttons within the EDI. We wrote Visual Basic for Applications (VBA) macros to allow rapid and accurate transfer of factor ranges to our experimental design worksheet (included in Appendix C). These VBA macros also allowed us to quickly and accurately enter the computed DOE values for each factor, for each design point, directly into the ASC-U input worksheets. We then created a database file for each design point from the spreadsheet input file. Each design point requires its own database that consists of eight tables, as described in Chapter III. Each of these databases provides the requisite input data for the ASC-U simulation to run.

2. Latin Hypercube Designs

After selecting the input factors, the next step in the process is to select an experimental design that will provide the responses required to fully explore the range of

input factors. To efficiently explore complex computer simulations, experimental designs possessing the following characteristics are desirable:

- Approximate orthogonality of input variables.
- Good “space-filling” properties (that is, design points are scattered throughout the region with minimal unsampled areas).
- Ability to examine many variables (20 or more) efficiently.
- Ease in generating the design.⁶²

We initially decided to use the space-filling design of Nearly Orthogonal Latin Hypercubes (NOLH) to explore the regions of the response surface. These designs give experimenters the ability to develop a thorough set of explanatory variables represented in the model.⁶³ Here, an experimenter can use a continuous range of values for each factor under investigation. Another key property of good designs is that the inner product of the columns of the design matrix is zero. That is, the columns are **orthogonal**. The NOLH design actually minimizes the correlation between the columns of the design matrix to produce a “nearly” orthogonal design (0.03 pair-wise correlation or better).

At this stage of the research effort, we found that we required 257 design points to explore the impacts of 24 factors. Since ASC-U is a deterministic simulation, this would have required us to make 257 individual runs of the simulation, with optimization intervals varying from 0.1 to 0.9. A preliminary test run revealed that a single design point with an optimization interval of 0.239 required nearly 17 hours to complete. A second test run (different design point using an optimization interval of 0.83) lasted nearly eight hours. With this information, we surmised that we had reasonable upper and lower bounds for our run time. This meant that, worst case, we could expect approximately 4,369 hours of processing time.

Fortunately, we had the opportunity to employ another Latin Hypercube design called the Optimized Flexible Latin Hypercube (OFLH). This design is the result of research done by Colonel Alejandro Hernandez, United States Army, Graduate School of

⁶² LTC Thomas M. Cioppa, United States Army, “Efficient Nearly Orthogonal and Space-Filling Experimental Designs for High-Dimensional Complex Models,” Doctoral Dissertation, Naval Postgraduate School, Monterey, CA, 2002, p. 9.

⁶³ Ibid.

Operational and Information Sciences, Naval Postgraduate School, Monterey, California.⁶⁴ By crossing 21 of the continuous ASC-U factors with two binary ASC-U factors in the OFLH design, we were able to reduce the number of design points from 257 to 96. This design produced a 0.039 pair-wise correlation between the columns of the design matrix. Overall, we realized a 62.6% decrease in computer run time from the worst case run time using the NOLH design. This improvement was certainly to our advantage, so we decided to use the OFLH design for our ASC-U data analysis (included in Appendix A).

B. MEASURES OF EFFECTIVENESS (MOEs)

ASC-U is designed to generate a number of different MOEs. These include coverage, coverage by UAV type, GCS load over time, mission assignment, mission package utilization, run information, RVT coverage, schedule, terse Remote Viewing Terminal coverage, UAV ready time over time, UAV utilization and UAV utilization over time. A more comprehensive MOE that ASC-U produces within its optimization logic is **Total Value**. This quantitative MOE is generated during the assignment of UAVs to mission areas and reflects the value obtained by a UAV as it provides coverage to a mission requirement. Total Value provides a “composite score” for GCS capacity, mission package availability, UAV type and UAV coverage time over mission areas. This number is fairly easy to calculate from the ASC-U output files and should give reasonable insight in the overall quality of model performance.

In addition to Total Value, we were also interested in how the very short optimization intervals would affect the overall quality of the solution. We expected that with each smaller interval, the Total Value would become larger since rapidly occurring optimization events should provide UAV assignment to virtually all open missions. We also were interested in how well the individual UAV types provided coverage to their assigned mission areas throughout the simulation. ASC-U provides the raw data for this MOE in the coverage by type database table.

⁶⁴ COL Alejandro Hernandez, United States Army, “Expanding the Combinatorial Space of the Nearly Orthogonal Latin Hypercube,” Doctoral Dissertation, Naval Postgraduate School, Monterey, CA, 2007, unpublished.

C. LINEAR REGRESSION MODELS

When experimenters want to examine the relationship between variables they can create an empirical model using linear regression. For example, a model that might relate the viscosity of a polymer to the temperature and catalyst feed rate is

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \varepsilon ,$$

where y represents the viscosity, x_1 represents the temperature, and x_2 represents the catalyst feed rate. This is a **multiple linear regression** model with two independent variables. We typically refer to the independent variables as predictor variables or regressors. The word “linear” is used because the model is a function of the unknown parameters β_0 , β_1 , β_2 . The model describes a plane in two-dimensional x_1 , x_2 space. The parameter β_0 defines the intercept of the plane. We can refer to β_1 and β_2 as “partial regression coefficients” because β_1 measures the expected change in y per unit change in x_1 when x_2 is held constant, and β_2 the expected change in y per unit change in x_2 when x_1 is held constant. Generally, the response variable y may be related to k regressor variables.⁶⁵ The model

$$y = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_k x_k + \varepsilon$$

is called a multiple linear regression model with k regressor variables. The parameters β_j ($j = 0, 1, \dots, k$) are called the regression coefficients. This first order model describes a hyperplane in the k -dimensional space of the regressor variables $\{x_j\}$. The parameter β_j represents the expected change in response y per unit change in x_j when all the remaining independent variables x_i ($i \neq j$) are held constant.

More complex models can also be analyzed with multiple linear regression techniques. For example, consider adding an interaction term to the first-order model in two variables, like⁶⁶

⁶⁵ Douglas C. Montgomery, *Design and Analysis of Experiments*, 6th Ed., John Wiley & Sons, 2005, p. 374.

⁶⁶ Ibid.

$$y = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \dots + \beta_{12} x_1 x_2 + \varepsilon .$$

If we allow $x_3 = x_1 x_2$ and $\beta_3 = \beta_{12}$ then the preceding model can be written as

$$y = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \varepsilon ,$$

which is a standard linear regression model with three regressors. Experimenters can also consider second-order response surface models in two variables, like

$$y = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \beta_{11} x_1^2 + \beta_{22} x_2^2 + \beta_{12} x_1 x_2 + \varepsilon .$$

If we allow $x_3 = x_1^2$, $x_4 = x_2^2$, $x_5 = x_1 x_2$, $\beta_3 = \beta_{11}$, $\beta_4 = \beta_{22}$, and $\beta_5 = \beta_{12}$, this becomes

$$y = \beta_o + \beta_1 x_1 + \beta_2 x_2 + \beta_3 x_3 + \beta_4 x_4 + \beta_5 x_5 + \varepsilon ,$$

which, again, is a standard linear regression model. Basically, any model that is linear in the parameters (the β s) is a linear regression model, regardless of the shape of the response surface it generates.⁶⁷

Experimenters make certain assumptions in order to perform meaningful regression analysis. These include:

- The relationship between the response y and the regressors is linear, at least approximately.
- The error term ε has zero mean and constant variance.
- The errors are uncorrelated.
- The errors are normally distributed.

Experimenters must consider the validity of these assumptions and conduct analysis to determine the adequacy of the regression model. Violations of these assumptions could lead to an unstable model that produces erroneous conclusions. There is no guarantee that the standard summary statistics, such as the t or F statistics or R^2 will reveal departures from the underlying assumptions.⁶⁸ These “global” model properties do not ensure model adequacy. Instead, experimenters examine the residuals of a given regression model, defined by $e_i = y_i - \hat{y}_i$, $i = 1, 2, \dots, n$, where y_i is an observation and $\hat{y}_i$

⁶⁷ Douglas C. Montgomery, *Design and Analysis of Experiments*, 6th Ed., John Wiley & Sons, 2005, p. 375.

⁶⁸ Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Peck, *Introduction to Linear Regression Analysis*, 3rd Ed., 2003, John Wiley & Sons, pp. 131-132.

is the corresponding fitted value. Plotting the residuals is a very effective way to determine how well a regression model fits the data, and to check the previously mentioned assumptions.⁶⁹

To conduct our data analysis, we selected the JMP[®] Statistical Discovery Software from the SAS Institute because of its robust statistical capabilities and overall ease of use. To begin the analysis, we generated a scatterplot, shown in Figure 9, for Total Value against each of the 96 design points. We selected a few of the design points that showed higher Total Value. We were interested in whether or not these particular points shared some commonality that caused them to have the higher Total Values. In Section D, we discuss the regression analysis that further investigates these preliminary observations.

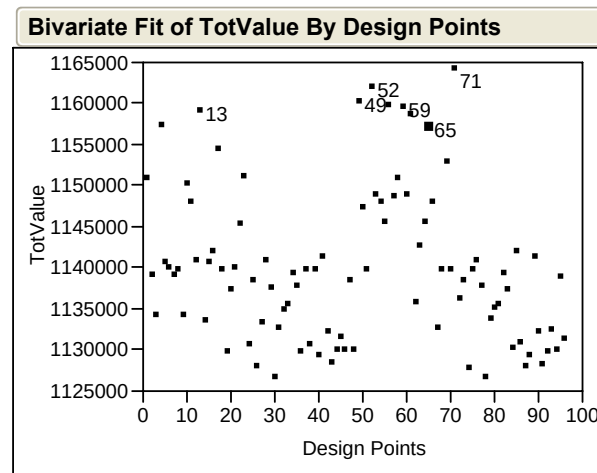


Figure 9. Scatterplot of Total Value by Design Point

D. TOTAL VALUE MODEL

This section provides analysis of a multiple regression model using Total Value as the response variable. Consistent with our research objectives, we wanted to understand which ASC-U factors have the greatest influence on the Total Value MOE.

⁶⁹ Douglas C. Montgomery, Elizabeth A. Peck, and G. Geoffrey Peck, *Introduction to Linear Regression Analysis*, 3rd Ed., 2003, John Wiley & Sons, pp. 131-132.

1. Main Effects Model

We began the analysis by considering the main effects regression model, consisting of 23 regressor variables. In order to establish a baseline, we used a Standard Least Squares regression in JMP® with a probability to enter and leave of 0.05 and 0.10, respectively. Figure 10 displays the actual value versus predicted value plot for the regression.

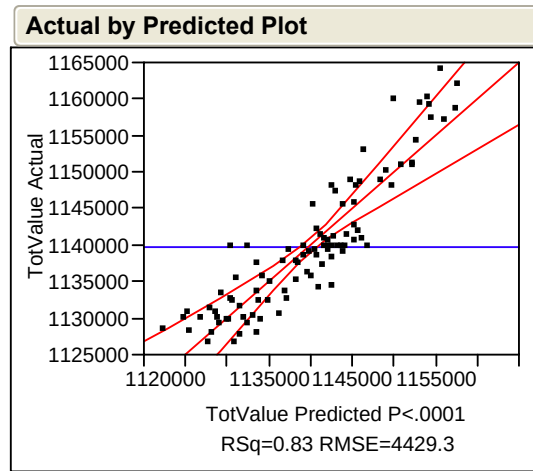


Figure 10. Actual vs. Predicted Total Value for Main Effects

Four factors were statistically significant with a $\text{Prob}>|t| < .0001$. Six other factors were significant at 0.0025 or less. All totaled, 10 of the 23 factors were significant in this model. For those data points that fell outside the confidence interval, we did not observe any noteworthy circumstances.

The R-square value provides an indication of what proportion of the variability in the response variable the regression model explains. It should be used with caution because the value is inflated when more regressors are added to the model. When this occurs, although a higher R-square value results, the model is not necessarily improved. Experimenters can use another statistic called the Adjusted R-square

($R^2_{Adj} = 1 - \frac{SS_{Res}/(n-p)}{SS_T/(n-1)}$), where n is the number of data points under observation and p

is the number of estimated parameters. The Adjusted R-square actually introduces a

penalty for what is called **over-fitting** a regression model (or adding unnecessary terms), and will only increase when the addition of a regressor reduces the residual mean square. The main effects model produced an R-square value of 0.8264, an Adjusted R-square of 0.7710, and a residual mean square of 19,619,072. We will compare these values to the First Order Model with Interactions that we evaluate next. Since we are adding many more terms, we need to look for a reduction in the residual mean square to determine if the new model is indeed superior.⁷⁰

2. First Order Model with Interactions

In order to further investigate trends in our data and identify other significant factors, we created a first order model with interactions. We began with all of the main effects and two-way interactions as potential regressors, using the “factorial to degree 2” macro in the “Fit Model” utility of JMP®. A mixed stepwise regression approach is used to identify statistically significant terms, followed by a Standard Least Squares regression as described for the main effects model.

Figure 11 shows the actual value versus predicted graph for the first order model with interactions. Here again, we observe that the predicted plot and the actual data appear fairly stable. The stepwise process selected 11 main effects and 10 interactions. The model produced an R-square of 0.9292. Eleven of the terms were significant with a p -value $< .0001$. The remaining ten terms were significant with p -values < 0.0485 . As expected, the addition of many more terms led to a higher R-square value than that of the Main Effects model. More importantly, the model produced a residual mean square of 7,791,050, which is a 60.3% decrease from the Main Effects Model. This indicates that we certainly have created a model with better predictive capability. It is interesting to note that this model selected the Operating Time for each UAV (OperE, OperT, Oper4), except the SUAV (OperS). The only significant factor for the SUAV was its Operating Radius (RadiusS). Operating Radius was also selected for the only FCS UAV in our model, the Class IV UAV. The model selected Time Horizon for both the ERMP and the TUAV. The Transition Time for the TUAV was also selected.

⁷⁰ Douglas C. Montgomery, Elizabeth A. Peck, and Geoffrey Peck, *Introduction to Linear Regression Analysis*, 3rd Edition, 2003, John Wiley & Sons, pp. 90-91.

