

A Methodology for the Quantification of Doctrine and Materiel Approaches in a Capability-Based Assessment

A Dissertation
Presented to
The Academic Faculty

by

Steven Anthony Tangen

In Partial Fulfillment
Of the Requirements for the Degree
Doctor of Philosophy

School of Aerospace Engineering
Georgia Institute of Technology
May 2009

Copyright © 2009 by Steven Anthony Tangen

A Methodology for the Quantification of Doctrine and Materiel Approaches in a Capability-Based Assessment

Approved by:

Professor Dimitri Mavris
Advisor and Committee Chair
School of Aerospace Engineering
Georgia Institute of Technology

Professor Daniel Schrage
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Carlee Bishop
Senior Research Engineer
Georgia Tech Research Institute

Professor Mark Costello
School of Aerospace Engineering
Georgia Institute of Technology

Mr. David Brown
Air Force Research Laboratory

Date Approved: 3 April 2009

for Mom and Dad

ACKNOWLEDGEMENTS

This dissertation has been an awesome journey and if I have learned anything, it is that when walking a difficult path, reaching the destination cannot be done alone. I want to first thank my parents for all of their love and support which has been the foundation for all I have achieved in my academic career. I also want to thank my brothers who were a constant source of encouragement. Without my family, none of this would be possible.

I want to thank Tiffany Corbo for her constant, unyielding assistance and belief. Whether listening, proof-reading, or motivating, no one was closer to my day-to-day efforts and I could not have done this without you.

There have been many students at ASDL who helped me in one way or another during my time at Georgia Tech. I would like to thank Pat Biltgen for all of his assistance and guidance; my research team including Ben Poole, Diana Talley, Kelly Fry, Ryan Iandoli, Juan Cajigas, and Annie Jones; also William Engler, Stephane Dufresne, Santiago Balestrini, and Hernando Jimenez. It has been a pleasure to be a part of ASDL and to work with all of you. I would especially like to thank Russell Young, Chung Lee, and Jonathan Murphy for being my go-to-guys throughout my dissertation journey. You were invaluable to me from start to finish.

I would like to thank the American Society of Engineering Education and the Air Force Office of Scientific Research for supporting my graduate studies through the National Defense Science and Engineering Graduate Fellowship. It has been a tremendous experience and this support helped me this research possible.

I would like to thank each of my committee members, Dr. Daniel Schrage, Dr. Carlee Bishop, Dr. Mark Costello, and Mr. Dave Brown for their generous gifts of time and assistance. Each of you provided invaluable insight, feedback, suggestions, guidance and for all that I am forever grateful. Your knowledge, experience, and dedication were inspiring and have set the standards to which I will hold myself in my professional life.

And finally, I want to express my sincerest gratitude to my advisor, Dr. Mavris. Over the past four years I have learned more under your guidance than I could have possibly fathomed. I will be forever grateful for the both opportunities you provided me as well as the knowledge and experience which has become an indelible part of me. Thank you for all of your efforts to make me a better engineer and for helping make this dissertation a reality.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS	iv
LIST OF TABLES.....	viii
LIST OF FIGURES	ix
LIST OF ACRONYMS	xi
SUMMARY	xii
INTRODUCTION	1
1.1 Defense Acquisition.....	1
1.2 Military Transformation.....	5
1.3 Motivating Argument.....	6
1.4 DOTMLPF Downselect.....	9
1.5 Research Objective	11
1.6 Research Questions.....	11
1.7 Document Summary	13
BACKGROUND	14
2.1 Overview.....	14
2.2 What is Doctrine?	15
2.2.1 History of Doctrine	16
2.2.2 Modern Air Force Doctrine	17
2.2.3 Example	18
2.2.4 Disambiguation of Terms	20
2.3 Doctrine and Technology.....	23
2.4 Hypothesis #1.....	27
2.5 Example Problem.....	28
2.6 Defense Acquisition System	30
2.6.1 Joint Capabilities Integration and Development System	34
2.6.2 Detailed Capability-Based Analysis Process	36
2.7 Hypothesis #2.....	38
2.8 Formulation of Methodology Technical Challenges	38
2.9 Methodology Research Questions	41
METHDOLOGY FORMULATION.....	43
3.1 How to Quantitatively Evaluate Doctrine.....	43
3.1.1 Wargaming.....	44
3.1.2 Discussion on Modeling and Simulation	48
3.1.3 Selecting an Effectiveness Simulation Environment.....	53
3.1.4 Review of Studies in Doctrine and Capabilities	61

3.1.5	Review of Doctrine Analysis Techniques.....	64
3.1.6	Further Discussion on Agent-Based Modeling.....	68
3.2.	How to Accelerate Analysis.....	72
3.3.	How to Incorporate Cost.....	75
3.3.1	Cost Analysis	75
3.3.2	Cost Estimation Techniques	78
3.3.3	Costing Assumptions	80
3.4.	How to Present and Compare Doctrine and Materiel Results	81
3.4.1	Visualization Tools	82
3.5	Hypothesis #3.....	91
RESEARCH PLAN.....		92
4.1	Example Problem Elaboration	92
4.1.1	History of UAV Development	92
4.1.2	MQ-1 Predator	93
4.1.3	MQ-9 Reaper	95
4.1.4	UAVs Today	96
4.1.5	Challenges for Future Unmanned Aircraft	98
4.2	Selection of Study Problem	103
4.3	Verification and Validation.....	105
IMPLEMENTATION		109
5.1	Step I: Problem Definition	109
5.2	Step II: Create Analysis Environment	116
5.3	Step III: Baseline Analysis.....	135
5.4	Step IV: Gap Analysis	137
5.5	Step V: Identify Materiel/Non-Materiel Approaches	142
5.6	Step VI: Analysis of Materiel/Non-Materiel Approaches	160
5.7	Step VII: Reassess Gap.....	197
5.8	Step VIII: Recommendations.....	199
CONCLUSION.....		201
6.1	Revisit Research Questions.....	201
6.2	Summary of Methodology	205
6.3	Limitations of Methodology	211
6.4	Future Work	217
6.5	Research Conclusions	218
APPENDIX A	DOTMLPF Discussion.....	221
APPENDIX B	Taxonomy of Variables	231
APPENDIX C	Stepping Through A Simulation	235
APPENDIX D	Analysis Code	248
REFERENCES		267

LIST OF TABLES

Table 1: Summary of Selected Acquisition Programs [57], [58].....	2
Table 2: Summary of Simulator Attributes.....	58
Table 3: Input Variables for Baseline Analysis	135
Table 4: Frequency of Targets Killed in Baseline Scenario	136
Table 5: Frequency of Targets Killed in Advanced Scenario.....	138
Table 6: Statistics of Performance Gap	139
Table 7: Morphological Matrix of Approaches	147
Table 8: Notional Values for Aircraft Types	148
Table 9: Number of Levels per Variable	161
Table 10: Execution Times for Full Factorial Experiment.....	162
Table 11: Input Settings for Design of Experiments	163
Table 12: Aircraft Acquisition Costs	175
Table 13: Technology Impact Matrix	176
Table 14: Morphological Matrix of Methods	206
Table 15: Full Factorial Number of Cases.....	215

LIST OF FIGURES

Figure 1: Overview Chart of the Defense Acquisition Process	4
Figure 2: Enhanced View of Early Acquisition Process.....	7
Figure 3: Three Decision Support Processes of Defense Acquisition	30
Figure 4: Acquisition Process by Milestone [59]	33
Figure 5: Early JCIDS Process [59].....	35
Figure 6: Methodology Diagram	37
Figure 7: Decomposition of Experimentation Types.....	50
Figure 8: Notional Dynamic Sensitivity Derivatives.....	83
Figure 9: Changing Parametric Sensitivity Derivatives.....	84
Figure 10: Notional Multivariate Matrix	86
Figure 11: Filtering Y1	88
Figure 12: Filtering Y2	89
Figure 13: Results of Filtering Process.....	90
Figure 14: UAV Timeline [73]	93
Figure 15: MQ-1 Predator Specifications [148]	95
Figure 16: MQ-9 Reaper Specifications [149].....	96
Figure 17: Sample Task List From UJTL	112
Figure 18: Sample Task From AFTL.....	113
Figure 19: Basic NetLogo Display	118
Figure 20: Simulation Architecture	120
Figure 21: Setting up the Simulation Environment	122
Figure 22: Sensor Diagram	126
Figure 23: Diagram of the Hunter Algorithm	129
Figure 24: Analysis Code with Wrapper	134
Figure 25: Performance Histogram for Gap Analysis	139
Figure 26: Functional Decomposition of Mission	142
Figure 27: Functional Decomposition of Find.....	143
Figure 28: Functional Decomposition of Deploy Sensor	143
Figure 29: Functional Decomposition of Patrol.....	144
Figure 30: Three AOO Division Methods	150
Figure 31: Box Division of AOO.....	152
Figure 32: Random-Walk Patrol Pattern.....	154
Figure 33: Parallel Search Pattern.....	155
Figure 34: Orbit Patrol Pattern.....	156
Figure 35: Border Patrol Pattern	156
Figure 36: Dynamic Sensitivity Derivatives for Single Aircraft Type	165
Figure 37: New Area Division Yields New “Best” Search Pattern	165
Figure 38: Using Dynamic Sensitivity Derivatives to Illustrate Doctrine Interactions ..	166
Figure 39: Interaction between Materiel and Doctrine	168
Figure 40: Multivariate Plot for Single Aircraft Type	170
Figure 41: Filtered Multivariate Plot for Single Aircraft Type.....	171