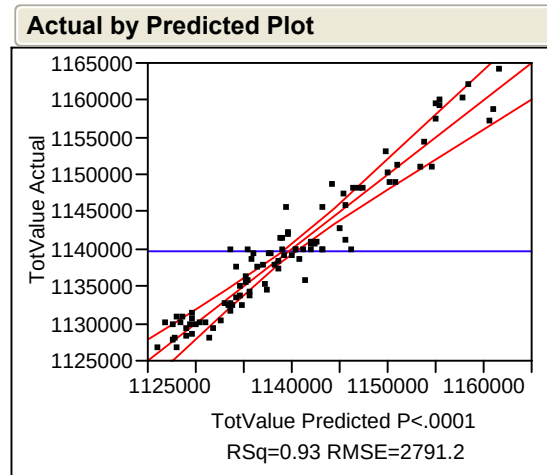


Figure 11. Actual vs. Predicted Total Value for First Order with Interactions

3. Full Quadratic Model

For this regression, we brought our previous models together to create a full quadratic model allowing us to include main effects, interaction terms and polynomial terms. We used the same process as previously described for the main effects and first order model with interactions. JMP® provides analysts with the capability to add polynomial terms by way of a user-defined setting for the desired degree (of the polynomial terms being added). At this point, we chose polynomial to degree “2” for our quadratic terms. We decided to push a littler harder to keep insignificant terms out of the model (since we were now looking at a very large number of terms). To accomplish this, we considered changing the probability to leave from 0.10 to 0.05 in order to cut off insignificant terms.

Figure 12 shows the actual value versus predicted graph for the full quadratic model. Interestingly, the full quadratic model produced the exact same output as the first order model with interactions. This tells us that when the stepwise regression chose significant terms, none of them were quadratic.

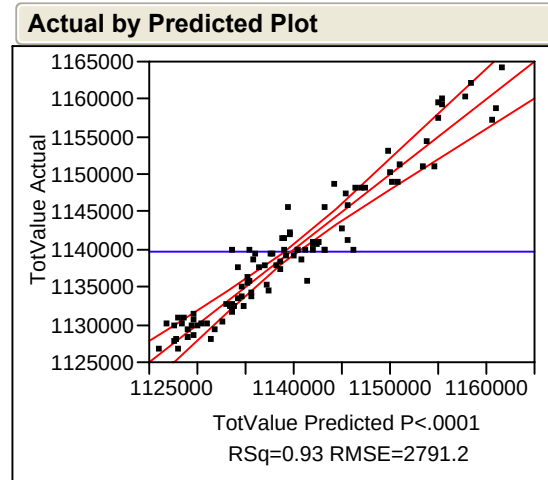


Figure 12. Actual vs. Predicted Total Value for Full Quadratic Model

4. Final Total Value Model

To build the final model for the Total Value response variable, we examined the stepwise history for the full quadratic model. This model consists of 10 interaction terms shown in Table 12, as well as their associated main effects. JMP[®] provides the R-square value as each factor is added.

Step History							
Step	Parameter	Action	"Sig Prob"	Seq SS	RSquare	Cp	p
1	(HorizE-45)*(EarRet-0.5)	Entered	0.0000	4.354e+9	0.5351	323.6	4
2	(OperE-36)*(HorizT-7.5)	Entered	0.0000	1.273e+9	0.6914	191.17	7
3	(Radius4-75)*(RadiusS-16)	Entered	0.0000	7.469e+8	0.7832	115.92	10
4	(EarRet-0.5)*(SecAreas-0.5)	Entered	0.0000	4.983e+8	0.8444	65.718	12
5	(OperE-36)*(EarRet-0.5)	Entered	0.0001	2.054e+8	0.8697	45.375	13
6	(TransT-1)*(OperT-6)	Entered	0.0001	2.385e+8	0.8990	25.434	16
7	(OperE-36)*(Oper4-6)	Entered	0.0036	1.102e+8	0.9125	17.441	18
8	(Oper4-6)*(SecAreas-0.5)	Entered	0.0236	46128036	0.9182	14.423	19
9	(OptInt-0.5)*(SecAreas-0.5)	Entered	0.0369	56069301	0.9251	12.324	21
10	(Oper4-6)*(EarRet-0.5)	Entered	0.0426	33149782	0.9292	10.718	22

Table 12. Significant Interactions and R-Square Value for Total Value Model

The statistical output for the Final Total Value Model is shown in Figure 13.

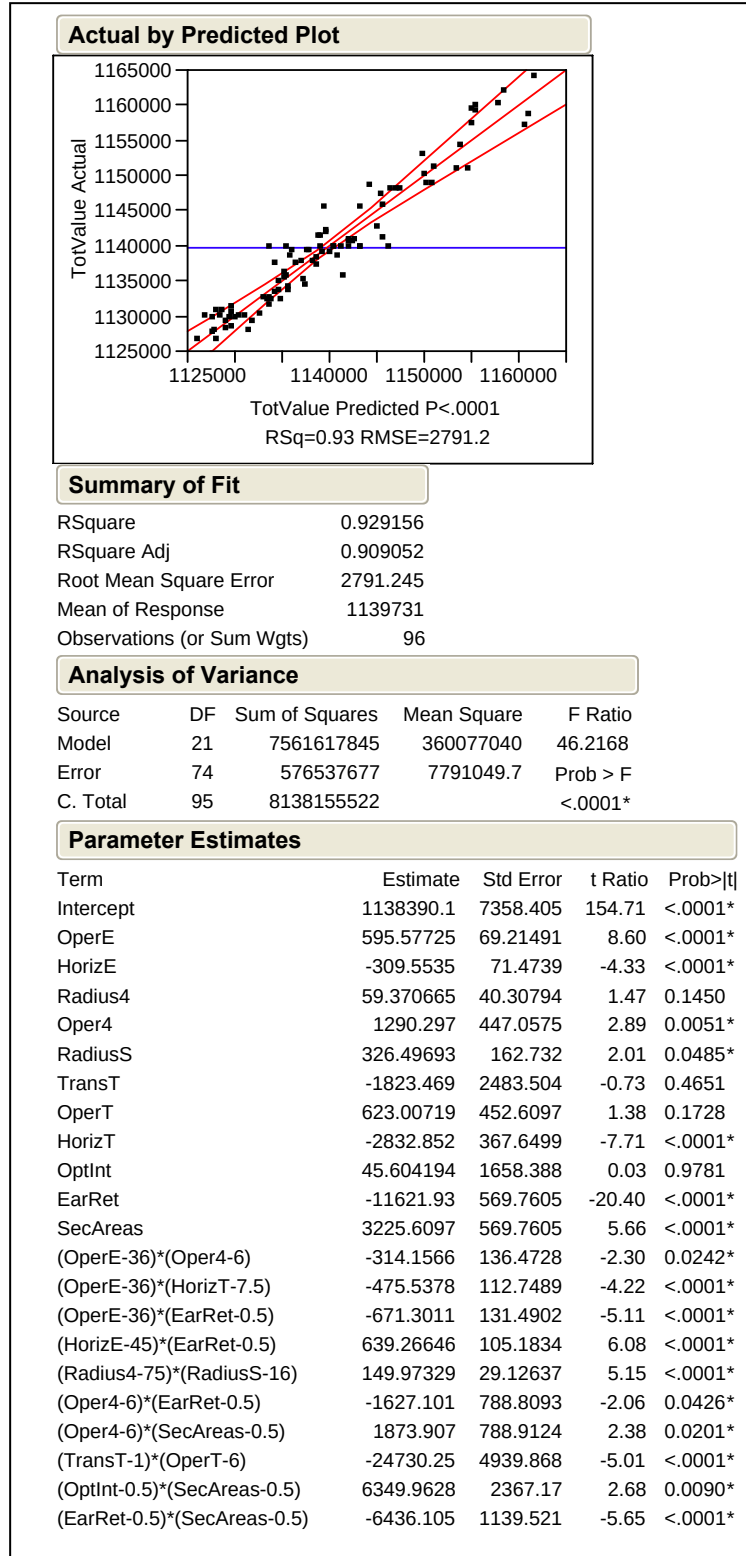


Figure 13. Statistical Output for Final Total Value Model

As we mentioned earlier in this section, experimenters make certain assumptions when performing regression analysis. If these assumptions are violated, we cannot hope to gain meaningful insight from the analysis. The first assumption that we must verify is that there exists an approximately linear relationship between the regressors and the response variable. This is best determined by examining a plot of Actual Value by Predicted Value, which is shown in Figure 14.

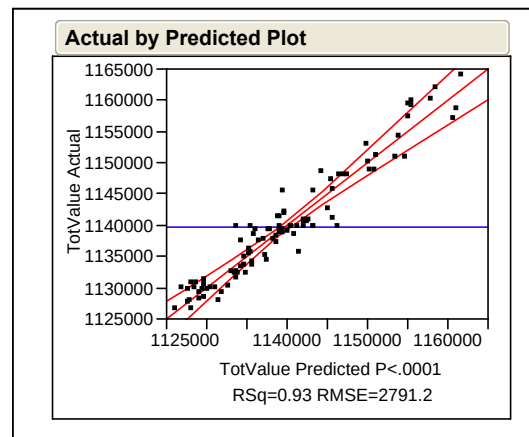


Figure 14. Actual by Predicted Plot for Final Total Value Model

Figure 14 confirms that there is an approximately linear relationship between the regressors and the response variable. Next, we must verify that the residuals have zero mean and constant variance. Here, we turn to a plot of the Total Value residuals to check this assumption. Figure 15 provides the plot we need to examine.

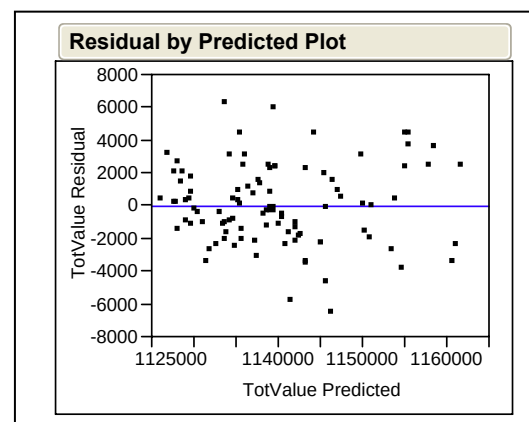


Figure 15. Residual by Predicted Plot for Total Value Residuals

Since the residuals in Figure 15 are randomly scattered around a mean of zero, and we do not observe any discernable pattern, we conclude that the assumption is satisfied. Finally, a normal probability plot can be used to check the normality assumption. The plot is designed such that the cumulative normal distribution will graph as a straight line. Let $e_{[1]} < e_{[2]} < \dots < e_{[n]}$ be the residuals ranked in ascending order. By plotting $e_{[i]}$ against the cumulative probability $P_i = (i - \frac{1}{2})/n$, $i = 1, 2, \dots, n$, on the normal plot, the resulting points should fall on a straight line,⁷¹ with emphasis falling on the middle values. The normal plot, histogram (overlaid with a normal curve), and an outlier box plot for the Total Value residuals are all shown in Figure 15. Additionally, JMP® reports the *Shapiro-Wilk W Test* value of 0.9923 at a significance level of 0.8585. Thus, the residuals satisfy the normality criteria.

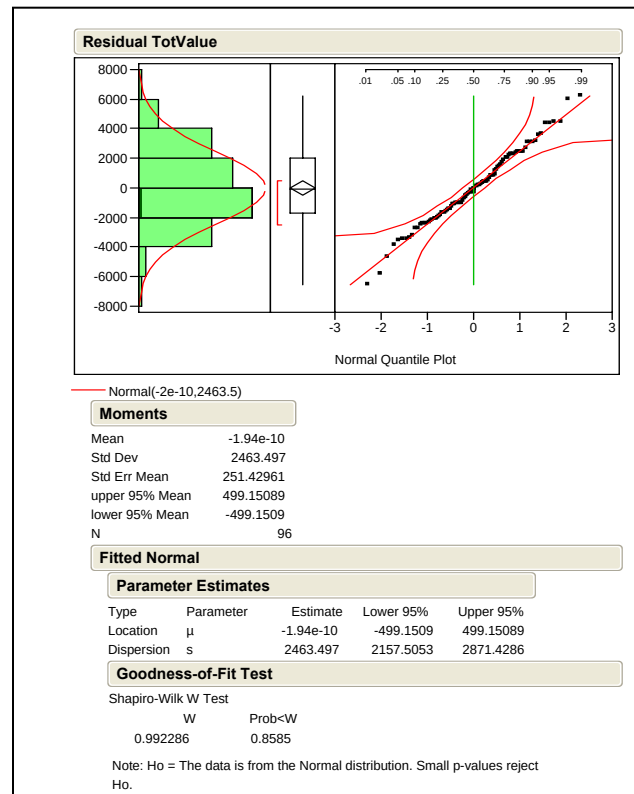


Figure 16. Histogram, Outlier Box Plot, and Normal Plot for Total Value Residuals

⁷¹ Douglas C. Montgomery, Elizabeth A. Peck, and Geoffrey Peck, *Introduction to Linear Regression*

Another factor that requires further investigation is the ASC-U Optimization Interval (Recall that ASC-U has two parameters within the simulation that it uses for Optimization events, the Optimization Interval and the Time Horizon. Because of the way ASC-U builds optimal UAV schedules by way of an assignment process, we predicted that if we gave that model frequent opportunities to perform assignments, we would see an increase in the Total Value with each smaller interval. We chose to set the range on the Optimization Interval at 0.1 to 0.9. After collecting the results of the model runs, the results are different than what we expected. The data show that even when multiple ASC-U runs have incrementally smaller optimization intervals, the Total Value generated by the model does not steadily increase with these smaller and smaller Optimization Intervals. We expect that most of this oscillation is explained by the settings on the binary heuristics, which allow UAVs to follow different rule sets after they complete their primary mission.

We determined that it would be beneficial to take an aggregate view of the results by design point. In our crossed design, the settings for the factors are repeated every 24 points, differing only by the settings on the binary heuristics. This creates four different Total Value results for each Optimization Interval across the entire design. We decided to group the repeated intervals together and compute the mean Total Value for each group. The graph in Figure 17 plots the mean Total Value against the grouped Optimization Intervals that we used in this thesis.