Figure 42: Logic Tree for Computing Realization Time	179
Figure 43: Dynamic Sensitivity Derivatives for Complete Tradeoff Environment.....	181
Figure 44: Results of Monte Carlo Simulation for Complete Design Space.....	182
Figure 45: Pareto Optimal Among Immediate Solutions	183
Figure 46: Individual Frontiers for Immediate Solutions	184
Figure 47: Pareto Optimal for 1-year solutions	185
Figure 48: Progression of Pareto Frontiers over Time.....	186
Figure 49: Fixed Doctrine, Technology Infusion for Two Types Starting Points	187
Figure 50: Improved Doctrine, Technology Infusion.....	188
Figure 51: Change in Effectiveness due to Doctrine vs. Materiel	189
Figure 52: Varying Number of Aircraft.....	190
Figure 53: Design Space Filtered to 98% Requirement.....	191
Figure 54: Solutions Available in 2 Years.....	192
Figure 55: Solutions Available in 1 Year	193
Figure 56: Potential Paths to the Solution.....	194
Figure 57: Doctrine/Materiel Methodology with Enablers.....	207
Figure 58: Simulation Run Time	213
Figure 59: Department of Defense Organizational Chart.....	223
Figure 60: Basic NetLogo Display	236
Figure 61: Initialized Environment.....	237
Figure 62: Aircraft Patrolling Area of Operations.....	238
Figure 63: Querying Agents.....	238
Figure 64: Appearance of a Target	240
Figure 65: Agents Executing the Mission.....	242
Figure 66: Nearing Target Acquisition	243
Figure 67: Target Acquired.....	244
Figure 68: Pursuit and Engagement.....	245
Figure 69: Returning to Base	247

LIST OF ACRONYMS

AFDD	Air Force Doctrine Document
AMA	Analysis of Materiel/Non-Materiel Approaches
AOO	Area of Operations
CBA	Capability-Based Assessment
CONOPS	Concepts of Operation
CONEMP	Concept of Employment
CSAR	Combat Search and Rescue
DAG	Defense Acquisition Guidebook
DAS	Defense Acquisition System
DCR	DOTMLPF Change Recommendation
DoE	Design of Experiments
DOTMLPF	Doctrine, Organization, Training, Materiel, Leadership and Education, Personnel, and Facilities
EBO	Effects-Based Operations
FAA	Functional Area Analysis
FNA	Functional Needs Analysis
FSA	Functional Solution Analysis
ICD	Initial Capabilities Document
JCIDS	Joint Capabilities Integration Development System
M&S	Modeling and Simulation
PPB&E	Planning, Programming, Budgeting, and Execution
UAV	Unmanned Aerial Vehicle

SUMMARY

Due to the complexities of modern military operations and the technologies employed on today's military systems, acquisition costs and development times are becoming increasingly large. Meanwhile, the transformation of the global security environment is driving the U.S. military's own transformation. In order to meet the required capabilities of the next generation without buying prohibitively costly new systems, it is necessary for the military to evolve across the spectrum of doctrine, organization, training, materiel, leadership and education, personnel, and facilities (DOTMLPF). However, the methods for analyzing DOTMLPF approaches within the early acquisition phase of a capability-based assessment (CBA) are not as well established as the traditional technology design techniques. This makes it difficult for decision makers to decide if investments should be made in materiel or non-materiel solutions.

This research develops an agent-based constructive simulation to quantitatively assess doctrine alongside materiel approaches. Additionally, life-cycle cost techniques are provided to enable a cost-effectiveness trade. These techniques are wrapped together in a decision-making environment that brings crucial information forward so informed and appropriate acquisition choices can be made. The methodology is tested on a future unmanned aerial vehicle design problem.

Through the implementation of this quantitative methodology on the proof-of-concept study, it is shown that doctrinal changes including fleet composition, asset allocation, and patrol pattern were capable of dramatic improvements in system

effectiveness at a much lower cost than the incorporation of candidate technologies. Additionally, this methodology was able to quantify the precise nature of strong doctrine-dogma and doctrine-technology interactions which have been observed only qualitatively throughout military history. This dissertation outlines the methodology and demonstrates how potential approaches to capability-gaps can be identified with respect to effectiveness, cost, and time. When implemented, this methodology offers the opportunity to achieve system capabilities in a new way, improve the design of acquisition programs, and field the right combination of ways and means to address future challenges to national security.

CHAPTER I

INTRODUCTION

1.1 *Defense Acquisition*

The Department of Defense is the entity charged with “providing the military forces needed to deter war and protect the security of the United States [151].” Part of providing this protection means equipping the military with the best weapons, personnel, and procedures possible. Given that the United States has long maintained a position of military superiority largely through technological superiority, outfitting this force is a demanding and difficult job that requires significant resources [20], [114] .

To gain an appreciation for the magnitude of resources required, consider Table 1 below. It contains several major recent acquisition programs: the F-22 Raptor air superiority fighter aircraft, the F-35 Lightning Joint Strike Fighter, the AIM-120 Advanced Medium-Range Air-to-Air Missile (AMRAAM), the AEGIS radar and missile defense system, the SSN-774 Virginia Class nuclear attack submarines, and the GBU-39 small diameter bomb. For each program, the listed start date is when the military identified that some gap needed to be addressed and the IOC or initial operating capability is the date that the specific system entered, or is projected to enter, active military service. In addition to providing the total number of years from idea to implemented, Table 1 contains the total amount of money spent or projected to be spent on these systems.

Table 1: Summary of Selected Acquisition Programs [57], [58]

Program Name	Start Date	IOC	Length (years)	Total Program Cost
F-22 Raptor	1981	2004	23	\$65 billion
F-35 Lightning	1993	2012	19	\$300 billion
AIM-120 AMRAAM	1975	1991	16	\$13 billion
AEGIS	1969	1983	14	\$43 billion
SSN-774 Class	1993	2004	11	\$67 billion
GBU-39 SDB	2001	2006	5	\$4 billion

As is illustrated in this table, the decision to design and produce a new weapon system can be a lengthy and costly process. High profile programs like the F-22 have received wide-spread publicity but looking at radar systems and weapons suggests that any new system faces a daunting fiscal challenge. Across the board, new military systems require years, even decades, to complete and cost billions of dollars.

Keep in mind that these figures are not an indictment of any one of these defense programs. In fact, several of these programs are well regarded for their program management. The small diameter bomb has tremendous accuracy and power for a very reasonable cost [58]. The most recent submarine of the Virginia class, the USS New Hampshire, was delivered 8 months and \$54 million under budget [69]. The point here is that modern defense systems at any level are extremely complex and that complexity demands significant investment by the military and therefore the country.

Exploring the realm of complex and costly military systems further leads to an investigation of defense acquisition. According to the Defense Acquisition Guidebook, defense acquisition “exists to manage the Nation’s investments in technologies, programs, and product support necessary to achieve the National Security Strategy and support the United States Armed Forces [151].” Said another way, the acquisition process aim to procure under economic constraints the ways and means of the military branches

so they can affect United States security policy. A straightforward directive, the process of acquisition is a difficult one that must balance the needs of the warfighter, military leaders, political leaders, and the citizens it is designed to ultimately protect. Due to the demanding nature of acquisition, the process has many steps, many regulations, and therefore a great deal of complexity. Below in is the Integrated Framework Chart developed by the U.S. Department of Defense depicting *all* of the steps in the acquisition process [59].

Examining this chart can be overwhelming at first. There are a huge number of steps that cover everything from conceptual studies, to research and development, to prototyping and so forth. The time it takes to successfully accomplish different aspects of the defense acquisition process can vary widely from days to months to years depending on the system. The process chart and the operation of the acquisition process it depicts will be more fully explained later. For now, it is immediately obvious that there are a great many essential tasks to perform in an acquisition program and combined with the complexities of the technology being procured, it is a natural result that new military systems incur the types of cost and timeframes shown in Table 1.