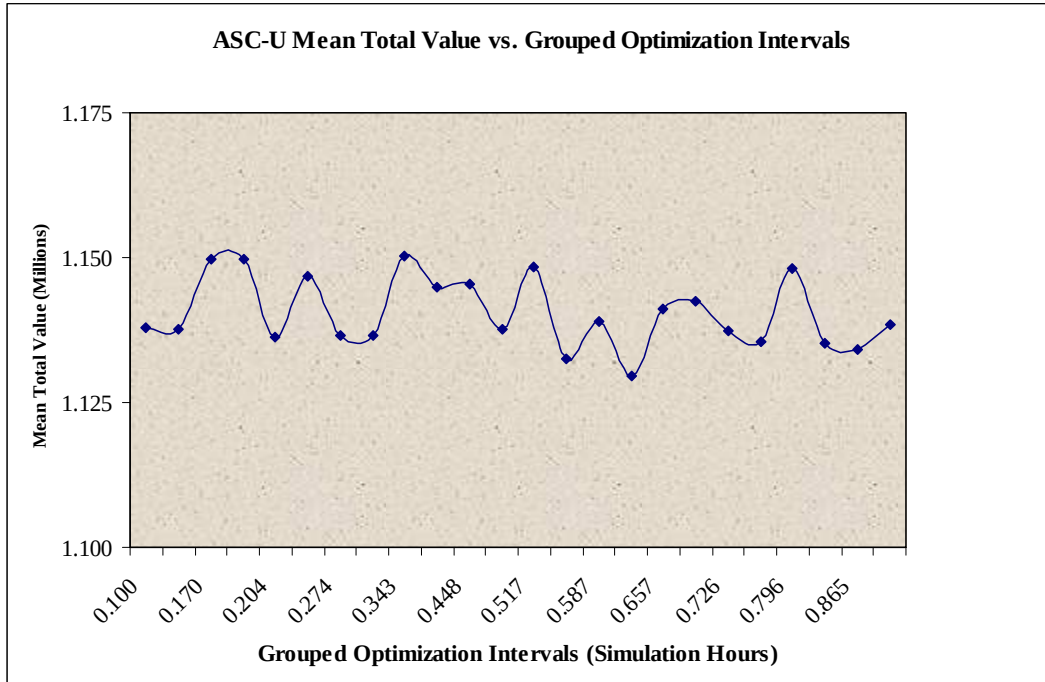


Figure 17. ASC-U Mean Total Value vs. Grouped Optimization Intervals

What is important to note in Figure 17 is that there is only a 1.74% difference in the lowest and the highest mean Total Value. These results tell us that the simulation is very consistent across the range of the input factors used in this experimental design.

5. Block Analysis

At this point in the analysis, we wondered what the data would show if we broke the design into blocks by heuristic settings. Recall that the OFLH design (shown in Appendix A), was crossed with the binary heuristics Early Return and Secondary Areas. We chose to set the third binary heuristic, Appended Areas, to the opposite value of Early Return (mutually exclusive factors) in every case. Therefore, it was not necessary to include Appended Areas in the design (yielding 23 factors instead of 24). For clarity, the definitions are:

- **Early Return** (Go home versus stay on station) – When the UAV completes its assigned mission, a value of TRUE allows the UAV to fly

home, if possible, and return in time for the next mission. A value of FALSE forces the UAV to stay on station until the end of its operating time.

- **Secondary Areas** (Go obtain more value from another mission) – When satisfying two missions located at the same point (separated by some amount of time), a value of TRUE allows the UAV to complete another mission between the two missions already assigned. A value of FALSE forces the UAV to stay on station and wait for the start time of the second mission.
- **Appended Areas** (Primary mission done; other missions available?) – When the UAV completes the current mission, a value of TRUE allows the UAV to locate another mission within its remaining Operating Time. A value of FALSE prohibits the UAV from locating another mission to satisfy.

We decided to create four different blocks from the OFLH design to correspond with the different employment tactics caused by the TRUE/FALSE heuristic settings. Table 13 shows the setup for the block analysis.

Block	Design Points	EarlyReturn	SecondaryAreas	AppendedAreas
1	1 - 24	FALSE	FALSE	TRUE
2	25 - 48	TRUE	FALSE	FALSE
3	49 - 72	FALSE	TRUE	TRUE
4	73 - 96	TRUE	TRUE	FALSE

Table 13. Designation of OFLH Blocks by Heuristic Value

The goal for the block analysis was to determine whether or not any of the particular heuristic combinations (Block 1, 2, 3, or 4) produced a higher mean Total Value than the complete design (all 96 design points together). First, we created four separate subset tables in JMP® and generated histograms and a smooth curve fit for each the block. The results are shown in Figure 18.

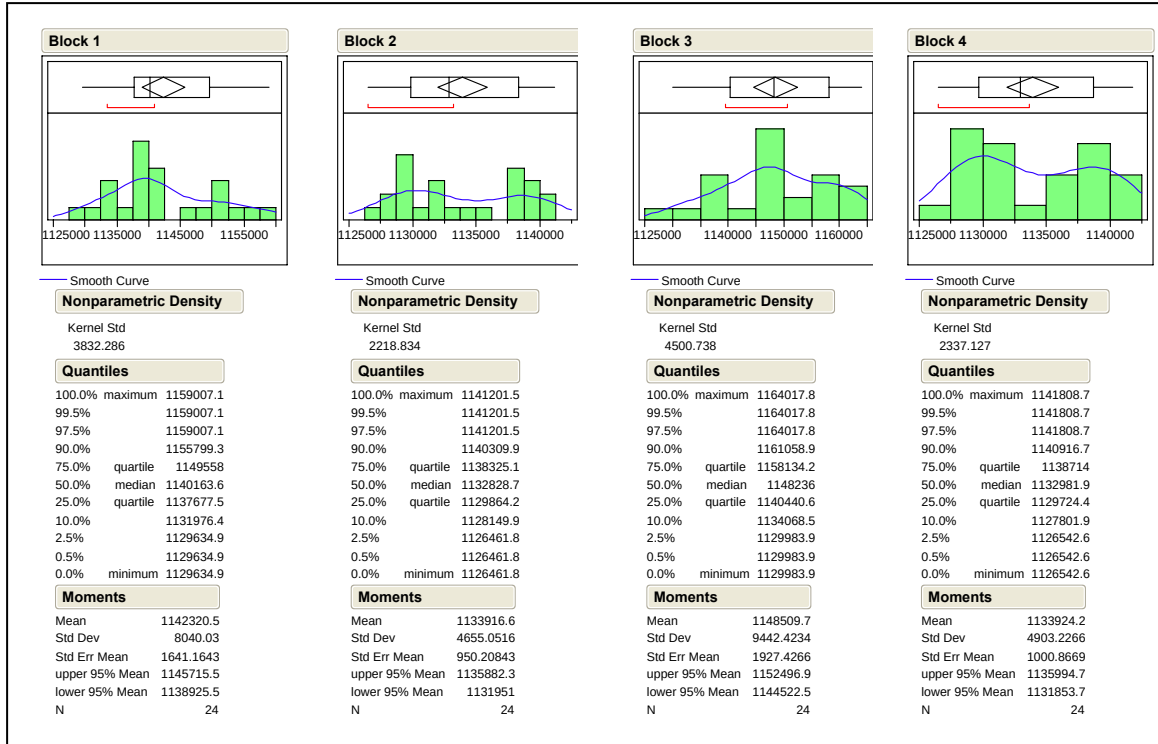


Figure 18. Histograms and Smooth Curve Fit for OFLH Block Analysis

Interestingly, the Block 3 mean Total Value is actually higher than the mean value of the overall Total Value model. This leads us to believe that the optimal UAV employment strategy to maximize total value is to set Early Return to FALSE, Secondary Areas to TRUE, and Appended Areas to TRUE. This optimal strategy is reflected in Table 14.

Block	Design Points	EarlyReturn	SecondaryAreas	AppendedAreas
3	49-72	FALSE	TRUE	TRUE

Table 14. Components of OFLH Block with Highest Mean Total Value

This result is consistent with the output from the Regression Tree produced earlier in this chapter. There, after the first split, the analysis indicated that we should keep the EARLY RETURN heuristic < 0 , or set to FALSE, in order to see an increase in the mean

Total Value. There appears to be consistent evidence from the data analysis that this particular employment strategy will yield the best results when the goal is to maximize Total Value.

6. Significant Factors and Interpretation of Regression Coefficients

After completing the Total Value regression runs, we observed several significant factors in the model. Additionally, we expected that each of these factors would make a positive contribution to the response variable (yielding positive regression coefficients $\hat{\beta}_j$). To check our predictions, we compared them to the actual signs on the regression coefficients from the model output. Table 15 contains the results.

Significant Factor	Predicted Sign on Regression Coefficient	$\hat{\beta}_j$	Prob > t
OperE	+	595.6	<.0001
HorizE*	+	-309.6	<.0001
Oper4	+	1,290.3	<.0015
RadiusS	+	326.5	<.0485
HorizT*	+	-2,832.9	<.0001
EarRet*	+	-1,1621.9	<.0001
SecAreas	+	3,225.6	<.0001
*indicates a difference between Predicted and Actual sign on Regression Coefficient			

Table 15. Significant Factors and Signs on Regression Coefficients for Final Total Value Model

We observed that three of the significant factors (HorizE, HorizT, and EarRet) have signs on the regression coefficients that appear counterintuitive. We will explain this finding in more detail later in this section.

Recall from Chapter III that the Time Horizon ranges from 1.0 to 1.5 times the Operating Time plus Transition Time for each UAV in the scenario. This value is essential in determining which missions to consider in the look-ahead time window to launch a UAV. Of these seven significant factors, three are UAV performance characteristics (OperE, Oper4, and RadiusS), and the other four are model settings (HorizE, HorizT, EarRet, and SecAreas). We wanted to determine which factors

(performance characteristics and model settings) had the greatest impact on Total Value. To accomplish this, we used another tool within JMP® called a Prediction Profiler, shown in Figure 19.

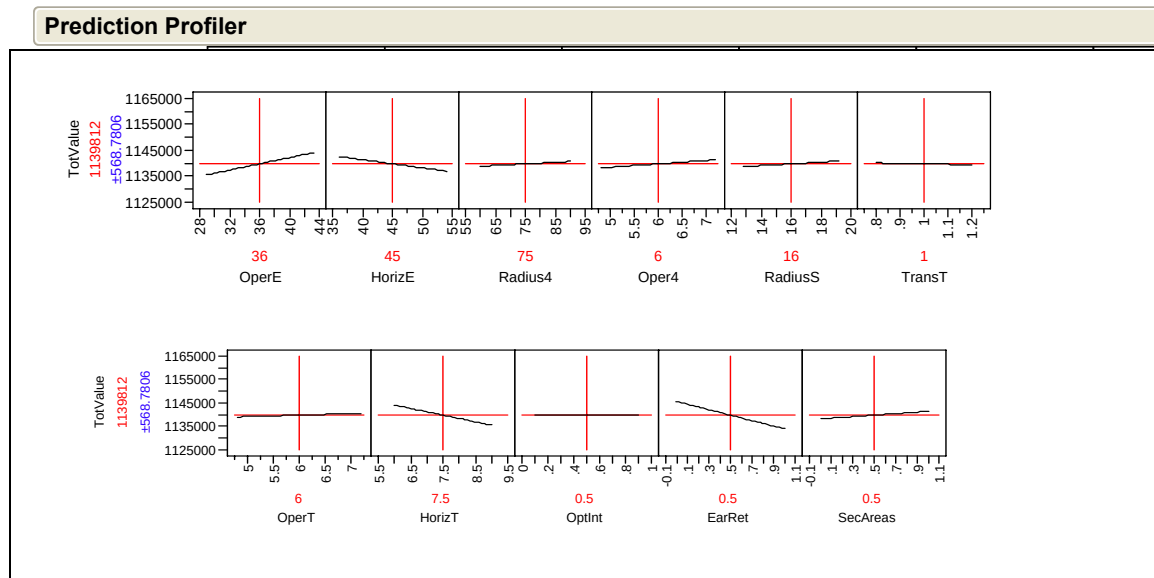


Figure 19. Prediction Profiler for Final Total Value Model

The Prediction Profiler indicates that the Operating Time for the ERMP improves Total Value as the factor (OperE) increases across its range. It appears that the more opportunity the ERMP has to take advantage of its extended range, the higher value it generates from increased mission coverage. As the ERMP Time Horizon (HorizE) increases across its range, Total Value decreases. This tells us that there is a threshold for the ERMP Time Horizon that we should observe. We set the range on the ERMP Time Horizon at 36 to 54 hours. We suspect that at some value in this range, the simulation will attempt to allocate UAVs to missions that have a higher value in the future, thus sacrificing opportunities to gain value from missions that occur sooner in the look-ahead window. Likewise, as Time Horizon values for the Tactical UAV (HorizT) increase across their range, Total Value shows a noticeable drop. Finally, we see a decrease in Total Value when the EarRet heuristic is set to TRUE (or “on”). This could result from the simulation forcing UAVs home that have the endurance to fill other missions during a gap in coverage. At this point we observe a direct relationship between Total Value and

four of the significant factors (OperE, Oper4, RadiusS, and SecAreas). On the contrary, we observe an inverse relationship between Total Value and the three remaining significant factors (HorizE, HorizT, and EarRet). All other factors appear to have a relatively flat slope or slightly positive slope and indicate a small or mild effect on the response variable. Additionally, the factors in the Prediction Profiler do not exhibit any curvature, because no quadratic terms are included in the model.

We also used a non-parametric tool in JMP[®] called regression trees to shed more light on significant factors. Regression trees provide a nonparametric tool (no distributional assumptions are made) that experimenters can use to analyze data. The algorithm starts with all the data in a single group, and then looks recursively over all potential quantitative and qualitative explanatory variables to locate the best split.⁷² Figure 20 shows the regression tree after four splits.

⁷² Susan M. Sanchez, *OA 4333 Simulation Analysis*, Naval Postgraduate School, Spring 2007, In-class handout, “JMP in a nutshell...,” p. 2.