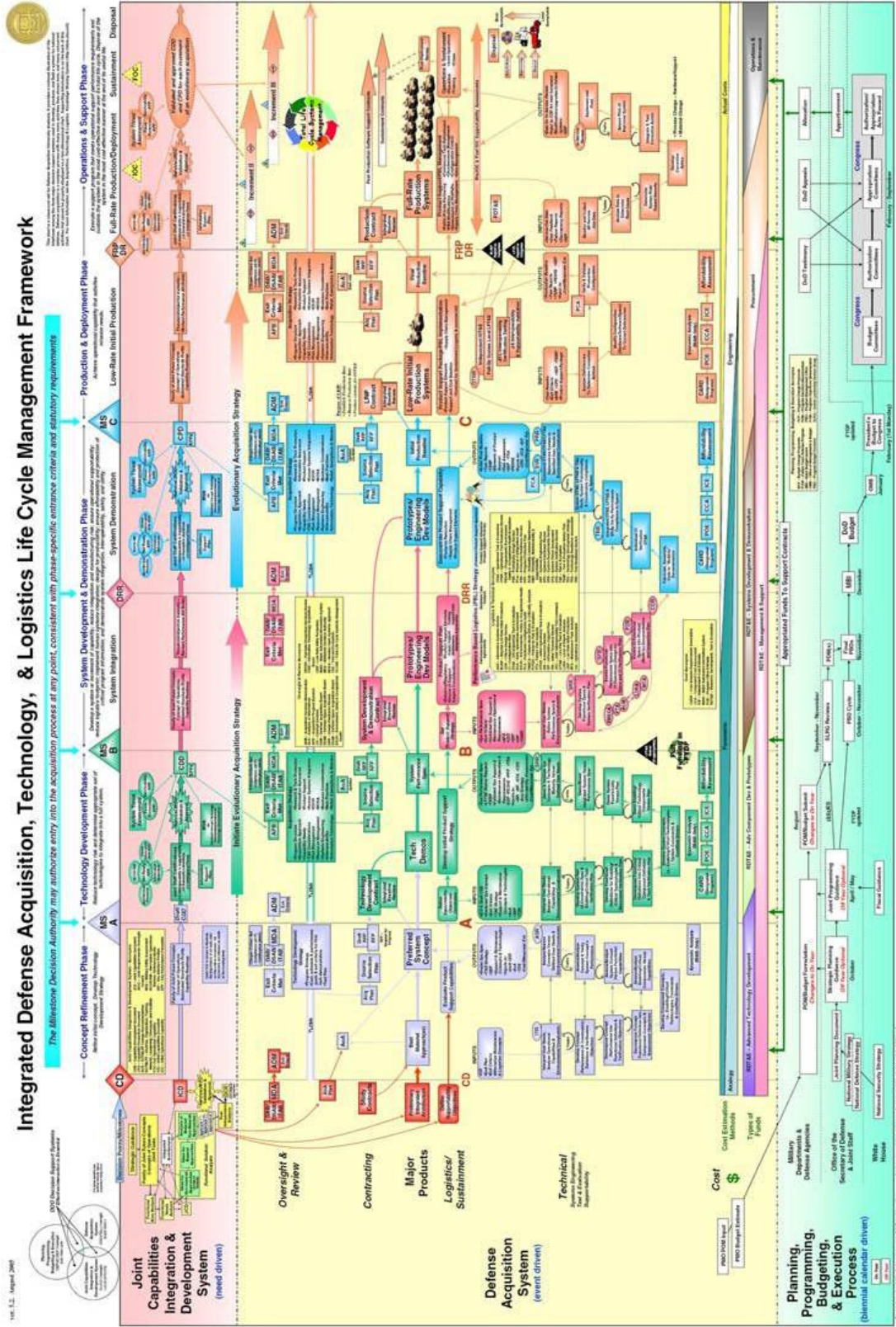


Figure 1: Overview Chart of the Defense Acquisition Process

1.2 *Military Transformation*

Only a few decades ago the predominant threat to U.S. security was the nation-state; wealthy developed countries with well defined borders and military forces. However, modern security challenges come from many disparate sources including networks of terrorists distributed across continents and within populations [115]. In 2001 the Quadrennial Defense Review stated many facts about the reality facing the United States. Rapid developments around the globe, such as the decline of the Soviet Union, can dramatically reshape the threat landscape. Today more than ever, adversaries are characterized by asymmetric threats. Conflicts are occurring in more and diverse areas. Groups of people, not just nation-states are capable of obtaining dangerous weapons, including weapons of mass destruction [114]. This reality was recognized and the Department of Defense (DoD) began to transform.

The military needed to deal with the new threat reality is quite different from that of yesteryear. For example, the Air Force has established Air and Space Expeditionary Forces (AEF) as a new model of organization [141]. The concept here is that forward deploying large contingents of aircraft is no longer necessary with the decline of the Soviet Union and in fact no longer possible as airbases of neutral countries are not open given the absence of immediate threat.

Furthermore, many of the assets and systems of the current military were designed for a combat reality from several years ago or longer. This raises the issue of whether or not the systems are appropriate and should also transform to meet the needs of the military. A classic example of this is the RAH-66 Comanche helicopter program. An expensive system that was meant to provide scout and fire support against the Soviet

Union, the Comanche was a victim of budget cuts when the threat it was designed to defeat dissolved. Examples such as this show that, in the face of military transformation, the ways and means of conducting warfare may have to change.

1.3 Motivating Argument

There are two statements to take from the above discussions. First, the United States military is undergoing a radical transformation and will need to acquire new military systems to complete this transformation. Secondly, designing and acquiring a new military system is costly and time consuming. From these two premises, the following conclusion can be made syllogistically: *The United States military will be forced to spend enormous amounts of money and time to complete its transformation.*

It can further be argued from this conclusion, that if military acquisition programs continue to be so lengthy then they may not come to fruition in time to affect national strategy. This in turn renders the assets ineffective, requiring additional acquisition and the process could reach a breaking point where acquisition failures negatively impact the formulation of national security policy instead of policy driving acquisition.

This conclusion and its ensuing implication are no doubt disconcerting. The first question that should be asked is: Is there another way? This conclusion is based on two premises regarding transformation and acquisition as well as several implicit assumptions. Assumed within this discussion is the basic tenet of economics, namely that there is a scarcity of resources and infinite purchasing of new systems is not possible. This assumption is solid and for all practical purposes should be taken as fact. Likewise, within the second premise there is an assumption that new acquisition is costly and will remain that way. It is acknowledged that a great deal of research and energy is being put

into reducing design costs and cycle times but it is also asserted that at some point, there is a practical limit to how inexpensive a new system can be. Finally, the first premise states that the military is going through a transformation and will have to buy new systems. While the first part of that statement is backed by extensive evidence, the notion that acquiring new systems is the only way to meet the transformation is an assumption that again begs the question: Is there another way?

Returning to the complete defense acquisition process chart in Figure 1, there is a series of steps at the very beginning where the requirements for a potential system are being defined. These tasks are shown enlarged in Figure 2 relative to the overall defense acquisition system (modified from [59]).

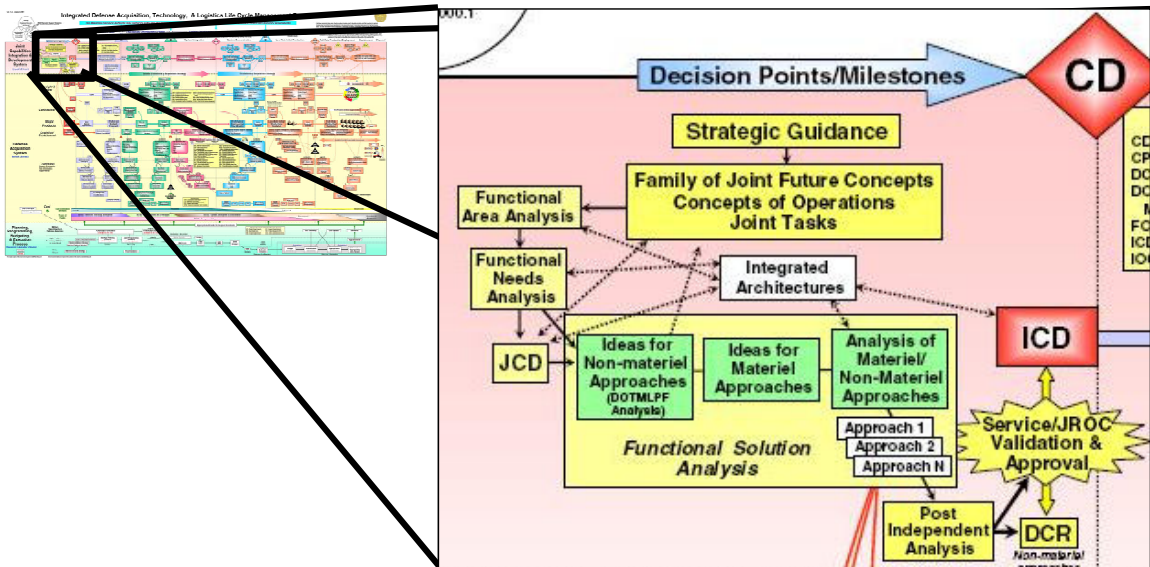


Figure 2: Enhanced View of Early Acquisition Process

This collection of tasks is known as a capability-based assessment, or CBA. A CBA takes military needs and translates them into system requirements. There are three main segments of a CBA. The functional area analysis defines what the military needs to

do. The functional needs analysis specifies the current ability to meet those needs. Finally, the functional solution analysis examines various options to determine which is best for addressing the needs. Within the functional solution analysis a complete range of materiel and non-materiel options are considered and recommendations are forwards from the CBA to the rest of the defense acquisition system.

Before continuing, consider the distinction between materiel and non-materiel approaches. Materiel items are the physical objects that the military uses to perform its functions. Aircraft, jeeps, bullets, radios, and boots could all be considered “materiel”. Non-materiel is essentially all the supporting elements of the defense system, particularly anything that does not require a new acquisition program. There is a term in the defense community for the total spectrum of materiel and non-materiel items: DOTMLPF. This stands for doctrine, organization, training, materiel, leadership and education, personnel, and facilities. The CBA is meant to consider potential solutions across all of these categories.

If the CBA results in a non-materiel solution being the best option and a new acquisition program is not needed, then many of the steps in Figure 1 do not need to be performed. A non-materiel solution could be increased training, additional purchases of already existing assets, new fighting tactics, and so forth. All of these options reduce the time line and cost associated with meeting a military need. It appears that another way has been found.

If there does exist a mechanism in the acquisition process for procuring non-materiel solutions to capability gaps, the next question is how well does this mechanism work? How does a Defense Department official decide whether to put his or her last

dollar into buying a new system or additional training or developing new doctrine? This may sound like a simple question but there is no simple answer. As it turns out, according to a recent presentation by the Air Force Studies and Analysis Agency, there are still issues with how non-materiel solutions are analyzed. Some of the highlights of this presentation are listed below:

- Rigorous application of a framework is necessary but not sufficient for success
- Measuring individual capabilities is not difficult; comparing the value/worth of different capabilities is the hard part
- Difficult to measure DOTLPP solutions vs. [materiel] solutions – need techniques to help do this.[130]

Examining these conclusions shows there is no consistent, rigorous, and traceable way to assess the different DOTMLPP options against each other. In fact, these statements say that the only class of approaches the military feels comfortable analyzing is M; the materiel approaches. If the non-materiel approaches cannot be analyzed to the same level of accuracy as the materiel approaches, then they are less likely to be fielded thereby incurring the cost of a new acquisition program where it may not be needed.

1.4 DOTMLPP Downselect

Simply put, DOTMLPP is broad and all encompassing. A complete CBA should investigate all approaches from all aspects of DOTMLPP. However, for the development of a methodology it is beyond the scope to examine all possible options. Therefore, it is necessary to select a subset of DOTMLPP. First it should be determined which area or

areas the DoD has the most interest in changing since the more relevant the study, the more useful the results will be. According to the CBA guidance, when performing a functional solution analysis, the analysts should investigate approaches in the following order:

1. Doctrine, organization, and leadership/education
2. Policy changes
3. Personnel and training
4. Improved existing materiel and facilities
5. Other agency materiel
6. Cooperative development
7. New materiel programs [21]

From this list, it is evident that military leadership has recognized that new materiel programs are extremely expensive and should be pursued only as a last resort.

This prioritized list has now scoped the problem to doctrine, organization, and leadership/education by considering only the highest priority approaches. Of this subset, it is worth examining how readily a comparison can be made between the non-materiel and the materiel. Leadership and education is the most difficult of these because there is less information available about this topic. Additionally, with little available information and less experience with the topic, it will be much more difficult to achieve relevant results by exploring professional development.

That leaves only doctrine and organization. Of the two, doctrine has historically held a closer tie with technology and materiel solutions than organization. Every new weapon gets employed in a new way. Both categories hold great interest to military planners but for the purposes of this study, the primary non-materiel approach category that will be considered is doctrine.

1.5 Research Objective

In summary, the acquisition of new materiel systems has historically been a costly and time consuming endeavor. At a time when the military is facing a period of great and rapid transition, it may not be possible to satisfy all defense needs with such acquisitions. There is an alternative which calls for making changes to the non-materiel aspects of the military, such as doctrine. The mechanism for analyzing and selecting from amongst materiel and doctrinal options is called a capability-based assessment. However, the quantification of doctrinal impacts has historically fallen short of the standards for CBAs making it extremely difficult to determine if cheaper and faster means for meeting military needs exist. What is needed is a methodology that rigorously quantifies doctrine in such a way that it can be evaluated in cost-effectiveness terms relative to materiel approaches.

The objective of this research is to develop a methodology that will quantify doctrinal changes and permit trade-offs of doctrine and materiel approaches in the context of a capability-based assessment.

1.6 Research Questions

Based on the desire to understand and incorporate doctrine into the capability-based assessment process the next step is to carve out specific research questions that will guide the remainder of this dissertation. The first research question is:

1. What is the relationship between the doctrinal design variables and materiel design variables in a capability-based aircraft system-of-systems design study?

The term doctrine has various connotations and it may be unclear how doctrine fits in to the overall defense acquisition picture. The method developed should be able to provide insight into the doctrine-materiel relationship and a deeper quantitative understanding would be immensely valuable.

The next research question is:

2. How much and in what way does the inclusion of a quantitative representation of doctrine influence the outcome of a capability-based assessment?

It has been stated that CBA's are being performed in the acquisition community however they do not include doctrine. So if doctrine now becomes a player in acquisition decisions, how much will the process and the results of a CBA change? This is an important element to understand in order to apply the method developed herein to future problems.

Finally, the third major research question is:

3. How can a method be developed that rigorously compares doctrinal approaches to materiel approaches within the context of a capability-based assessment?

There is a lack of quantitative techniques to analyze doctrine and then compare to materiel approaches. This main question will guide the literature review as it will be necessary to determine what elements are needed in a method, what the options are for those elements, and how to implement the method as a whole.

1.7 Document Summary

The remainder of the document breaks down in the following manner. Chapter II includes background on the fields of military doctrine, defense acquisition, and the example problem. Additionally, Chapter II culminates in the articulation of technical challenges and the genesis of additional research questions. In Chapter III, the methodology and hypotheses are generated through a focused literature review derived from the research questions in Chapter II. Chapter IV explains the example problem in greater detail and also summarizes the implementation of the method and how the experiments will be performed to address the hypotheses. Chapter V is a step-by-step account of the implementation of the methodology for the test case provided. This chapter provides great detail on how the experiments were constructed and performed as well as the outcomes of those experiments. Finally, Chapter VI revisits the research questions and hypotheses, draws conclusions from the research, provides lessons learned, and proposes future work. Additionally, there are several appendices which contain additional background detail and the code from the simulation models created for experimentation.

CHAPTER II

BACKGROUND

2.1 Overview

The research objective presented above is well grounded in need and if realized will provide a significant advancement to the state of the art in military acquisition. It has already been shown however that the problem shall be scoped to quantification of doctrine out of the set of all non-materiel approaches. Additionally, the space of sample problems is too large and a proof of concept must be selected. Furthermore, the entirety of the defense acquisition system has been mentioned thus far. However, it is not possible to research everything all at once. The goal is to provide a measureable increase in ability to analyze the doctrinal approaches alongside of materiel ones. Demonstrating this capability on a small sample problem within a subset of defense acquisition and DOTMLPF will be sufficient for this dissertation and serves to establish a foundation for future research. This chapter will first explore further the concept of doctrine. Also, the defense acquisition process shall be addressed in additional detail. Then, an example problem which will serve as the proof of concept demonstration shall be detailed. Finally, additional technical challenges will be identified which in turn lead to more research questions, setting the stage for the methodology development.

2.2 What is Doctrine?

Since doctrine plays a central role in this dissertation it is imperative to gain a thorough understanding of the subject, beginning at the very foundation and examining it from all sides. Studies of doctrine can be confounded by the wide array of definitions available in many distinct fields. According to Ehrhart,

A fundamental problem with Air Force doctrine is the absence of any real consensus as to what doctrine is and just what it is supposed to do...By trying to stretch a single term, "doctrine," to accommodate all things, we wind up with an amorphous concept that falls short in all areas. [38]

This statement underscores the importance of clearly defining doctrine as well as reinforces the difficulty in translating doctrine into capability-based solutions.

Beginning with the most general definition, Merriam-Webster defines the term doctrine in several ways. It can be associated with teaching, principles of law, or a statement of policy [70]. These terms are vague and diverse, leading to confusion regarding how doctrine should be interpreted in the context of this problem. Therefore, this section disambiguates terms and provides a literature survey in order to gain an appreciation for what doctrine is, as well as some of the research being performed on doctrine today. The outline of this chapter is as follows. First, a brief history of Air Force doctrine is presented to provide historical context. Second, this chapter explains how the Air Force defines doctrine today. Following this definition a survey of terms that often are used and misused when considering doctrine is presented. Finally there is a summary of how doctrine changes today along with a thorough literature review of state-of-the-art research in doctrine. By the end of this chapter it will be clear what is meant by the term doctrine.

2.2.1 History of Doctrine

The roots of Air Force doctrine stretch back to a time before there was even a distinct Air Force service. As aircraft and the accompanying technology matured during World War II, military planners saw that these new machines could provide a crucial advantage in combat that might avoid the stalemate-bloodbaths of World War I [71]. However, even within the ever growing circle of people who saw the advantages of air power, there were stark differences of opinion regarding the manner in which air power should be used. One camp was of the opinion that the primary role of air power was to support ground offensives. On the other hand, there were those who felt that air power could be used independently and equal to ground forces [71]. To address this and other debates the War Department published a field manual, FM 100-20, entitled “Command and Employment of Air Power” in 1943. This was the first official document regarding how air power would be used in this conflict and beyond.

FM 100-20 established several key tenets that would shape air power development in the United States. This document stated that air power was an equal and independent entity with land power and neither should be an auxiliary of the other. Furthermore, this document established that the first task of air power should be establishment of air superiority; then and only then could the full potential of air power be exploited [156]. The concept of centralized control-decentralized execution was put forth here as well. This is the notion that a central controller will direct the overall activities of the units in a particular theater but will leave the nuts-and-bolts of the execution to the individual aircraft. Many ideas similar to these were set forth in this document and established the beginning of air power doctrine.

2.2.2 Modern Air Force Doctrine

The above history gives an idea about the origins of Air Force doctrine as the original guiding principles for the use of air power in combat. Now the question becomes: what is the state of doctrine today? First consider how the Air Force defines doctrine: “Air and Space doctrine is a statement of officially sanctioned beliefs, warfighting principles, and terminology that describes and guides the proper use of air and space forces in military operations [142].” Basically, doctrine is the answer to the question: how should the military fight?

While doctrine is “official”, doctrine is not necessarily a set of hard and fast rules that must be followed to the letter without exception. Doctrine is meant to be flexible as well. Doctrine is a set of guiding principles and beliefs and can come from sources other than published manuals. A perfect example of this is the doctrine behind the air war during Operation Desert Storm. The strategy developed by Col. John Warden was not accepted doctrine when it was implemented in Iraq. However, Warden’s ideas were implemented and achieved great success [157]. On one hand, doctrine is authoritative, documented, instructions. On the other hand it is a guide that, when tailored to specific operations, becomes the means by which the Air Force executes its missions.

2.2.2.1 Levels of Doctrine

Now that the general concept of doctrine has been presented it is possible to further describe Air Force doctrine today. There are three levels that are used to describe doctrine in the Air Force: Basic, Operational, and Tactical. In that order they proceed from general to specific. Each level of doctrine will be defined below and an example will be provided to demonstrate the differences.