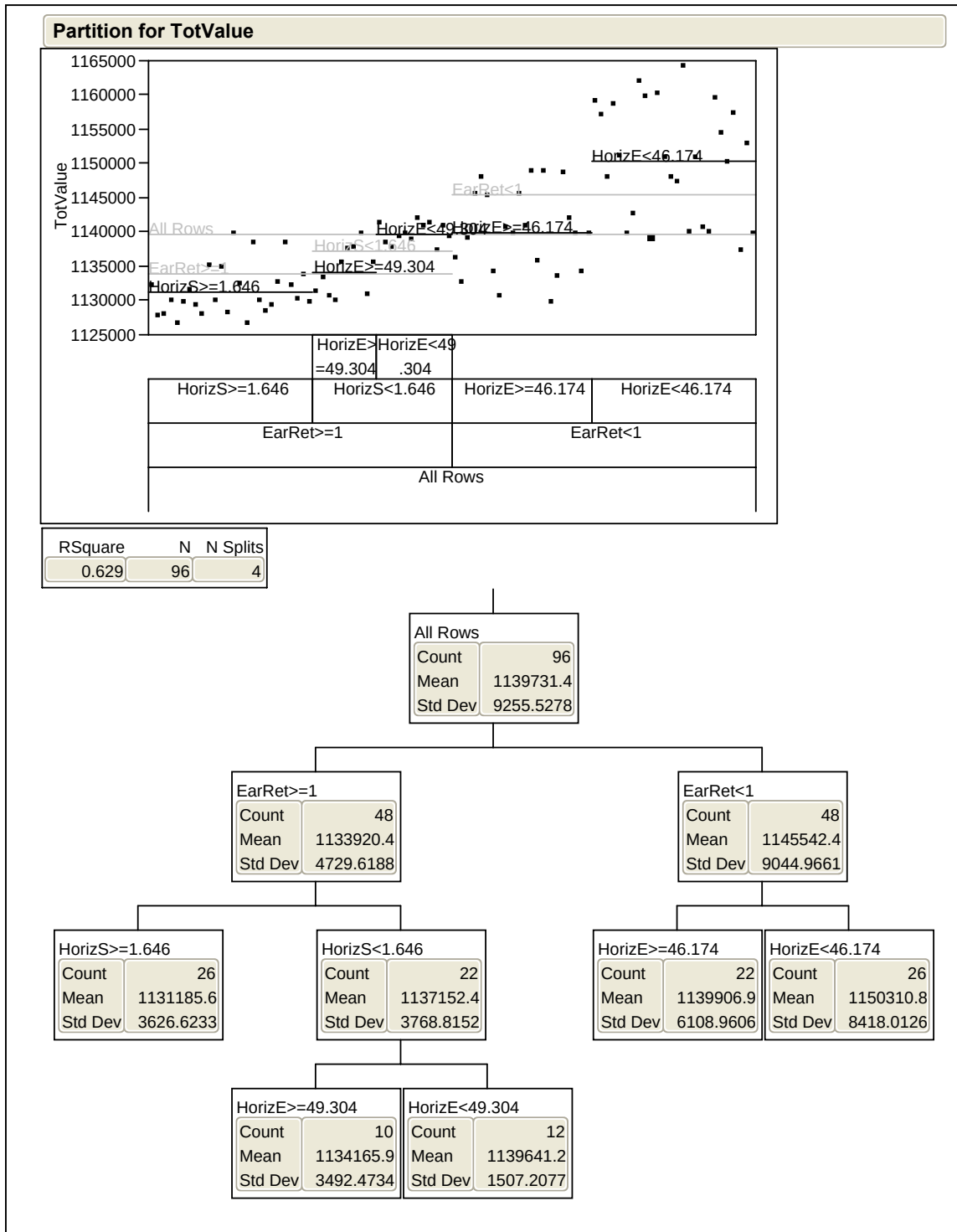


Figure 20. Total Value Regression Tree After Four Splits

In the first split, the tree indicates that when the heuristic EarRet is set to FALSE (or “off”), Total Value improves. In this particular case, we see an increase of 0.51% in mean Total Value. The ERMP Time Horizon (HorizE) generates the next split that provides an improvement in Total Value. As we expected from our observation of HorizE in the Prediction Profiler, there is a threshold at which the simulation appears to assign UAVs to higher value missions in the future (while losing value for missions closer in). According to the regression tree, this value is 46.174 simulation hours (in this scenario). Recall that the upper bound on the factor setting in our design for HorizE is 54. Thus, when the HorizE factor settings exceed 46.175, we actually lost Total Value with the ERMP. Taken together, turning off EarRet and setting HorizE to no more than 46.174 we realize a 1% improvement in mean Total Value. Ultimately, we find that the three significant UAV performance characteristics (OperE, Oper4, and RadiusS) and one of the significant model settings (SecAreas) have positive effects on Total Value, while the remaining significant model settings (HorizE, HorizT, and EarRet), have negative impacts.

As mentioned earlier in this section, we wanted to investigate the counterintuitive signs on the regression coefficients for the significant factors in the Final Total Value Model. Regression coefficients may have the wrong sign for the following reasons:

- The range of some of the regressors is too small,
- Multicollinearity is present,
- Computational errors have been made, or
- Important regressors have not been included in the model.⁷³

Within this particular experiment, we exercised complete control over the ranges of the factors we selected. After consultation with subject matter experts, we determined that a +/- 20% adjustment to each base factor setting would provide reasonable upper and lower bounds for our factors. In a scenario where an experimenter does not have control over these ranges, counterintuitive signs on regression coefficients could result. We also knew that we could eliminate the possibility of multicollinearity because our Optimized

⁷³ Douglas C. Montgomery, Elizabeth A. Peck, and Geoffrey Peck, *Introduction to Linear Regression Analysis*, 3rd Edition, John Wiley & Sons, 2003, pp. 120-121.

Flexible Latin Hypercube design (with a pairwise correlation of 0.039) is specifically engineered to prevent linear dependence among the columns of the design matrix. We acknowledged that computational errors are possible, but unlikely, with a robust, commercial statistical software package like JMP®. This left us to consider whether we may have left some important regressors out of the model. In this case, the signs on the regression coefficients are not necessarily wrong. The partial nature of the regression coefficients can cause sign reversal to occur. For example, $\hat{\beta}_1$ is a “partial” regression coefficient for the multiple regression model $y = \beta_0 + \beta_1 x_1 + \beta_2 x_2$ which measures the effect of x_1 , given that x_2 is also in the model.⁷⁴

7. Significant Interactions

Although the Optimization Interval for the final Total Value Model has p-value of 0.9781, there is an interaction between the Optimization Interval and the binary heuristic Secondary Areas with a p-value of 0.0090. While it certainly appears that the data supports the notion that the Optimization Interval was not significant in this analysis, we cannot ignore this interaction. We used the interaction profile in JMP® to show the interaction. In the plot in Figure 21, the y-axis represents total value. Solid lines indicate an interaction between the terms. The x-axis represents the range on the term. Changes in the terms’ value affect the Total Value according to the slope of the lines found in the plots. The plots across each row display two lines, one each for the high and low values of the term listed on the diagonal. We can determine changes to the response variable when the row term is set to its high or low value and we vary the range of a term found in the column to the left or right of the diagonal, holding other effects constant.

⁷⁴ Douglas C. Montgomery, Elizabeth A. Peck, and Geoffrey Peck, *Introduction to Linear Regression Analysis*, 3rd Edition, John Wiley & Sons, 2003, pp. 120-121.

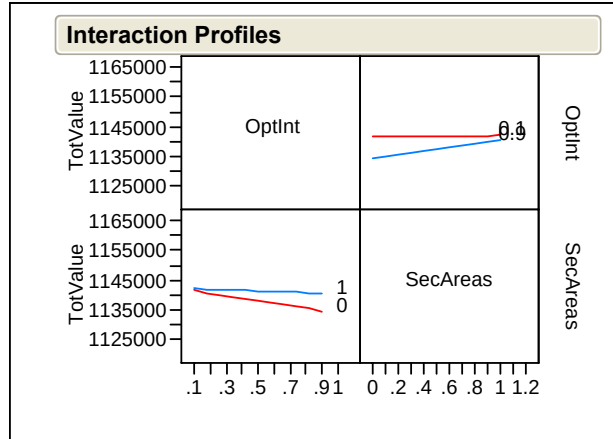


Figure 21. Interaction Plot for Optimization Interval and Secondary Areas

Looking at the top row, we see the Optimization Interval crossed with Secondary Areas. The top line represents the change in Total Value when the Optimization Interval is set to 0.1. The fact that this line is practically flat (with a slightly positive slope) means that there is a minimal increase in Total Value when Secondary Areas goes from FALSE to TRUE. When the Optimization Interval is set to its highest level of 0.9, Total Value improves when Secondary Areas goes from FALSE to TRUE. Likewise, the bottom row shows that when Secondary Areas is set to its highest value of 1 (or TRUE) the line is relatively flat, with a slight negative slope as the Optimization Interval increased across its range. When Secondary Areas is set to its low value of zero (or FALSE), Total Value decreases as the Optimization Interval increases across its range.

This reveals that when the Optimization Interval is set to a very low value, the impact of Secondary Areas being set to TRUE is minimal. This is because the extremely frequent optimization events (at an Optimization Interval of 0.1, the model reoptimizes every 1/10 of a simulation hour, or every six simulation minutes) will not allow the model to miss opportunities to gain value from UAV coverage. Also of note are the long run times for this short of an Optimization Interval. Run times exceeded 48 hours for the 0.1 interval. Another fact that must be considered with frequently occurring optimization events is the technique required to solve an integer programming problem. Some integer programs can have a massive number of variables. With a commercial mathematical

programming solver like CPLEX, experimenters have the advantage of an industrial strength presolve routine that essentially shrinks a problem down to a manageable size before solution occurs. Although `lp_solve` has presolve options, implementation does not automatically reduce simulation run time. When `lp_solve` has to use the technique of branch-and-bound to solve a massive integer programming problem, simulation runs may require a significant amount of time. Another viable solution for the optimization logic is a formulation that takes advantage of the underlying network structure within the simulation. Using a multicommodity network flow formulation (MCNF) may completely remove the need for the optimization logic the churn through branch-and-bound. Although most MCNF problems do not guarantee a unimodular solution, a careful formulation can yield integer solutions. For programmatic purposes, this long run time may be acceptable. In an operational setting, rapid solution times are desirable and must be pursued if ASC-U ever hopes to transition to use in a real-world contingency environment.

One of our goals for this thesis was to determine guidelines for selecting the Optimization Interval (Interval) – specifically, to see whether there is an optimal value for the Interval below one simulation hour. To accomplish this objective, we compared the design points with Intervals of 0.9 and 0.1 in each block of our design to new Intervals of 10 and 1, respectively. We also wanted to investigate whether the long run times for Intervals below one simulation hour provided substantial benefit. Table 16 shows that the Total Value changes less than 1% when the Interval moves from 0.9 to 0.1; however, it improves up to 36% when set from 10 to 1. This data reveals that the relative change in Total Value from 0.9 to 0.1 does not provide a substantive improvement in the solution. Although the Total Value quantities for Intervals below one simulation hour differ slightly, there is no optimal value for the Interval of practical significance. Given the minimal change in Total Value, we did not observe a substantial benefit from the extended run times.

Design Point	Original Interval	Total Value	Run Time	Design Point*	New Interval	Total Value	Run Time
9	0.9	1134187.609	8 hrs	9*	10	896642.171	1 hr
18	0.1	1142082.586	48 hrs	18*	1	1135316.524	5 hrs
		+ 0.696%				+ 26.619%	
33	0.9	1135448.976	8 hrs	33*	10	823755.926	1 hr
42	0.1	1132118.782	48 hrs	43*	1	1123318.671	5 hrs
		- 0.293%				+ 36.365%	
57	0.9	1148541.073	8 hrs	57*	10	931465.658	1 hr
66	0.1	1147930.960	48 hrs	66*	1	1147697.031	5 hrs
		- .053%				+ 23.214%	
81	0.9	1135519.106	8 hrs	81*	10	823605.850	1 hr
90	0.1	1132182.904	48 hrs	90*	1	1123181.150	5 hrs
		+ 0.294%				+ 36.374%	
+ or - sign indicates percent increase or decrease respectively for Total Value							

Table 16. Percent Change Comparison for ASC-U Optimization Intervals

E. SENSITIVITY ANALYSIS

In Chapter III, we stated that ASC-U is a deterministic simulation. This deterministic nature provides the same output for each run if the input parameters are left unchanged. Since ASC-U originally developed as a programmatic tool to help provide analysis to support different U.S. Army UAV investment strategies, ASC-U's deterministic roots are understandable. We considered the capability that ASC-U has to create executable UAV flight schedules that provide coverage over designated Mission Areas, and asked ourselves a different question: how might the UAV coverage for identified Mission Areas change in the presence of adverse random events such as sensor failure, maintenance problems, or enemy fire? The question naturally leads us out of the planning, budgeting, and programming world into an operational setting. Here, military commanders might be able to take advantage of the UAV scheduling prowess of ASC-U and integrate its flight schedules with other airborne assets in a particular objective area. We surmised that UAV coverage over designated Mission Areas would surely degrade in the presence of adverse random events. This is a logical conclusion that does not require advance modeling and simulation techniques. As we thought more about the potential extension for ASC-U into an operational setting, we asked ourselves another question: in the presence of adverse random events, can we **quantify** the

potential degradation in UAV coverage? This is a more interesting question, so we set out to construct a model to help us determine the answer.

We decided to construct a stochastic spreadsheet model to help us investigate our research objective. We considered two different sampling techniques for our stochastic model: Monte Carlo Sampling and Latin Hypercube Sampling. The following provides some insight into these two techniques.

The Cumulative Distribution Function $F(x)$ gives the probability P that the random variable X is less than or equal to x , i.e.,⁷⁵

$$F(x) = P(X \leq x).$$

$F(x)$ ranges from 0 to 1. This equation can be examined in the reverse direction. Then, the question becomes: what is the value of $F(x)$ for a given value of x ? The name for this “reverse look” is the **inverse function** and is written as:

$$G(F(x)) = x.$$

This concept of the inverse function is central to the generation of random samples from each distribution in a stochastic model. Random samples are generated from a probability distribution using a pseudo-random number r , whose value lies between 0 and 1. This number is passed to the following equation to establish the value to be generated:

$$G(r) = x.$$

Typically, a Uniform(0, 1) distribution is used to generate the number r to provide equal chances of an x value occurring in any percentile range. The inverse function technique is used in several sampling methods. Figure 22 shows a graphical relationship between $F(x)$ and $G(F(x))$.⁷⁶

⁷⁵ David Vose, *Risk Analysis – A Quantitative Guide*, 2nd Edition, John Wiley & Sons, 2000, p. 57.

⁷⁶ Ibid., pp. 57-58.