Basic doctrine is the broadest level of doctrine and constitutes the most general warfighting principles. Examples of basic doctrine include centralized control and decentralized execution or establish air superiority. These are the type of principles that were laid forth in FM 100-20 are found today in Air Force Doctrine Document 1 [142]. While there are some differences between these two documents, basic doctrine has remained virtually unchanged since 1943.

Operational doctrine is more specific than basic doctrine and relates the use of air and space power to the operations and functions that the Air Force must perform. This level of doctrine includes the operation of command and control centers as well as how they plan and use their assets and intelligence to achieve effects. The steps needed to execute a mission are considered operational doctrine [142].

Tactical doctrine is the most specific level of doctrine. It delves down to the asset level and speaks very specifically to one mission or another. Additionally, tactical doctrine is prescribed in terms of assets, environmental variables, and objectives in the mission [142].

2.2.3 Example

To better understand the meaning of each level of doctrine, the following section examines sample statements from official Air Force doctrine documents at each of the three levels. The example mission is combat search and rescue (CSAR). This example was chosen because the tactical doctrine manuals for CSAR are not classified like the manuals for other missions, thus allowing for a consistent comparison through all three levels.

According to AFDD1, combat search and rescue is the set of “air operations conducted to recover distressed personnel during wartime or contingency.” This document further states how the Air Force will “organize, train, and equip personnel to conduct CSAR missions, using the fastest and most effective means, across the range of military operations [142].” Note the broad, general language. Reading this document, it becomes quite clear that CSAR is an important mission for the Air Force to perform. However, these statements are not sufficient to actually conduct the rescue of an individual.

Air Force Doctrine Document 2-1.6, entitled “Personnel Recovery Operations” is the operational level doctrine statement for the U.S. Air Force. The goals of this document are to elaborate on how CSAR should be conducted, what types of assets might be used, who has authority, how information gets relayed, and how training and planning should be conducted. One example of the level of detail is the set of five steps for all CSAR missions: report, locate, support, recover, reintegrate. Under the “report” heading, the document says “Awareness and notification initiate the [CSAR] process. Rapid and accurate notification is essential for a successful recovery. Threat conditions permitting, [isolated persons] should attempt to establish contact with friendly forces [132].” Compare this level of detail with that of basic doctrine. Basic doctrine stated that CSAR is important while operational doctrine provides the steps of a CSAR mission as well as standards for the execution of that mission.

Joint Publication 3-50.21 entitled “Joint Tactics, Techniques, and Procedures for Combat Search and Rescue” is the authoritative document about how CSAR operations are performed at the tactical doctrine level. This document seeks to provide guidance to

military personnel conducting CSAR operations. The level of detail here is much finer than that in the previous two documents. As an example, consider the following statements within a section titled “Awareness and Notification.” One subsection discusses the actions of aircrews who witness a crash. It states the aircrew should “remain in the area as conditions permit or until relieved by other aircraft. Do not circle directly over the survivor. This may serve to mark the survivor’s position for hostile forces [75].”

Another example instruction is “Consider switching IFF to EMERGENCY and transmitting “MAYDAY on GUARD frequency. This technique should be carefully weighed against the probability of enemy detection [75].” Clearly these two statements are much more specific than basic or operational doctrine. Instead of talking about the importance of CSAR or the way the operation should be performed, this doctrinal statement gives guidance for specific actions directly to the personnel. Additionally, notice how both statements contain qualifiers that direct the personnel to change their actions if the conditions are not favorable. This reinforces the earlier statement that while doctrine is authoritative, it requires judgment for employment.

2.2.4 Disambiguation of Terms

One of the most difficult aspects when dealing with doctrine is that the term means so many things to so many different people. One’s interpretation depends not only on one’s field, but also on the level of experience and even the culture in which one learned his or her field. The above discussion outlined how doctrine is defined by the United States Air Force and addressed the scope of doctrine. However, in order to enable meaningful discussion of the analyses that follow, this section will elaborate on several terms that often are used and misused when discussing doctrine. Please note that while

every effort has been made to do a thorough review, there are likely other interpretations of the terms that follow. The author does not detract from these views nor is this paper meant to dissuade the reader from further examining items of interest. However, it is necessary to delineate, for the purposes of this paper, clear distinctions between terms so that the reader can logically follow the research approach and conclusions that follow.

2.2.4.1 Tactics and Doctrine

The first term to consider is ‘tactics’. Much has been said so far about tactical doctrine but is this the same thing as tactics? Recall that doctrine was defined as foundational principles that guide the use of military forces [72]. Now it is important to establish a well-founded definition for tactics. According to the DoD Dictionary of Military and Associate Terms, tactics are defined as “the employment and ordered arrangement of forces in relation to each other. Also procedures, techniques [72].” Another definition comes from Shaw’s book “Fighter Combat: Tactics and Maneuvering” in which he states tactics include attack formations, firing positions, speeds, engagement decision criteria, and similar parameters [118]. In general most literature presents tactics as the nuts-and-bolts of military operations while doctrine is the broad general principles that guide operations. However, it was shown earlier in Joint Publication 3-50.21 that a doctrinal document can have very specific guidance as well. Consider an additional statement from the CSAR tactical doctrine manual regarding performing a daytime rescue over land:

When the CSAR recovery element is on the ingress route at approximately 2-4 miles from the extraction point, the wingman establishes separation of approximately 1/4 mile. When the LZ is in sight, the lead aircraft commences to the approach and landing. On short final, the wingman passes off the right side of

lead, maintaining an airspeed of 80-100 knots and a distance of approximately 400 meters. [75]

Clearly the level of detail here matches Shaw's description of what constitutes a tactic as well as the DoD Dictionary definition. Therefore it seems reasonable to conclude, for the purposes of this paper, that *tactical doctrine and tactics are synonymous*.

2.2.4.2 Strategy and Doctrine

The DoD Dictionary defines strategy as “a prudent idea or set of ideas for employing the instruments of national power in a synchronized and integrated fashion to achieve theater, national, and/or multinational objectives [72].” This definition speaks to very high level goals and crosses domains other than just military. Shaw has a slightly different view of strategy. He classifies it as preparations and decisions about the battle made beforehand that are meant to allow the accomplishing of large-scale objectives [118].

Mazzei defines strategy as “a course of action”, as opposed to the fundamental principles of doctrine [92]. Yarger considers strategy to be a delicate balance between goals and the available ways, means, and resources [163]. Strategy is about planning so as to achieve a favorable outcome and increase the probability of that outcome when compared to random chance.

Clearly, strategy and doctrine are not the same. Strategy is a goal-oriented plan while doctrine is guidance for employment of resources. However, strategy is concerned with high-level objectives which suggests *there exists an important relationship between strategy and basic level doctrine*.

2.2.4.3 CONOPS and Doctrine

Concepts of operation, or CONOPS, are “a verbal or graphic statement that clearly and concisely expresses what the joint force commander intends to accomplish and how it will be done using available resources [72].” An Air Force Instructional manual describes CONOPS as how a command decides to deploy and use a system as a way to meet a mission requirement. Again, contrast this with doctrine defined as guidance for the employment of forces. CONOPS appears to be closely linked to operational and possibly even tactical doctrine though definitely not synonymous with either. CONOPS are very specific to a particular mission and system and it can be said that *CONOPS are a specific instantiation of doctrine*.

2.3 Doctrine and Technology

Now that doctrine has been fully defined, it is appropriate to examine the role doctrine has played in warfare. Of particular interest is the historical linkage between doctrine and technology. If doctrine is the how of military operations and technology is the what, then one might anticipate that there is a very strong connection between the two. In fact, doctrine and technology have been compared to “the chicken and the egg”; one does not exist without the other and it cannot be determined which one came first [38]. In order to fully explore the connection between these two pillars of national defense, it is worth a look back at several illuminating historical examples.

First consider the development of beyond visual range (BVR) missiles and their employment in the 1960s. At the time, improving missile and radar performance so that engagements could take place before the target aircraft was ever seen was thought to be a revolution in warfare. However, between the years 1965 and 1982, of the 632 firings of

missiles capable of attacking targets beyond visual range, only 10% were actually fired beyond a range of 5 miles. Furthermore, during that entire time, only four BVR kills were reported [157]. All remaining air to air engagements were executed with munitions other than BVR missiles or the BVR missiles were not employed at the limits of their envelope. The primary reason for this minimal exploitation of a new technology is that the doctrine required positive identification of the target prior to initiating an engagement. The rules of engagement prevented pilots from attacking unless certain criteria were met so this is one example where the implementation of technology was heavily constrained by doctrine.

The next example involves the development of the machine gun in the early 1900s. While this new technology was being developed there were some armies who were not modifying their doctrine. Therefore, at the onset of World War I, many of the Allied armies were decimated by using Napoleonic style frontal assaults in the face of machine gun fire [38],[48]. A similar example can be found in naval doctrine. For years after the introduction of moveable turrets, naval battles were fought in classic broad-side fashion with often disastrous results [48].

On the opposite side of the equation, there are also examples where doctrine evolved faster than technology. A perfect example of this is the early strategic bombing raids of World War II. It was thought that the use of strategic bombers attacking vital enemy population centers would decimate national will and vital industries. However, the doctrine at the time relied exclusively on the bombers conducting the attacks without organic defense capability or escorting aircraft. Until the P-51 was effectively integrated

into these raids, bombers were lost at alarming rates and were not able to bring about the type of strategic effect that was anticipated [38].

These examples all illustrate the disastrous results when doctrine and technology are not aligned. There are also examples which illustrate impact of using compatible doctrine and technology. First, consider the use of laser guided bombs (LGBs) in Operation Desert Storm compared to Vietnam. In Vietnam, LGBs were viewed as an improvement over the current state of the art and increased accuracy on the destruction of targets. However, the selection of targets and the doctrine of bombing changed very little if at all. By comparison, in Desert Storm, the doctrine changed and LGBs were used to strike precisely at key centers. Through the view of an enemy as a system-of-systems the choice of how and where LGBs were used created a profound effect against Iraq. Instead of simply using fewer bombs to take out a single target as was the benefit of LGBs in Vietnam, the use of these precision munitions in Desert Storm on parallel and interconnected targets dramatically reduced Coalition casualties and quickly achieved victory [157].

Finally, perhaps one of the most widely known examples of a doctrinal change influencing the outcome of a military engagement is the German Blitzkrieg of 1940. At the time, Allied forces actually held a slight numerical advantage and according to some sources, a technological edge in tanks and other equipment. However, it was the German side which adopted a combined arms approach and through coordinated attacks was able to achieve profound victory [157],[38].

All of these examples, whether positive or negative, prove that there is a definite linkage between doctrine and technology which is capable of significantly altering the

outcome of a military action. The United States in Vietnam or the Soviet Union in Afghanistan are poignant reminders that technological superiority is neither a necessary nor a sufficient condition for victory. There are many other factors and one of the most important must be doctrine. It is reasonable to conclude that neither technological advancement nor new doctrine in isolation are not nearly as powerful as coordination of the two. The goal should be to find a “better fit between doctrine and technology [157].”

Another point to consider is the notion of technology development in a static doctrine framework. For one such example, consider again Germany in the 1940s. While the operational doctrine of blitzkrieg was sound, through much of the war, technologies were developed without any consideration of national strategy or objectives. Germany was unable to develop an atomic weapon but poured massive resources into ballistic missiles and even hypersonics research [48]. Developing new technologies for technology sake can possibly lead to advances, but often times these programs result in dead-end projects or require expensive modifications to fully transition technology to operational. Examples of such projects include the XB-70 bomber or the development of an atomic airplane [48].

Doctrine and technology feed each other; they are almost impossible to separate. Designing them together is likely to lead to a better solution. According to Hallion, “Neither [doctrine and technology] is independent of the other; rather, each generates a synergistic impulse that encourages and strengthens the other [48].” There is value in making technology changes. There is also value in simply modifying doctrine. A concise summary of what the above historical examples indicate is offered by Berg:

Doctrine deserves respect because it has been intertwined with technology for millennia and explains how to use technological tools to achieve military

purposes. Skillfully integrating doctrine and technology can lead to victory, but technology without doctrine to guide its use has little military significance. [10]

2.4 Hypothesis #1

Thus far the discussion of doctrine seems to indicate that there is a strong historical tie between doctrine and technology. Whenever a new system has been invented, the method of employment has changed as well. Also, many of the key advancements in centuries of warfare have come about not by new technologies, but by new concepts in the conduct of war.

Based on this historical trend, the hypothesis to Research Question 1 is stated thusly:

Doctrine variables will provide greater cost-effectiveness than materiel design variables. Furthermore, the interaction effect between doctrine and materiel will substantially improve the effectiveness of both.

2.5 Example Problem

This research effort is a proof of concept, a demonstration, and a study into the applicability and capability of a new design approach. In order to adequately investigate the method, an example problem is needed to serve as a test-bed. The example problem should be well defined, manageable in scope, and have sufficient documentation or information available to aid in the construction of experiments to support the research questions defined in this dissertation.

In fact, the original notion for this research was borne from a study being led by the Air Force Research Laboratory in coordination with Air Combat Command, the Georgia Tech Research Institute, and Georgia Tech's Aerospace Systems Design Laboratory which is the author's organization. The program is a systems engineering approach to formulating an ICD for a Revolutionary Hunter/Killer aircraft. Basically, the idea is to employ various design methods to traceably develop an affordable approach that can be the future of the Air Force's unmanned attack fleet. Currently, the Air Force employs the MQ-1 Predator and MQ-9 Reaper in this role but the mission will soon grow beyond the capabilities of these aircraft.

Now, in the course of the above research effort, it was noted that, as a means of simplifying the problem, only materiel approaches were being considered in the ICD. Further research into non-materiel approaches yielded the information in other chapters in this document showing the deficiencies that the design has in terms of capability to assess non-materiel approaches. Given that the Revolutionary Hunter/Killer program is of high priority to the Air Force and is already well defined along with the fact that there exists a great deal of public domain information on the MQ-1 and MQ-9 this problem

was determined to be an ideal example problem for the investigation into non-materiel approaches.

Later in this document the current generation of unmanned air systems, in particular the Predator and Reaper, will be defined in terms of their capability, performance, and relation to the larger picture of Air Force operations. Furthermore this document will identify the envisioned future state of unmanned air systems. Once this information is gathered, it will be possible to highlight critical issues and from there, determine what experiments will be performed in this research project.

2.6 Defense Acquisition System

Earlier in the document, the defense acquisition process was presented as a whole unit as a way to illustrate the complexity involved. However, it is now essential to delve deeper into the workings of the defense acquisition system. This will not only help scope the problem but also illuminate the relationship of this research to the larger picture.

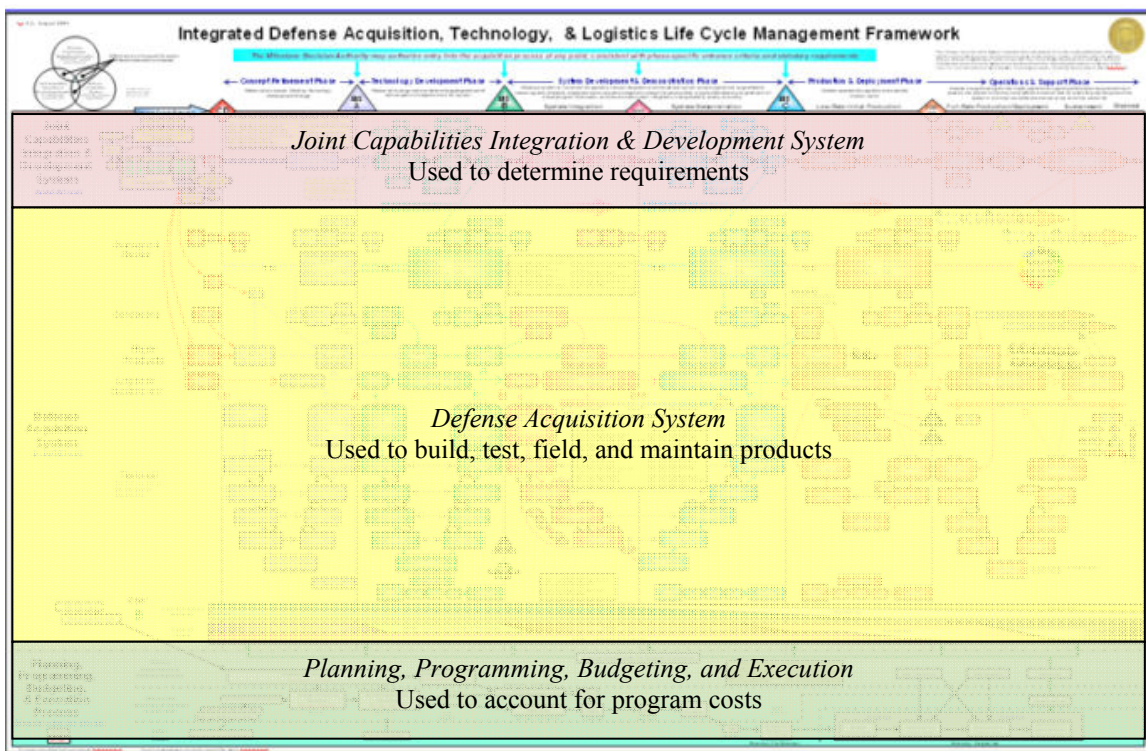


Figure 3: Three Decision Support Processes of Defense Acquisition

Recall the complex and extensive chart in Figure 1 showing all of the defense acquisition system; the number of steps and amount of information exchange seems overwhelming. However, taken at a high-level view, there are three main areas of this chart that can be explained to help simplify the problem. There are three separate sub-

processes, called decision support processes that make up defense acquisition and are shown in Figure 3 by the color-coded sections (modified from [59]). These three processes are the Joint Capabilities Integration and Development System (JCIDS), Defense Acquisition System (DAS), and Planning, Programming, Budgeting, and Execution (PPBE) which are depicted in the top pink row, middle yellow row, and bottom green row respectively. Examining each of these three areas one at a time will help scale down the problem to the relevant area of interest.

First, JCIDS is the way by which the DoD identifies what it needs to do from broad strategy and then distills that information down into a set of capabilities that must be performed [151]. The term capability here means “the ability to achieve a desired effect under specified standards and conditions through combinations of means and ways” [21]. The notion of designing for capabilities is a relatively recent paradigm shift in defense acquisition. Basically, there are certain effects or tasks the military must create or perform and they must be done to specified standards in specified conditions. *The capability itself is independent of the ways and means used to achieve it.* That statement is at the heart of what JCIDS is all about.

By focusing on the joint forces, that is to say the four military branches and potentially other agencies as the situation warrants, JCIDS is meant to reduce redundancies, leverage similar acquisition efforts, and get the right means into the right place at the right time. A true design for capabilities does not presume what those means are. As an example, a theoretical capability might be: Deliver lethal effects to time-sensitive hardened targets in a jungle environment within 1 hour in all weather. A high-speed aircraft with heavy weapons or an airdropped SEAL team with high explosives

might both be viable ways and means for meeting that capability. A thorough JCIDS analysis will have considered all potential options. The JCIDS will output to the other decision support systems what the eventual ways and means should be so they can be created and delivered.