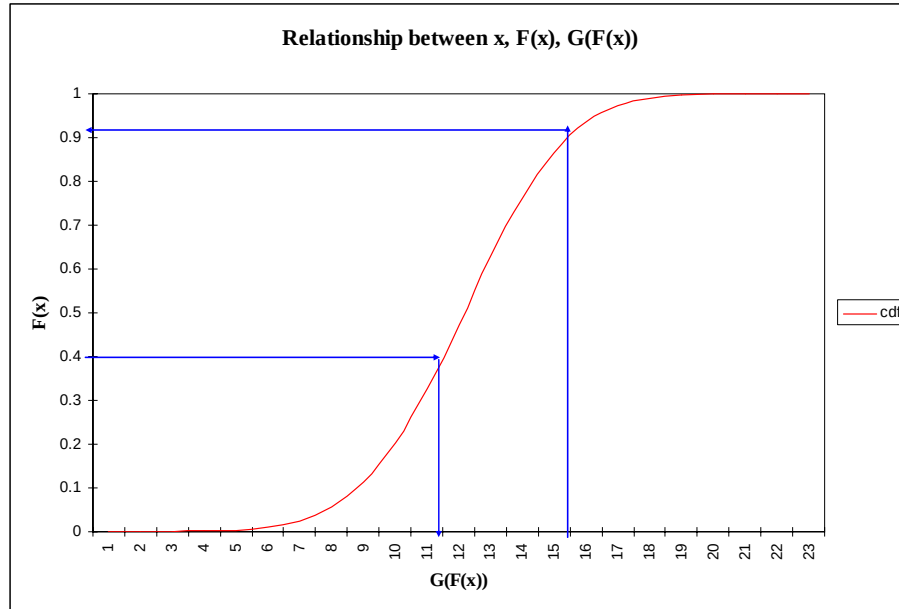


Figure 22. Relationship between x , $F(x)$, $G(F(x))$ (after)⁷⁷

Monte Carlo sampling employs the previously described sampling method exactly. The drawback is that the randomness of its sampling method translates into over and under sampling from a distribution and does not replicate the shape of the input distribution, unless one performs a very large number of iterations.⁷⁸

Latin Hypercube Sampling, on the other hand, addresses this issue by using a method of sampling that appears to be random, but also guarantees to replicate the input distribution with greater precision than Monte Carlo sampling. It uses a technique known as “stratified sampling without replacement” and proceeds as follows:

- The distribution is divided into n intervals of equal probability, where n is the number of iterations made on the model. These “strips” of probability start thin and become progressively wider toward the tails of the distribution as the density decreases, as shown in Figure 23.⁷⁹

⁷⁷ David Vose, *Risk Analysis – A Quantitative Guide*, 2nd Ed., John Wiley & Sons, 2000, pp. 57-58.

⁷⁸ Ibid., p. 59.

⁷⁹ Ibid.

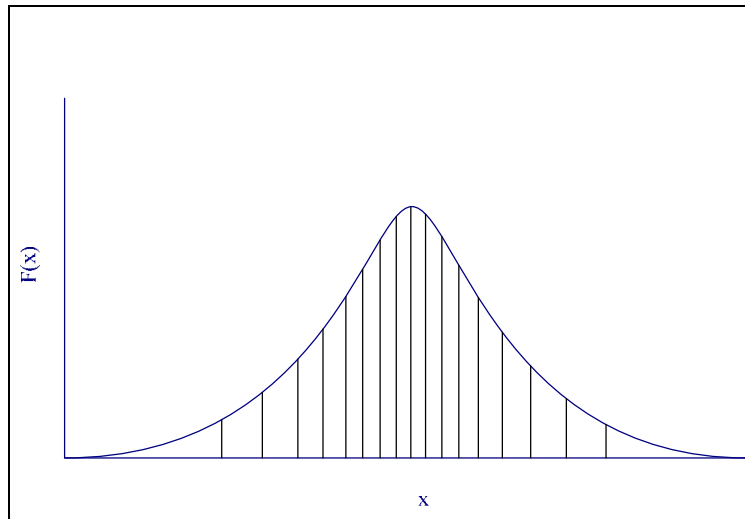


Figure 23. Effect of Stratification in Latin Hypercube Sampling (after)⁸⁰

- In the first iteration, one of these intervals is selected using a pseudo-random number.
- A second randomly generated number determines where, inside that interval, $F(x)$ should fall.
- $x = G(F(x))$ is calculated for that value of $F(x)$.
- The process is repeated for the second iteration, while the interval used in the first iteration is marked as “used” and will not be selected again (these one-dimensional samples are combined and randomly permuted to “cover” a unit hypercube in a stratified manner).
- This process repeats for each iteration, since the number of iterations n is the same as the number of intervals, each interval will only be sampled once and the distribution will be accurately reproduced over the range of $F(x)$.⁸¹

⁸⁰ David Vose, *Risk Analysis – A Quantitative Guide*, 2nd Ed., John Wiley & Sons, 2000, p. 59.

⁸¹ Ibid., p. 60.

Latin Hypercube Sampling consistently provides values for the distribution's statistics that are closer to the theoretical values of the input distribution than Monte Carlo Sampling.⁸²

We turned to Frontline Systems Risk Solve Engine and interactive simulation to power our stochastic spreadsheet model. This powerful software tool integrates with spreadsheet software to provide unprecedented simulation capabilities. We took advantage of the Mersenne Twister pseudo-random number generator and Latin Hypercube sampling available in Risk Solve Engine. With Risk Solve Engine we had the ability to execute 100,000 trials per simulation run with each recalculation of the model.⁸³ We built the stochastic spreadsheet model around the following equation:

$$\text{Total Value} = \text{Time On Station} * (\text{Value Rate}/100) + \text{Survivability Constant} - \text{Random Event Value}$$

We import the Schedule Table and the Coverage Table from the ASC-U output database into separate worksheets in our spreadsheet model. We calculate “Time On Station” by using the difference between Off Station Time and On Station Time for each mission in the Schedule worksheet. We pull the “Value Rate” directly from the Coverage worksheet and scale it by 100. Since the UAVs in this thesis are relatively small, not easily detected, and difficult to target, we included an empirically determined “Survivability Constant” to account for this fact. The Random Event Value is determined by a random process that chooses Sensor Failure $\sim \text{Lognormal}(\mu, \sigma)$, Maintenance problems $\sim \text{Exponential}(\lambda)$, Enemy Fire $\sim \text{Uniform}(0, 2)$, or No Event (value of 0) with an equal probability of 0.25 for each run of the model. Figure 24 provides a snapshot of the stochastic spreadsheet model, with representative values displayed for a particular run.

⁸² David Vose, *Risk Analysis – A Quantitative Guide*, 2nd Ed., John Wiley & Sons, 2000, p. 59.

⁸³ Frontline Systems, *Risk Solver Engine – User Guide*, Version 7, 2006, pp. 37 and 42.

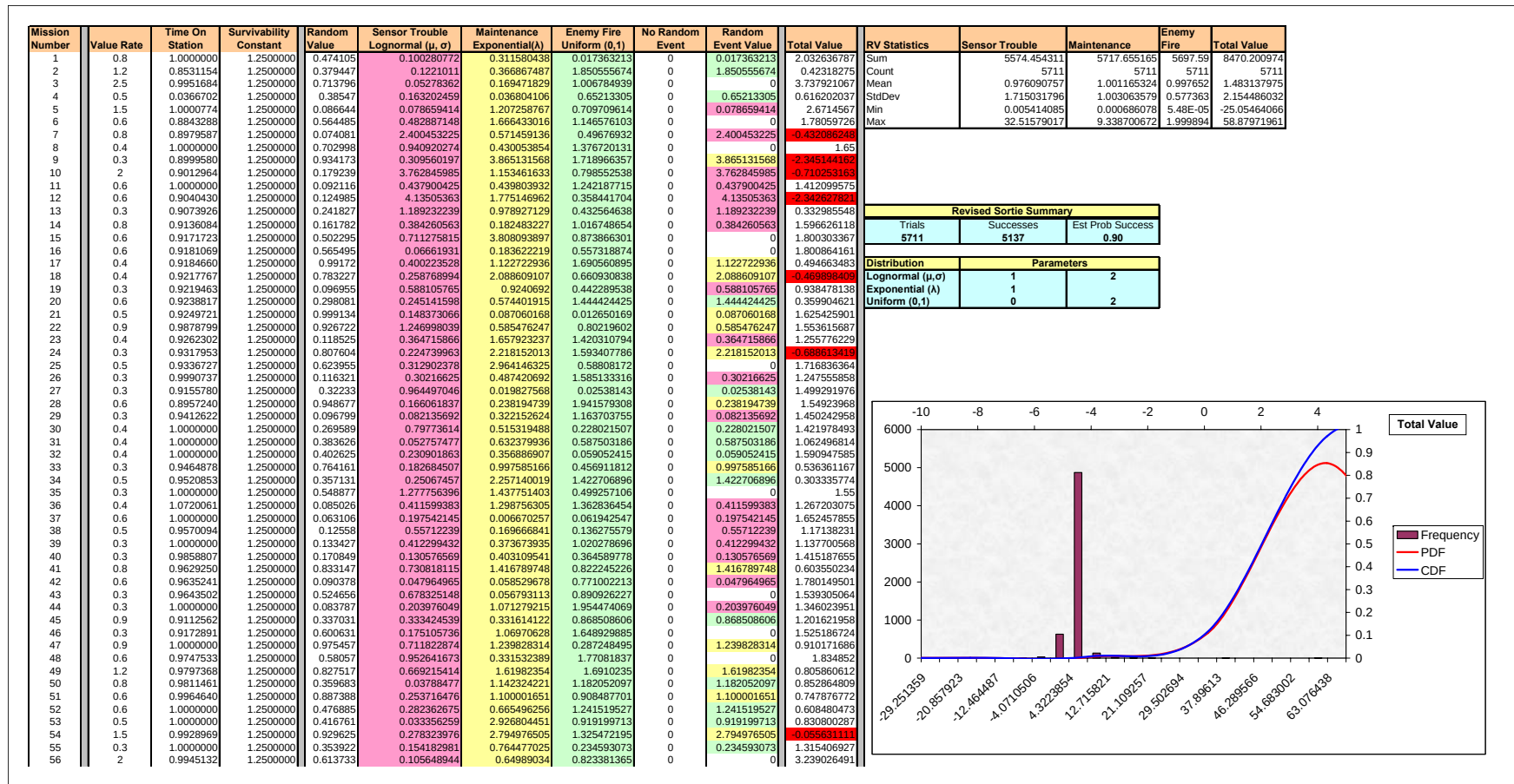


Figure 24. Stochastic Spreadsheet Model (Snapshot)

We randomly chose 30 of our 96 design points and used the Schedule and Coverage Tables from those ASC-U runs as input for the model. Each design contained 5,711 UAV missions (or trials). After the model makes the Total Value calculation, we look for two results. First, we look to determine which random event the model selected for each UAV mission. We do this by matching the appropriate adverse random event color code with the color in the Random Event Value column. The model is constructed such that the occurrence of an adverse random event does not automatically force a UAV sortie to lose mission coverage. Second, we look for negative numbers in the Total Value column (cells are conditionally formatted to turn red) that signify that the randomly selected adverse event value was greater than the product of Time On Station and the Value Rate plus the Survivability Constant. When Total Value is negative, we count the UAV sortie as unable to provide the required coverage. We make this calculation for each mission in the input file. The model computes an overall probability of success, along with standard summary statistics for Total Value, a histogram, and a plot of the probability density function (pdf) and cumulative density function (cdf). We ran 100,000 trials using Latin Hypercube Sampling for each simulation run and collected the probability of success (in the presence of adverse random events) for each design point. Figure 25 shows the results for the 30 randomly selected design points.

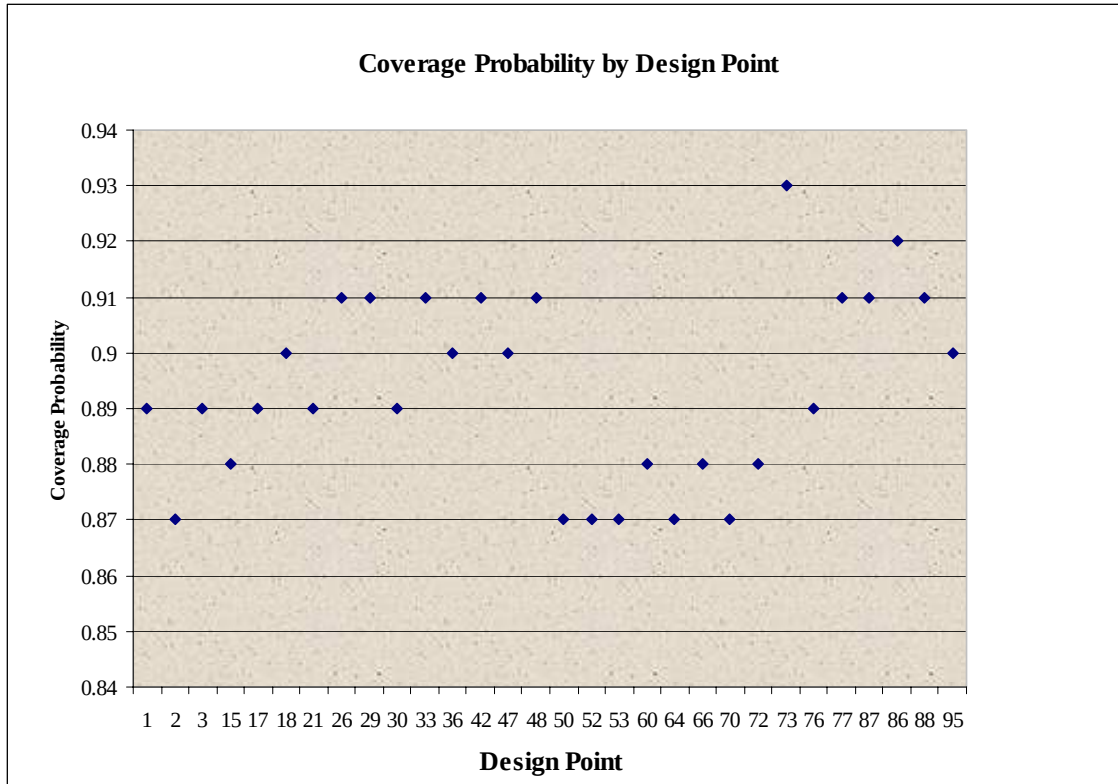


Figure 25. Coverage Probability by Design Point

Recall that we set out to quantify the potential degradation in UAV coverage in the presence of adverse random events. Figure 24 shows us that an operational commander might expect to experience a probability of successful UAV coverage that ranges from 0.87 to 0.93. We have made several distributional assumptions in the construction of this stochastic model that need to be more rigorously determined. Although the results are only approximations, the analysis provides some insight into how the fog and friction of war might create operational scenarios that are less than ideal.