Next, the DAS, the middle row of Figure 3, is the “management process that guides all DoD acquisition programs [151].” Through a series of decision milestones, the DAS takes the concept or concepts generated by JCIDS and creates viable and useable products for the warfighter. The Defense Acquisition System covers research and development, prototyping and testing, low-rate production, sustainment, and eventually disposal [151]. The DAS row covers the majority of the chart and when one examines how many steps go into the actual creation of a product, it becomes clear why.

Finally, the bottom row in the chart is the set of tasks for the PPBE decision support process. Essentially, PPBE is the process by which the Defense Department manages its investments[151]. As the name illustrates, PPBE consists of four overlapping phases: planning, programming, budgeting, and execution. Planning takes place at the highest military levels, i.e. the Office of the Secretary of Defense and the Joint Chiefs. It is a way to take national policy and strategy and align those needs with resources to develop and sustain acquisition programs. The programming stage takes the strategic guidance from the planning stage and develops specific programs that affordably meet the national goals. More specific than the planning stage, programming involves creating resource allocations over time as well as what-if scenarios should a given program not reach fruition. Concurrently, the budgeting stage frames the programs in budgetary terms that coincide with what is required for Congressional approval. Finally, the execution

stage is set up to provide leadership with information about how well the allocation of resources is supporting the programs which in turn support the national military strategy [151].

Each of the three decision support processes has been defined separately but in actuality, they can only function when working in concert. At each point in time, the complete defense acquisition framework must be able to answer the following questions: What needs to be done; what can we technically accomplish, and what is economically viable?

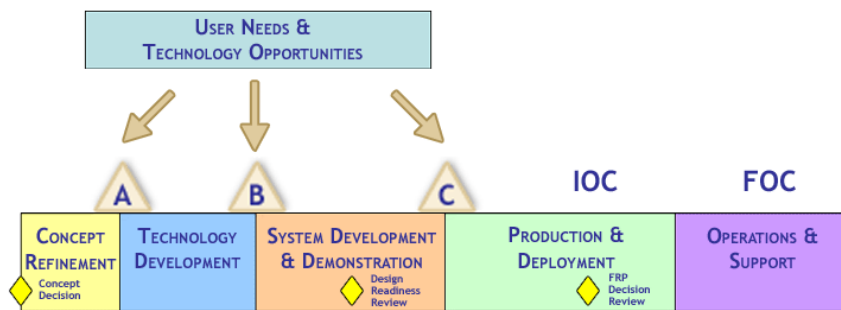


Figure 4: Acquisition Process by Milestone [59]

Figure 4 above shows each stage of the acquisition process and the accompanying milestones where decisions are made. The same milestones can be found in Figure 1 but this simplified diagram more readily addresses the following point. Before each milestone (A, B, IOC, etc) the three decision support systems must share information and coordinate their efforts. The three key questions listed just above must be answered at different levels of fidelity by various groups at the different stages of acquisition. While the above discussion is clearly not meant to be a complete description of defense

acquisition, it should illustrate the relationships between different pieces as well as give a flavor for the enormous scope of the complete process.

Now recall that this research is interested in addressing how doctrinal approaches influence not only aircraft design but also total system-of-system capabilities. From the brief description of the three decision support systems the one most relevant to the research goal of this thesis is JCIDS, where the capabilities and conceptual design are performed. The next section will go into greater detail about the JCIDS to further develop the research framework.

2.6.1 Joint Capabilities Integration and Development System

Figure 5 below shows the early steps of the JCIDS taken from upper-left portion of the complete defense acquisition diagram in Figure 1. The reason these steps were highlighted is because this represents the start of the entire acquisition process. Key decisions are made here that dictate what will happen throughout the rest of design, development, operation, and so forth. JCIDS can essentially be thought of as the means by which system or concept requirements are created. In fact, the predecessor of the JCIDS was the Requirements Generation System or RGS [76].

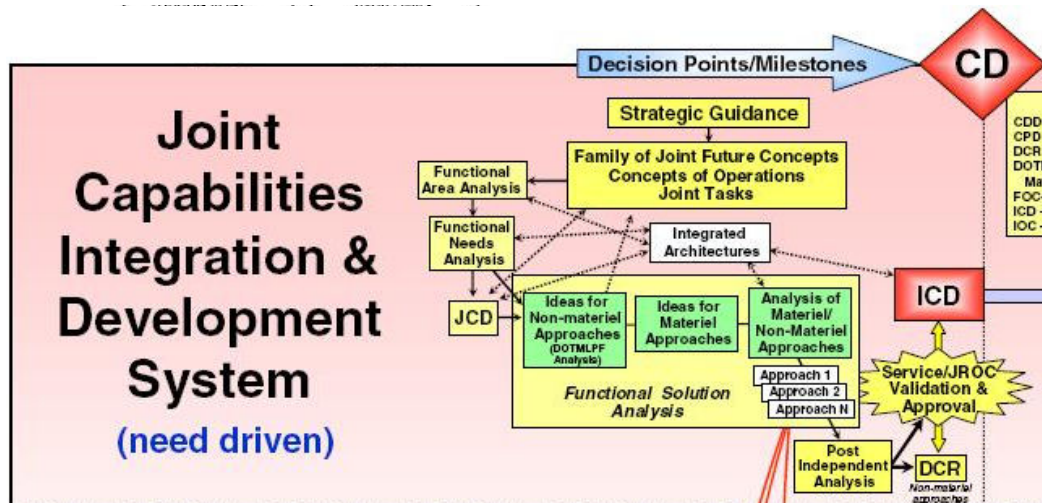


Figure 5: Early JCIDS Process [59]

The heart of JCIDS is something called a capabilities-based assessment (CBA); a four-step procedure that is used to define what the solution to a military need should be. The first step is called the functional area analysis (FAA). An FAA distills strategic guidance and military needs into a problem statement. The FAA defines what needs to be done as well as the standards to which it must be performed [21]. Once the FAA defines the problem, the next step is the functional needs analysis, or FNA. The purpose of the FNA is to establish how effective the current and planned military is at meeting the needs identified in the FAA [21]. On one hand, the FNA analyzes the current means for achieving a given effect to the standards prescribed in the FAA and on the other hand, the FNA searches for areas where joint concepts overlap with the capability at hand. This second portion of the FNA is one of the key differences in JCIDS compared to earlier requirements processes.

If the FNA shows that there is a capability gap that cannot be met by current or planned forces, then the CBA proceeds to the functional solution analysis (FSA). The

FSA is meant to identify possible non-materiel and materiel approaches for closing the capability gaps and then assess the effectiveness of these approaches with respect to the standards prescribed in the FAA [21]. The FSA produces a prioritized list of materiel and non-materiel approaches that meet the capability need. These results then go through a post independent analysis and are used to create either an initial capabilities document (ICD) or a DOTMLPF change recommendation (DCR).

2.6.2 Detailed Capability-Based Analysis Process

The CBA process described above is still somewhat vague. Keep in mind that the JCIDS has only been in existence for just over five years and detailed directions regarding its execution are varied between organizations. However, one of the most comprehensive explanations of the capability-based assessment is found in the Capabilities-Based Assessment User's Guide published by the Force Structure, Resources, and Assessments Directorate (JCS J-8). This document breaks down the FAA, FNA, and FSA elements of a CBA and provides clear guidance to facilitate CBA creation.

In order to execute the research being presented in this dissertation it is necessary to lay out the steps of the CBA process and discuss the information required and produced. This elaboration on the CBA along with a literature review of the current-state of the art for CBAs will lead to the technical gaps that this research will then attempt to close. Listed below is a summary of the steps in a CBA, adapted from the User's Guide.

1. Problem definition
2. Define scenarios
3. Define tasks
4. Define metrics
5. Select analysis method(s)/tool(s)

6. Reconcile analysis procedure
7. Perform baseline analysis
8. Identify extent of gaps
9. Identify non-materiel approaches
10. Evaluate non-materiel approaches
11. Identify materiel approaches
12. Evaluate materiel approaches
13. Present results, provide recommendations for further study [42]

In the above sequence, steps 1-4 fall under the FAA heading, steps 5-8 are part of the FNA and steps 9-13 are performed in the FSA. This process serves as the template for the method developed here.