This chapter discussed our methodology, experimental design, and data analysis for the ASC-U simulation output we collected. Chapter VI will present the conclusions from our analysis, discuss immediate impact, and provide recommendations for future research.

VI. CONCLUSIONS

The Modeling, Virtual Environments, and Simulations Institute (MOVES) and the U.S. Army Training and Doctrine Command (TRADOC) Analysis Center (TRAC) at the Naval Postgraduate School, Monterey, California, developed the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation to aid in the analysis of future U.S. Army UAV requirements. TRAC selected ASC-U to provide insight into the programmatic decisions addressed in the U.S. Army UAV-Mix Analysis that directly impact future development and fielding of the Future Combat System.

This thesis presents an alternative optimization model for ASC-U, explores the statistical and practical significance 23 simulation factors and their impact on the Total Value Measure of Effectiveness, and provides sensitivity analysis for how UAV coverage may degrade in the presence of adverse random events. We used robust experimental design, block analysis, and an innovative Optimized Flexible Latin Hypercube (OFLH) design consisting of 96 design points to evaluate a representative sample derived from an Army 2018 scenario. This robust design process gave us the ability to explore the ASC-U solution quality across a range of model factors. The analysis provides insights that will be useful to future modeling efforts and the operational elements of FCS. The conclusions suggest the following:

- The alternative optimization model developed in this thesis can successfully maximize value obtained by UAV assignment, while maintaining the required relationships between UAVs, mission packages, GCSs, and mission areas without the use of a heuristic.
- The Optimization Interval alone is not significant when the duration of the interval falls below one simulation hour.
- Incrementally smaller Optimization Intervals do not guarantee that a higher Total Value will result when the heuristics Early Return, Secondary Areas, and Appended Areas are included in model runs.

- ASC-U run times are long for Optimization Intervals below one simulation hour (ranging from a low of 8 hours to a high of 48 hours per run).
- To maximize Total Value in ASC-U runs, the heuristic Early Return should be set to FALSE;; the heuristic Secondary Areas should be set to TRUE; and the heuristic Appended Areas should be set to TRUE.
- ERMP Operating Time, Class IV Operating Time, and SUAV Operating Radius are the most significant UAV performance characteristics in this scenario.
- An OFLH design proves tremendously valuable for robust analysis of simulation models containing many (20 or more) factors and significantly decreases runtime.
- The long run times for Optimization Intervals below one simulation hour do not provide a significant improvement in the relative ASC-U solution quality.
- As the model factors change over pre-defined ranges, the solution quality appears to remain consistent with expected results.
-

A. IMMEDIATE IMPACT

The U.S. Army Chief of Staff tasked TRADOC to evaluate the current and future force options regarding UAVs to develop an investment strategy. This analysis supports the overall effort to provide rigorous analytical support for the Future Combat System UAV component. This has an impact on both the near-term and long-term Army force structure for Unmanned Aerial Systems and the U.S. Army UAV Mix Analysis.

This thesis addresses the TRAC directives by offering an alternative optimization model, providing an analysis of the optimization interval, robust experimental design of a representative sample of an Army 2018 scenario, and insight into how UAV coverage may degrade in the presence of adverse random events.

B. FOLLOW-ON RESEARCH

This thesis utilized integer programming techniques, robust experimental design, and sensitivity analysis to uncover numerous insights regarding the assignment of UAVs to mission areas in an advanced discrete event simulation. Although we have made significant progress in these areas, much work remains. We suggest the following list for future research.

- Implement the alternative optimization model developed in this thesis and compare the results to the existing optimization logic.
- Formulate a multicommodity network flow optimization model that takes advantage of the underlying network structure within the simulation.
- Study the effect that different numbers and differing types of UAVs have on overall solution quality.
- Examine the relationship between UAV Operating Time and UAV Time Horizon to determine whether an optimal relationship exists.
- Investigate additional operational scenarios developed by TRAC or other Department of Defense UAV communities.
- Expand the Design of Experiments integration to create a permanent front-end that embeds robust design in the model.
- Incorporate Probability Line of Sight into the model and explore the results against those generated by the existing simulation.
- Create a methodology to examine how ASC-U schedules might integrate with an Air Tasking Order process in an operational environment.
- Develop a decision analysis tool that captures the theoretical, computational and operational consequences of replanning UAV sorties that may be lost due to adverse random events.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX A. OPTIMIZED FLEXIBLE LATIN HYPERCUBE DESIGN

k1	k2	k3	k4	k5	k6	k7	k8	k9	k10	k11	k12	k13	k14	k15	k16	k17	k18	k19	k20	k21	d1	d2
1.383	247.287	434.783	41.948	43.826	0.939	220.957	67.826	5.739	6.261	0.187	68.661	16.696	1.389	1.76	0.887	219.304	143.478	6.783	7.174	0.239	0	0
1.565	268.243	565.217	42.574	44.609	0.991	277.2	70.435	5.009	6.391	0.13	70.122	15.861	1.365	1.847	0.922	192.313	132.609	5.739	9	0.761	0	0
1.722	226.33	582.609	31.93	52.435	0.87	245.061	76.957	6.261	8.739	0.146	67.2	13.913	1.412	1.359	0.991	209.183	123.913	7.096	6.783	0.309	0	0
1.696	205.374	539.13	37.565	42.261	0.922	233.009	83.478	4.904	6.522	0.138	99.339	18.922	1.083	1.617	1.061	178.817	134.783	6.887	6	0.343	0	0
1.748	251.478	408.696	36.939	51.652	1.2	237.026	66.522	6.47	6.913	0.135	81.809	16.974	1.177	1.387	1.13	202.435	121.739	4.8	7.043	0.657	0	0
1.643	234.713	521.739	35.061	36.783	0.852	241.043	79.565	5.948	8.087	0.172	100.8	13.078	1.06	1.674	0.852	232.8	126.087	5.426	8.217	0.83	0	0
1.67	196.991	417.391	33.183	50.087	1.096	224.974	90	6.887	6.652	0.174	86.191	15.583	1.342	1.732	0.8	172.07	150	6.052	8.348	0.483	0	0
1.774	276.626	469.565	35.687	41.478	1.043	265.148	74.348	6.783	7.435	0.19	75.965	15.304	1.201	1.933	1.096	185.565	104.348	6.991	6.13	0.691	0	0
1.33	255.67	495.652	33.809	49.304	1.009	253.096	75.652	4.8	7.696	0.185	92.035	18.643	1.506	1.445	0.817	165.322	108.696	5.948	6.391	0.9	0	0
1.8	243.096	556.522	34.435	36	1.078	196.852	86.087	5.426	7.304	0.182	77.426	18.365	1.577	1.502	1.183	195.687	113.043	5.53	8.87	0.378	0	0
1.409	201.183	547.826	36.313	45.391	0.8	273.183	60	6.991	6.13	0.177	93.496	15.026	1.53	1.473	1.148	175.443	128.261	5.217	7.435	0.448	0	0
1.513	192.8	573.913	43.2	48.522	0.957	184.8	69.13	6.574	8.87	0.169	71.583	17.809	1.13	1.875	0.957	161.948	115.217	5.322	7.696	0.726	0	0
1.278	259.861	426.087	40.07	40.696	0.887	257.113	84.783	6.678	8.478	0.161	78.887	19.2	1.224	1.416	0.939	222.678	119.565	5.113	7.304	0.17	0	0
1.617	285.009	400	40.696	50.87	0.817	216.939	78.261	5.53	9	0.159	94.957	14.191	1.483	1.56	1.113	168.696	139.13	6.47	8.478	0.552	0	0
1.461	272.435	504.348	30.678	39.13	1.026	204.887	63.913	5.113	7.826	0.167	80.348	13.357	1.154	1.703	1.009	158.574	147.826	4.904	6.522	0.135	0	0
1.487	209.565	513.043	38.191	46.174	1.183	249.078	82.174	5.217	7.957	0.151	83.27	12.8	1.553	1.789	0.87	212.557	110.87	5.009	6.652	0.274	0	0
1.357	222.139	452.174	38.817	37.565	1.113	188.817	62.609	6.052	6.783	0.148	87.652	13.635	1.295	1.33	0.904	188.939	100	7.2	8.087	0.517	0	0
1.591	289.2	530.435	32.557	46.957	0.904	212.922	71.739	7.096	7.043	0.133	96.417	17.252	1.436	1.961	0.835	182.191	102.174	5.843	7.826	0.1	0	0
1.226	238.904	460.87	30.052	53.217	0.835	192.835	87.391	5.322	6	0.156	74.504	14.47	1.248	1.818	1.165	215.93	106.522	5.635	7.565	0.622	0	0
1.539	217.948	443.478	29.426	39.913	0.974	208.904	61.304	6.157	8.348	0.14	84.73	18.087	1.6	1.904	1.026	229.426	141.304	6.157	6.913	0.865	0	0
1.2	230.522	486.957	28.8	38.348	1.061	269.165	80.87	6.365	8.217	0.143	73.043	16.417	1.271	1.646	1.043	155.2	130.435	6.574	8.609	0.587	0	0
1.304	213.757	478.261	39.443	47.739	1.148	261.13	73.043	5.635	8.609	0.164	97.878	16.139	1.318	1.99	1.2	205.809	117.391	6.678	7.957	0.204	0	0
1.252	280.817	600	41.322	43.043	1.13	200.87	88.696	7.2	7.174	0.153	89.113	14.748	1.459	1.588	1.078	199.061	145.652	6.261	6.261	0.796	0	0
1.435	264.052	591.304	31.304	54	1.165	228.991	65.217	5.843	7.565	0.18	90.574	17.53	1.107	1.531	0.974	226.052	136.957	6.365	8.739	0.413	0	0
1.383	247.287	434.783	41.948	43.826	0.939	220.957	67.826	5.739	6.261	0.187	68.661	16.696	1.389	1.76	0.887	219.304	143.478	6.783	7.174	0.239	1	0
1.565	268.243	565.217	42.574	44.609	0.991	277.2	70.435	5.009	6.391	0.13	70.122	15.861	1.365	1.847	0.922	192.313	132.609	5.739	9	0.761	0	0
1.722	226.33	582.609	31.93	52.435	0.87	245.061	76.957	6.261	8.739	0.146	67.2	13.913	1.412	1.359	0.991	209.183	123.913	7.096	6.783	0.309	0	0
1.696	205.374	539.13	37.565	42.261	0.922	233.009	83.478	4.904	6.522	0.138	99.339	18.922	1.083	1.617	1.061	178.817	134.783	6.887	6	0.343	0	0
1.748	251.478	408.696	36.939	51.652	1.2	237.026	66.522	6.47	6.913	0.135	81.809	16.974	1.177	1.387	1.13	202.435	121.739	4.8	7.043	0.657	0	0
1.643	234.713	521.739	35.061	36.783	0.852	241.043	79.565	5.948	8.087	0.172	100.8	13.078	1.06	1.674	0.852	232.8	126.087	5.426	8.217	0.83	0	0
1.67	196.991	417.391	33.183	50.087	1.096	224.974	90	6.887	6.652	0.174	86.191	15.583	1.342	1.732	0.8	172.07	150	6.052	8.348	0.483	0	0
1.774	276.626	469.565	35.687	41.478	1.043	265.148	74.348	6.783	7.435	0.19	75.965	15.304	1.201	1.933	1.096	185.565	104.348	6.991	6.13	0.691	0	0
1.33	255.67	495.652	33.809	49.304	1.009	253.096	75.652	4.8	7.696	0.185	92.035	18.643	1.506	1.445	0.817	165.322	108.696	5.948	6.391	0.9	0	0
1.8	243.096	556.522	34.435	36	1.078	196.852	86.087	5.426	7.304	0.182	77.426	18.365	1.577	1.502	1.183	195.687	113.043	5.53	8.87	0.378	0	0
1.409	201.183	547.826	36.313	45.391	0.8	273.183	60	6.991	6.13	0.177	93.496	15.026	1.53	1.473	1.148	175.443	128.261	5.217	7.435	0.448	0	0
1.513	192.8	573.913	43.2	48.522	0.957	184.8	69.13	6.574	8.87	0.169	71.583	17.809	1.13	1.875	0.957	161.948	115.217	5.322	7.696	0.726	0	0
1.278	259.861	426.087	40.07	40.696	0.887	257.113	84.783	6.678	8.478	0.161	78.887	19.2	1.224	1.416	0.939	222.678	119.565	5.113	7.304	0.17	0	0
1.617	285.009	400	40.696	50.87	0.817	216.939	78.261	5.53	9	0.159	94.957	14.191	1.483	1.56	1.113	168.696	139.13	6.47	8.478	0.552	0	0
1.461	272.435	504.348	30.678	39.13	1.026	204.887	63.913	5.113	7.826	0.167	80.348	13.357	1.154	1.703	1.009	158.574	147.826	4.904	6.522	0.135	0	0
1.487	209.565	513.043	38.191	46.174	1.183	249.078	82.174	5.217	7.957	0.151	83.27	12.8	1.553	1.789	0.87	212.557	110.87	5.009	6.652	0.274	0	0
1.357	222.139	452.174	38.817	37.565	1.113	188.817	62.609	6.052	6.783	0.148	87.652	13.635	1.295	1.33	0.904	188.939	100	7.2	8.087	0.517	0	0
1.591	289.2	530.435	32.557	46.957	0.904	212.922	71.739	7.096	7.043	0.133	96.417	17.252	1.436	1.961	0.835	182.191	102.174	5.843	7.826	0.1	0	0
1.226	238.904	460.87	30.052	53.217	0.835	192.835	87.391	5.322	6	0.156	74.504	14.47	1.248	1.818	1.165	215.93	106.522	5.635	7.565	0.622	0	0
1.539	217.948	443.478	29.426	39.913	0.974	208.904	61.304	6.157	8.348	0.14	84.73	18.087	1.6	1.904	1.026	229.426	141.304	6.157	6.913	0.865	0	0
1.2	230.522	486.957	28.8	38.348	1.061	269.165	80.87	6.365	8.217	0.143	73.043	16.417	1.271	1.646	1.043	155.2	130.435	6.574	8.609	0.587	0	0
1.304	213.757	478.261	39.443	47.739	1.148	261.13	73.043	5.635	8.609	0.164	97.878	16.139	1.318	1.99	1.2	205.809	117.391	6.678	7.957	0.204	0	0
1.252	280.817	600	41.322	43.043	1.13	200.87	88.696	7.2	7.174	0.153	89.113	14.748	1.459	1.588	1.078	199.061	145.652	6.261	6.261	0.796	0	0
1.435	264.052	591.304	31.304	54	1.165	228.991	65.217	5.843	7.565	0.18	90.574	17.53	1.107	1.531	0.974	226.052	136.957	6.365	8.739	0.413	0	0
1.383	247.287	434.783	41.948	43.826	0.939	220.957	67.826	5.739	6.261	0.187	68.661	16.696	1.389	1.76	0.887	219.304	143.478	6.783	7.174	0.239	0	0
1.565	268.243	565.217	42.574	44.609	0.991	277.2	70.435	5.009	6.391	0.13	70.122	15.861	1.365	1.847	0.922	192.313	132.609	5.739	9	0.761	0	0
1.722	226.33	582.609	31.93	52.435	0.87	245.061	76.957	6.261	8.739	0.146	67.2	13.913	1.412	1.359	0.991	209.183	123.913	7.096	6.783	0.309	0	0
1.696	205.374	539.13	37.565	42.261	0.922	233.009	83.478	4.904	6.522	0.138	99.339	18.922	1.083	1.								

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B. GENERAL ALGEBRAIC MODELING SYSTEM (GAMS) CODE

The following GAMS code implements an Integer Program for the optimization logic within the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) discrete event simulation.

The GAMS code uses a randomly generated number between 20 and 250 to provide a "value of assignment" coefficient in the objective function. Within ASC-U, this value is calculated by an upstream component and provided to the optimization routine by the simulation.

The objective of the ASC-U optimization logic is to maximize the value of UAV sensor coverage over designated mission areas (where sensor requirements exist inside a fixed time window in the future). After the optimization completes within ASC-U, the optimal assignment locations flow to a downstream component of the ASC-U simulation for UAV launch sequencing.

```
21
22 OPTIONS
23 optcr          =0
24 limrow         =100
25 limcol         =100
26 solprint      =on
27 LP            = XA
28 MIP           = XA
29 ;
30
31 SETS
32 u  uav                      /uav1*uav20/
33 l  launch and recovery Sites /lrs1*lrs20/
34 j  mission packages         /msnPkg1*msnPkg20/
35 a  mission areas            /msnArea1*msnArea20/
36 g  ground control stations  /gcs1*gcs20/
37 ;
38
39 SET U_l(l,u);
40
41     U_l("lrs1","uav1") = YES;
42     U_l("lrs1","uav2") = YES;
43     U_l("lrs2","uav3") = YES;
44     U_l("lrs2","uav4") = YES;
45     U_l("lrs3","uav5") = YES;
46     U_l("lrs3","uav6") = YES;
47     U_l("lrs4","uav7") = YES;
48     U_l("lrs4","uav8") = YES;
49     U_l("lrs5","uav9") = YES;
50     U_l("lrs5","uav10")= YES;
51
52 SET J_l(l,j);
53     J_l("lrs1","msnPkg1") = YES;
54     J_l("lrs1","msnPkg2") = YES;
55     J_l("lrs2","msnPkg3") = YES;
56     J_l("lrs2","msnPkg4") = YES;
57     J_l("lrs3","msnPkg5") = YES;
58     J_l("lrs3","msnPkg6") = YES;
59     J_l("lrs4","msnPkg7") = YES;
60     J_l("lrs4","msnPkg8") = YES;
61     J_l("lrs5","msnPkg9") = YES;
62     J_l("lrs5","msnPkg10")= YES;
63
64 SET A_l(l,a);
65     A_l("lrs1","msnArea1") = YES;
66     A_l("lrs1","msnArea2") = YES;
67     A_l("lrs1","msnArea3") = YES;
68     A_l("lrs2","msnArea1") = YES;
```

```

69     A_l("lrs2","msnArea2") = YES;
70     A_l("lrs2","msnArea3") = YES;
71     A_l("lrs3","msnArea1") = YES;
72     A_l("lrs3","msnArea2") = YES;
73     A_l("lrs3","msnArea3") = YES;
74     A_l("lrs4","msnArea1") = YES;
75     A_l("lrs4","msnArea2") = YES;
76     A_l("lrs4","msnArea3") = YES;
77     A_l("lrs5","msnArea1") = YES;
78     A_l("lrs5","msnArea2") = YES;
79     A_l("lrs5","msnArea3") = YES;
80
81     SET G_a(a,g);
82     G_a("msnArea1","gcs1") = YES;
83     G_a("msnArea2","gcs2") = YES;
84     G_a("msnArea3","gcs3") = YES;
85     G_a("msnArea4","gcs4") = YES;
86     G_a("msnArea5","gcs5") = YES;
87
88
89     PARAMETERS c(g), v(u,l,j,a), n(l,j), m(l,u);
90
91     *capacity of gcs g
92     c(g)= 1
93     ;
94     *value received by UAV u at lrs l with msnPkg j assigned to msnArea a
95     v(u,l,j,a) = uniform(20,250)
96     ;
97     *upper limit on the number of msnPkgs j at lrs l
98     n(l,j) = 5
99     ;
100    *upper limit on the number of uavs u at lrs l
101    m(l,u) = 6
102    ;
103
104    BINARY VARIABLES
105    x(u,l,j,a,g) 1 if UAV u with msnPkg j at lrs l is assigned to msnArea a and
106    d controlled by gcs g
107    ;
108    VARIABLES
109    z                objective function variable
110    ;
111
112    EQUATIONS
113    eqnObj           objective function
114    eqnSensor        constraint on sensors
115    eqnUAV           constraint on uavs
116    eqnMsnAreaCap    constraint on uavs assigned to msnAreas
117    eqnUAVcap        constraint on msnAreas assigned to uavs
118    eqnGCScap        constraint on gcs capacity
119    ;
120
121    eqnObj..         z =e= sum((U_l(l,u), J_l(l,j), A_l(l,a), G_a(a,g)), v(u,l,
122    j,a)*x(u,l,j,a,g))
123    ;
124    eqnSensor(l,j).. sum((U_l(l,u), A_l(l,a), G_a(a,g)), x(u,l,j,a,g)) =l
125    = n(l,j)
126    ;
127    eqnUAV(l,u)..    sum((U_l(l,u), J_l(l,j), A_l(l,a), G_a(a,g)), x(u,l,
128    j,a,g)) =l= m(l,u)
129    ;
130    eqnMsnAreaCap(a).. sum((U_l(l,u), J_l(l,j), G_a(a,g)), x(u,l,j,a,g)) =l
131    = 1
132    ;
133    eqnUAVcap(u)..    sum((J_l(l,j), A_l(l,a), G_a(a,g)), x(u,l,j,a,g)) =l
134    = 1
135    ;
136    eqnGCScap(g)..    sum((U_l(l,u), J_l(l,j), A_l(l,a)), x(u,l,j,a,g)) =l
137    = 1
138    ;

```

```

      = c(g)
132 ;
133
134
135 MODEL ASCUflow4 /eqnObj, eqnSensor, eqnUAV, eqnMsnAreaCap, eqnUAVcap, eqnG
    CScap/
136 ;
137 *option mip = mpswrite;
138 SOLVE ASCUflow4 USING MIP MAXIMIZING z;
139 display z.l, x.l;
140 display U_l, A_l, J_l, G_a;

```

COMPILATION TIME = 0.000 SECONDS 3 Mb WIN221-144 Mar 14, 2006

S O L V E S U M M A R Y

```

      MODEL    ASCUflow4      OBJECTIVE  z
      TYPE      MIP            DIRECTION MAXIMIZE
      SOLVER    XA             FROM LINE 138

**** SOLVER STATUS      1 NORMAL COMPLETION
**** MODEL STATUS      1 OPTIMAL
**** OBJECTIVE VALUE          687.3653

```

```

RESOURCE USAGE, LIMIT      0.078      1000.000
ITERATION COUNT, LIMIT    4          10000

```

---- EQU eqnSensor constraint on sensors

	LOWER	LEVEL	UPPER	MARGINAL
lrs1.msnPkg1	-INF	.	5.000	.
lrs1.msnPkg2	-INF	.	5.000	.
lrs1.msnPkg3	-INF	.	5.000	.
lrs1.msnPkg4	-INF	.	5.000	.
lrs1.msnPkg5	-INF	.	5.000	.
lrs1.msnPkg6	-INF	.	5.000	.
lrs1.msnPkg7	-INF	.	5.000	.
lrs1.msnPkg8	-INF	.	5.000	.
lrs1.msnPkg9	-INF	.	5.000	.
lrs1.msnPkg10	-INF	.	5.000	.
lrs1.msnPkg11	-INF	.	5.000	.
lrs1.msnPkg12	-INF	.	5.000	.
lrs1.msnPkg13	-INF	.	5.000	.
lrs1.msnPkg14	-INF	.	5.000	.
lrs1.msnPkg15	-INF	.	5.000	.
lrs1.msnPkg16	-INF	.	5.000	.
lrs1.msnPkg17	-INF	.	5.000	.
lrs1.msnPkg18	-INF	.	5.000	.
lrs1.msnPkg19	-INF	.	5.000	.
lrs1.msnPkg20	-INF	.	5.000	.
lrs2.msnPkg1	-INF	.	5.000	.
lrs2.msnPkg2	-INF	.	5.000	.
lrs2.msnPkg3	-INF	.	5.000	.
lrs2.msnPkg4	-INF	1.000	5.000	.
lrs2.msnPkg5	-INF	.	5.000	.
lrs2.msnPkg6	-INF	.	5.000	.
lrs2.msnPkg7	-INF	.	5.000	.
lrs2.msnPkg8	-INF	.	5.000	.
lrs2.msnPkg9	-INF	.	5.000	.
lrs2.msnPkg10	-INF	.	5.000	.
lrs2.msnPkg11	-INF	.	5.000	.
lrs2.msnPkg12	-INF	.	5.000	.
lrs2.msnPkg13	-INF	.	5.000	.
lrs2.msnPkg14	-INF	.	5.000	.
lrs2.msnPkg15	-INF	.	5.000	.
lrs2.msnPkg16	-INF	.	5.000	.
lrs2.msnPkg17	-INF	.	5.000	.

lrs2.msnPkg18	-INF	.	5.000	.
lrs2.msnPkg19	-INF	.	5.000	.
lrs2.msnPkg20	-INF	.	5.000	.
lrs3.msnPkg1	-INF	.	5.000	.
lrs3.msnPkg2	-INF	.	5.000	.
lrs3.msnPkg3	-INF	.	5.000	.
lrs3.msnPkg4	-INF	.	5.000	.
lrs3.msnPkg5	-INF	.	5.000	.
lrs3.msnPkg6	-INF	.	5.000	.
lrs3.msnPkg7	-INF	.	5.000	.
lrs3.msnPkg8	-INF	.	5.000	.
lrs3.msnPkg9	-INF	.	5.000	.
lrs3.msnPkg10	-INF	.	5.000	.
lrs3.msnPkg11	-INF	.	5.000	.
lrs3.msnPkg12	-INF	.	5.000	.
lrs3.msnPkg13	-INF	.	5.000	.
lrs3.msnPkg14	-INF	.	5.000	.
lrs3.msnPkg15	-INF	.	5.000	.
lrs3.msnPkg16	-INF	.	5.000	.
lrs3.msnPkg17	-INF	.	5.000	.
lrs3.msnPkg18	-INF	.	5.000	.
lrs3.msnPkg19	-INF	.	5.000	.
lrs3.msnPkg20	-INF	.	5.000	.
lrs4.msnPkg1	-INF	.	5.000	.
lrs4.msnPkg2	-INF	.	5.000	.
lrs4.msnPkg3	-INF	.	5.000	.
lrs4.msnPkg4	-INF	.	5.000	.
lrs4.msnPkg5	-INF	.	5.000	.
lrs4.msnPkg6	-INF	.	5.000	.
lrs4.msnPkg7	-INF	1.000	5.000	.
lrs4.msnPkg8	-INF	.	5.000	.
lrs4.msnPkg9	-INF	.	5.000	.
lrs4.msnPkg10	-INF	.	5.000	.
lrs4.msnPkg11	-INF	.	5.000	.
lrs4.msnPkg12	-INF	.	5.000	.
lrs4.msnPkg13	-INF	.	5.000	.
lrs4.msnPkg14	-INF	.	5.000	.
lrs4.msnPkg15	-INF	.	5.000	.
lrs4.msnPkg16	-INF	.	5.000	.
lrs4.msnPkg17	-INF	.	5.000	.
lrs4.msnPkg18	-INF	.	5.000	.
lrs4.msnPkg19	-INF	.	5.000	.
lrs4.msnPkg20	-INF	.	5.000	.
lrs5.msnPkg1	-INF	.	5.000	.
lrs5.msnPkg2	-INF	.	5.000	.
lrs5.msnPkg3	-INF	.	5.000	.
lrs5.msnPkg4	-INF	.	5.000	.
lrs5.msnPkg5	-INF	.	5.000	.
lrs5.msnPkg6	-INF	.	5.000	.
lrs5.msnPkg7	-INF	.	5.000	.
lrs5.msnPkg8	-INF	.	5.000	.
lrs5.msnPkg9	-INF	1.000	5.000	.
lrs5.msnPkg10	-INF	.	5.000	.
lrs5.msnPkg11	-INF	.	5.000	.
lrs5.msnPkg12	-INF	.	5.000	.
lrs5.msnPkg13	-INF	.	5.000	.
lrs5.msnPkg14	-INF	.	5.000	.
lrs5.msnPkg15	-INF	.	5.000	.
lrs5.msnPkg16	-INF	.	5.000	.
lrs5.msnPkg17	-INF	.	5.000	.
lrs5.msnPkg18	-INF	.	5.000	.
lrs5.msnPkg19	-INF	.	5.000	.
lrs5.msnPkg20	-INF	.	5.000	.
---- EQU eqnUAV constraint on uavs				
	LOWER	LEVEL	UPPER	MARGINAL
lrs1.uav1	-INF	.	6.000	.
lrs1.uav2	-INF	.	6.000	.

lrs2.uav3	-INF	1.000	6.000	.
lrs2.uav4	-INF	.	6.000	.
lrs3.uav5	-INF	.	6.000	.
lrs3.uav6	-INF	.	6.000	.
lrs4.uav7	-INF	.	6.000	.
lrs4.uav8	-INF	1.000	6.000	.
lrs5.uav9	-INF	1.000	6.000	.
lrs5.uav10	-INF	.	6.000	.

---- EQU eqnMsnAreaCap constraint on uavs assigned to msnAreas

	LOWER	LEVEL	UPPER	MARGINAL
msnArea1	-INF	1.000	1.000	.
msnArea2	-INF	1.000	1.000	223.161
msnArea3	-INF	1.000	1.000	220.584
msnArea4	-INF	.	1.000	.
msnArea5	-INF	.	1.000	.

---- EQU eqnUAVcap constraint on msnAreas assigned to uavs

	LOWER	LEVEL	UPPER	MARGINAL
uav1	-INF	.	1.000	.
uav2	-INF	.	1.000	.
uav3	-INF	1.000	1.000	.
uav4	-INF	.	1.000	.
uav5	-INF	.	1.000	.
uav6	-INF	.	1.000	.
uav7	-INF	.	1.000	.
uav8	-INF	1.000	1.000	.
uav9	-INF	1.000	1.000	.
uav10	-INF	.	1.000	.
uav11	-INF	.	1.000	.
uav12	-INF	.	1.000	.
uav13	-INF	.	1.000	.
uav14	-INF	.	1.000	.
uav15	-INF	.	1.000	.
uav16	-INF	.	1.000	.
uav17	-INF	.	1.000	.
uav18	-INF	.	1.000	.
uav19	-INF	.	1.000	.
uav20	-INF	.	1.000	.

---- EQU eqnGCScap constraint on gcs capacity

	LOWER	LEVEL	UPPER	MARGINAL
gcs1	-INF	1.000	1.000	243.620
gcs2	-INF	1.000	1.000	.
gcs3	-INF	1.000	1.000	.
gcs4	-INF	.	1.000	.
gcs5	-INF	.	1.000	.
gcs6	-INF	.	1.000	.
gcs7	-INF	.	1.000	.
gcs8	-INF	.	1.000	.
gcs9	-INF	.	1.000	.
gcs10	-INF	.	1.000	.
gcs11	-INF	.	1.000	.
gcs12	-INF	.	1.000	.
gcs13	-INF	.	1.000	.
gcs14	-INF	.	1.000	.
gcs15	-INF	.	1.000	.
gcs16	-INF	.	1.000	.
gcs17	-INF	.	1.000	.
gcs18	-INF	.	1.000	.
gcs19	-INF	.	1.000	.
gcs20	-INF	.	1.000	.

---- VAR x 1 if UAV u with msnPkg j at lrs l is assigned to msnArea a and controlled by gcs g

	LOWER	LEVEL	UPPER	MARGINAL
uav3 .lrs2.msnPkg4 .msnArea2.gcs2	.	1.000	1.000	.
uav8 .lrs4.msnPkg7 .msnArea3.gcs3	.	1.000	1.000	.
uav9 .lrs5.msnPkg9 .msnArea1.gcs1	.	1.000	1.000	.

APPENDIX C. VISUAL BASIC FOR APPLICATIONS SCRIPT FOR THE EXPERIMENTAL DESIGN INTERFACE

Option Explicit

Sub EDILoop1()

```

    Dim factorRow As Integer
    Dim dpRow As Integer
    dpRow = InputBox("Enter the next design point: ", "Update")
    If dpRow > 96 Then
        MsgBox "Only 96 design points exist. Please try again.", vbInformation,
"Exceeds Range"
    End If
    factorRow = dpRow + 1

    ActiveSheet.Range("A1:G2").Cells(2, 2) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 21)
    ActiveSheet.Range("A1:G2").Cells(2, 5) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 22)
    ActiveSheet.Range("A1:G2").Cells(2, 6) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 23)
    'ActiveSheet.Range("A1:G2").Cells(2, 7) =
Worksheets("NOLHfactors_22to29").Cells(factorRow, 24)
End Sub

```

Sub EDILoop2()

```

    Dim factorRow As Integer
    Dim dpRow As Integer
    dpRow = InputBox("Enter the next design point: ", "Update")
    If dpRow > 96 Then
        MsgBox "Only 96 design points exist. Please try again.", vbInformation,
"Exceeds Range"
    End If
    factorRow = dpRow + 1

    ActiveSheet.Range("A1:T388").Cells(2, 10) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 6)
    ActiveSheet.Range("A1:T388").Cells(2, 11) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 7)
    ActiveSheet.Range("A1:T388").Cells(2, 12) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 8)
    ActiveSheet.Range("A1:T388").Cells(2, 16) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 9)
    ActiveSheet.Range("A1:T388").Cells(2, 17) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 10)

    If ActiveSheet.Cells(3, 4) = "Class IV" Then
        ActiveSheet.Range("J3:J77") = ActiveSheet.Cells(2, 10)
        ActiveSheet.Range("K3:K77") = ActiveSheet.Cells(2, 11)
        ActiveSheet.Range("L3:L77") = ActiveSheet.Cells(2, 12)
        ActiveSheet.Range("P3:P77") = ActiveSheet.Cells(2, 16)
        ActiveSheet.Range("Q3:Q77") = ActiveSheet.Cells(2, 17)
    End If

```

End Sub

Sub EDILoop3()

```

    Dim factorRow As Integer
    Dim dpRow As Integer
    dpRow = InputBox("Enter the next design point: ", "Update")
    If dpRow > 96 Then

```

```

        MsgBox "Only 96 design points exist. Please try again.", vbInformation,
"Exceeds Range"
    End If
    factorRow = dpRow + 1

    ActiveSheet.Range("A1:T388").Cells(78, 10) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 1)
    ActiveSheet.Range("A1:T388").Cells(78, 11) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 2)
    ActiveSheet.Range("A1:T388").Cells(78, 12) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 3)
    ActiveSheet.Range("A1:T388").Cells(78, 16) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 4)
    ActiveSheet.Range("A1:T388").Cells(78, 17) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 5)

    If ActiveSheet.Cells(78, 4) = "ERMP" Then
        ActiveSheet.Range("J78:J101") = ActiveSheet.Cells(78, 10)
        ActiveSheet.Range("K78:K101") = ActiveSheet.Cells(78, 11)
        ActiveSheet.Range("L78:L101") = ActiveSheet.Cells(78, 12)
        ActiveSheet.Range("P78:P101") = ActiveSheet.Cells(78, 16)
        ActiveSheet.Range("Q78:Q101") = ActiveSheet.Cells(78, 17)
    End If

End Sub

```

```

Sub EDILoop4()

    Dim factorRow As Integer
    Dim dpRow As Integer
    dpRow = InputBox("Enter the next design point: ", "Update")
    If dpRow > 96 Then
        MsgBox "Only 96 design points exist. Please try again.", vbInformation,
"Exceeds Range"
    End If
    factorRow = dpRow + 1

    ActiveSheet.Range("A1:T388").Cells(102, 10) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 11)
    ActiveSheet.Range("A1:T388").Cells(102, 11) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 12)
    ActiveSheet.Range("A1:T388").Cells(102, 12) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 13)
    ActiveSheet.Range("A1:T388").Cells(102, 16) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 14)
    ActiveSheet.Range("A1:T388").Cells(102, 17) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 15)

    If ActiveSheet.Cells(102, 4) = "SUAV" Then
        ActiveSheet.Range("J102:J308") = ActiveSheet.Cells(102, 10)
        ActiveSheet.Range("K102:K308") = ActiveSheet.Cells(102, 11)
        ActiveSheet.Range("L102:L308") = ActiveSheet.Cells(102, 12)
        ActiveSheet.Range("P102:P308") = ActiveSheet.Cells(102, 16)
        ActiveSheet.Range("Q102:Q308") = ActiveSheet.Cells(102, 17)
    End If

End Sub

```

```

Sub EDILoop5()

    Dim factorRow As Integer
    Dim dpRow As Integer
    dpRow = InputBox("Enter the next design point: ", "Update")
    If dpRow > 96 Then
        MsgBox "Only 96 design points exist. Please try again.", vbInformation,
"Exceeds Range"
    End If

End Sub

```

```

factorRow = dpRow + 1

    ActiveSheet.Range("A1:T388").Cells(309, 10) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 16)
    ActiveSheet.Range("A1:T388").Cells(309, 11) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 17)
    ActiveSheet.Range("A1:T388").Cells(309, 12) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 18)
    ActiveSheet.Range("A1:T388").Cells(309, 16) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 19)
    ActiveSheet.Range("A1:T388").Cells(309, 17) = Workbooks("OFLH-N24K21-
Hernandez").Worksheets("Full N96 N23").Cells(factorRow, 20)

    If ActiveSheet.Cells(309, 4) = "TUAV" Then
        ActiveSheet.Range("J309:J388") = ActiveSheet.Cells(309, 10)
        ActiveSheet.Range("K309:K388") = ActiveSheet.Cells(309, 11)
        ActiveSheet.Range("L309:L388") = ActiveSheet.Cells(309, 12)
        ActiveSheet.Range("P309:P388") = ActiveSheet.Cells(309, 16)
        ActiveSheet.Range("Q309:Q388") = ActiveSheet.Cells(309, 17)
    End If

End Sub

```

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

- Aerovironment Web site, Raven RQ-11B, http://www.avinc.com/publish/2006/10/04/AV_DATASHEET_RAVEN_B_10_6_06.pdf, retrieved 11 April 2007.
- Ahner, D.K., Buss, A.H., and Ruck, J.L., *ASC-U Users/Analysts Manual*, TRADOC Analysis Center, DRAFT, June 2006.
- Cioppa, T.M., LTC, USA, "Efficient Nearly Orthogonal and Space-Filling Experimental Designs for High-Dimensional Complex Models," Doctoral Dissertation, Naval Postgraduate School, Monterey, CA, 2002.
- Cioppa, T.M. and Lucas, T.W., "Efficient Nearly-Orthogonal and Space Filling Latin Hypercubes," *Technometrics*, Volume 49, Number 1, pp. 45-55, 2007.
- Defense Update, "Extended Range Multi-Purpose UAV," *International Online Defense Magazine*, Issue 2, 2005, <http://defense-update.com/products/e/ermpUAV.htm>, retrieved 12 April 2005.
- Eikland, K. and Notebaert, P., "Introduction to lp_solve 5.5.0.10," <http://lpsolve.sourceforge.net/5.5/>, retrieved 14 April 2007.
- Fetterman, L., COL, USA, "14+1+1 Systems Overview," Future Combat System (Brigade Combat Team) (FCS (BCT)), White Paper, 14 March 2007, www.army.mil/fcs, retrieved 11 April 2007.
- Francis, P.L., "The Army's Future Combat Systems' Features, Risks and Alternatives," Congressional Testimony, United States General Accounting Office, GAO-04-635T, 1 April 2004.
- Hernandez, A., COL, USA, "Expanding the Combinatorial Space of the Nearly Othogonal Latin Hypercube," Doctoral Dissertation, Naval Postgraduate School, Monterey, CA, 2007, unpublished.
- Kleijnen, J.P.C., Sanchez, S.M., Lucas, T.W. and Cioppa, T.M., "A User's Guide to the Brave New World of Designing Simulation Experiments," *INFORMS Journal on Computing*, Volume 17, Number 3, pp. 263-289, 2005.
- Law, A.M. and Kelton, W.D., *Simulation Modeling and Analysis*, 3rd Ed., McGraw-Hill, 2000.
- Montgomery, D.C., *Design and Analysis of Experiments*, 6th Ed., John Wiley & Sons, 2005.

- Montgomery, D.C., Peck, E.A., and Peck, G.G., *Introduction to Linear Regression Analysis*, 3rd Ed., 2003, John Wiley & Sons.
- Nannini, C.J., “Analysis of the Assignment Scheduling Capability for Unmanned Aerial Vehicles (ASC-U) Simulation Tool,” Master’s Thesis, Naval Postgraduate School, Monterey, CA, June 2006.
- Office of the Secretary of Defense, “Unmanned Aircraft Systems Roadmap 2005-2030,” online document, July 2005, <http://www.acq.osd.mil/usd/Roadmap%20Final2.pdf>, retrieved 1 April 2007.
- Parsch, A., “Appendix 2: Modern UAVs,” *Directory of U.S. Military Rockets and Missiles*, <http://www.designation-systems.net/dusrm/app2/q-7.html>, retrieved 11 April 2007.
- Roque, A., “Army Will Cut Future Combat System’s Class II and Class III UAVs,” *InsideDefense.com*, Inside Washington Publishers, 12 December 2006, retrieved 10 April 2007.
- Rosenthal, R.E., *GAMS – A User’s Guide*, GAMS Development Corporation, Washington, D.C., 2006.
- White, B., Director, Brigade Combat Team Operations, *Future Combat System Smart Book*, TACOM, 18 September 2006.
- Woolsey, L.A., *Integer Programming*, Wiley-Interscience Publication, 1998.

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California
3. Professor Susan M. Sanchez
Department of Operations Research
Monterey, California
4. Lieutenant Colonel Darryl K. Ahner, United States Army
TRAC-MTRY
Monterey, California
5. Lieutenant Colonel Jeffrey Schamburg, United States Army
TRAC-MTRY
Monterey, California
6. TRADOC Analysis Center
ATTN ATRC (Mr. Bauman)
Fort Leavenworth, Kansas
7. Professor Arnold H. Buss
Modeling, Virtual Environments, and Simulation Institute
Monterey, California
8. Professor Kevin Wood
Department of Operations Research
Monterey, California
9. Colonel Alejandro Hernandez, United States Army
Graduate School of Operational and Information Sciences
Monterey, California
10. Major Derek M. Oliver, United States Air Force
Naval Postgraduate School
Monterey, California