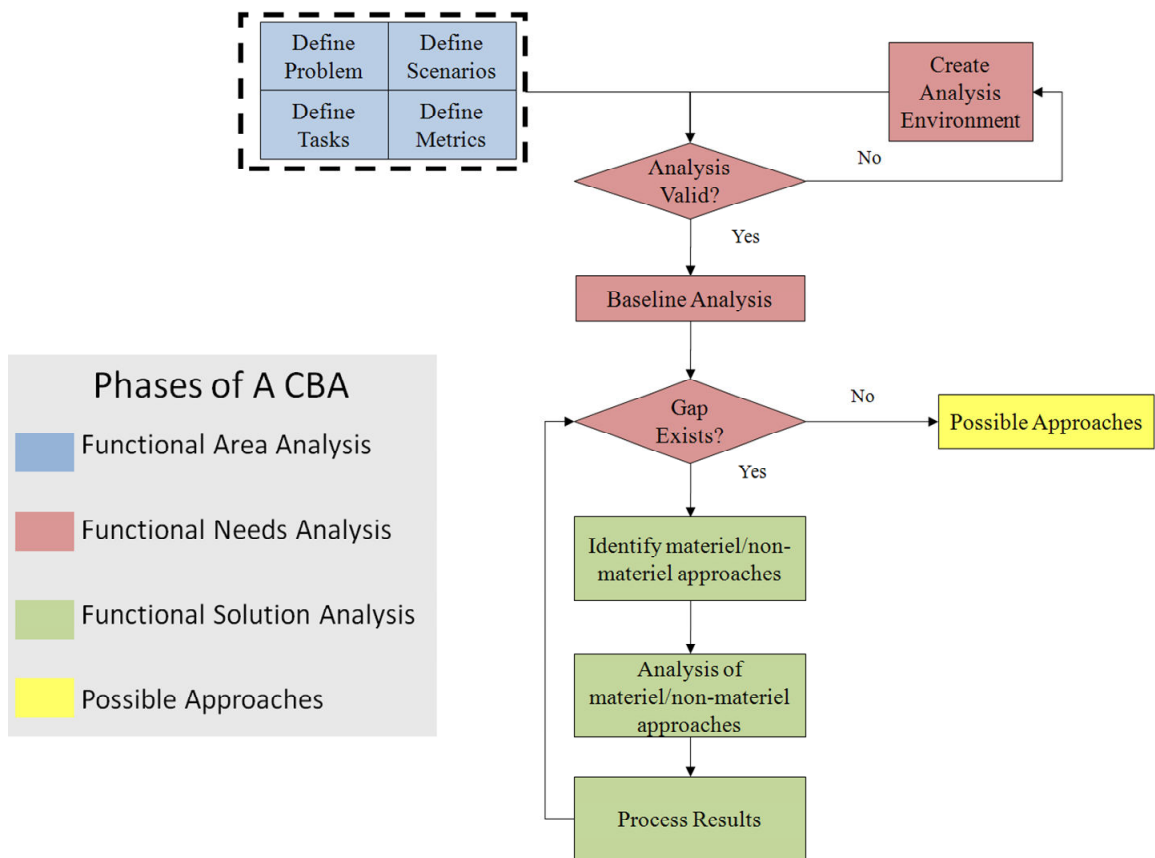


Figure 6: Methodology Diagram

2.7 Hypothesis #2

After examining the defense acquisition process and the CBA in particular, it seems clear that the CBA is a critical juncture that has a profound impact on the entire lifetime of an acquisition program. Given that Hypothesis 1 stated that doctrine will have a substantial impact on the design of a system, it seems reasonable to conclude that making doctrine a part of the analysis procedure will change how CBA's are performed. Accordingly, Hypothesis #2 is stated as:

The method presented will improve the quantification of doctrine, permitting a side-by-side comparison with materiel solutions. Furthermore, this method will be faster, more transparent, and more consistent than the current state-of-the-art in doctrine analysis. There will be dramatically different information fed between the CBA and other portions of the acquisition system.

2.8 Formulation of Methodology Technical Challenges

As was mentioned in the acquisition discussion, this process is still maturing and there are bound to be growing pains in the CBA process. First, consider a presentation given by the Air Force Studies and Analyses Agency regarding Capability-Based Planning. Listed within the conclusion of the report on CBA is a statement that it is “difficult to measure DOTLPPF solutions vs. M solutions – need techniques to help do this [130].” What this conclusion means is that when a DOTMLPPF analysis is done, there is a dramatic disconnect between the results for the materiel approaches and the doctrinal ones. Furthermore, according to the Office of Aerospace Studies which publishes a handbook for conducting an FSA, the preferred method for conducting a non-materiel analysis is subject matter expert (SME) survey [99]. Compared to more rigorous experimentation on materiel approaches which result in quantitative effectiveness and cost measures, non-materiel approaches are almost exclusively assessed qualitatively.

Additionally, a Military Operations Research Society (MORS) meeting on Capability-Based Planning concluded that “more model development is needed in representing perception, cognitive, decision, behavioral...concepts in a military context.” Based on these statements it can be asserted that *there is a need for quantitative analytical techniques to enable comparisons of doctrine approaches alongside materiel approaches.*

Another deficiency in the current CBA process can be found in the results of another MORS meeting, this time on Effects-Based Operations (EBO). This working group was comprised of experts in the fields of military operations, operations research, analysis, and various other associated disciplines. One of the conclusions by this group was that tools used in analyzing approaches need to be able to handle large solution spaces and not merely point solutions that come in ones and twos [49]. Doctrine is principally analyzed using SME opinion, prior live combat experience, training exercises, or wargaming. Clearly some of these are too dangerous or costly and time consuming to do comprehensive doctrine solution space investigations. Therefore, it can be stated that *there is a need for increasing the scope of the experimentation in the analysis of materiel/non-materiel approaches through increased analysis speed.*

Next, also listed as a conclusion of the Air Force Studies and Analysis Agency briefing was a statement emphasizing that even more difficult than evaluating a single capability is the task of comparing the *value* of multiple capabilities against each other [130]. A simple point, this assertion is loaded with meaning for the future of CBA. According to the MORS working group on CBP it is imperative to bring monetary

considerations into play at all points in the design and acquisition process. The JCIDS guidance requires an evaluation of cost for potential approaches [21] but if the effectiveness evaluation for non-materiel approaches is at a lower fidelity than materiel approaches, it stands to reason that the cost estimates are also insufficient. Therefore, it is possible to conclude that *it is necessary to incorporate cost analysis methods into the analysis of materiel/non-materiel approaches.*

Finally, the last technical challenge can be found by reexamining the two previous conclusions of the Air Force Studies and Analyses Agency. While one conclusion discusses materiel and non-materiel effectiveness and the other speaks to value comparison, the common thread is the notion that approaches need to be measured against each other [130]. This sentiment is echoed by Davis who asserts that a method for CBP needs to have a process for selecting between the ways and means in a capability and fiscal context [28]. Also, consider the second technical challenge regarding reduced analysis time. If the analysis is sped up and more approaches are evaluated, then the mechanism for comparison needs to be able to accommodate a sizeable set of approaches. Therefore, it can be asserted that *there is a need for a means to readily permit comparisons of large samples of approaches in the face of conflicting metrics in such a way as actionable information can be presented to the decision-maker.*

The above discussion illustrates several of the prominent documented challenges in the completion of a capability-based assessment. Each shortcoming affects one of the steps in the aforementioned CBA process and if the enablers for the process are deficient then the results will not be up to the standards demanded by the Defense Department and

ultimately the citizens of the United States. While addressing these gaps is by no means the panacea for defense acquisition, an incremental improvement in the ability of analysts and decision makers to effectively procure and design the correct solutions is an important step towards meeting the goals set forth by national leadership.

For convenience, the technical challenges identified above are summarized in the list below.

- a. There is a need for quantitative analytical techniques to enable comparisons of doctrine approaches alongside materiel approaches.
- b. There is a need for increasing the scope of the experimentation in the analysis of materiel/non-materiel approaches through increased analysis speed.
- c. It is necessary to incorporate cost analysis methods into the analysis of materiel/non-materiel approaches.
- d. There is a need for a means to readily permit comparisons of large samples of approaches in the face of conflicting metrics in such a way as actionable information can be presented to the decision-maker.

2.9 Methodology Research Questions

The technical challenges set the stage for the research outlined in this dissertation. If these gaps are closed, then substantial progress will have been made towards improving the conduct of CBA's. From these four challenges, research questions can be formed that will guide a comprehensive literature search. This literature search then allows for a selection of enablers that address the technical challenges, close the gaps, and make possible an improved CBA process. These methodology research questions

support the first overall research question regarding creation of a method. The new expanded research question #3 is presented hierarchically below:

3. How can a method be developed that rigorously compares doctrinal approaches to materiel approaches within the context of a capability-based assessment?
 - a. How can doctrinal variables be quantitatively assessed in such a way that they are comparable to materiel variables?
 - b. How can the exploration of a doctrine-materiel design space be accelerated to enable quantitative analysis of a broader space of possible approaches?
 - c. How can the relative cost difference between different approaches be captured and compared?
 - d. How can the expansive design space be best parsed and analyzed so that better information can be output from the capability-based assessment?

CHAPTER III

METHODOLOGY FORMULATION

This chapter is designed to address each of the methodology research questions in turn in order to develop hypotheses that can be tested using experimentation. Within each hypothesis formulation there is a literature review of relevant materiel as well as independent analysis of how to incorporate pieces into the overall method. During the course of this formulation, additional research questions may arise to help address the first four listed above. At the conclusion of this chapter, the hypotheses will be summarized and the research method presented.

3.1 *How to Quantitatively Evaluate Doctrine*

Given that there is not a thorough enough exploration of non-materiel approaches, the first issue to address before the research question can be answered is: what are the methods by which doctrine changes today? The first method by which new doctrine is formed comes through real combat experience. There is no more convincing way to establish the validity of a new doctrinal concept than to have successfully implemented it on the battlefield. Obviously, conducting actual battles as a way to evaluate potential doctrines is an unethical approach and is not an option.

The next most realistic arena to test doctrine besides actual combat is artificial combat. Training exercises are routinely conducted within the military and every effort is made to make conditions as realistic as possible. Such an arena would be ideal to

ethically test new concepts with as high a resolution as possible. However, the time and resources required to assemble combat elements is enormous. If these troops were then being used to test new ideas rather than hone their skills, there is an additional cost in terms of military effectiveness lost. Therefore, this option is not possible.

Third, and perhaps most widely used, is the work of subject matter experts (SMEs). The idea here is that if the people who have real combat experience and knowledge of military systems can be surveyed, their answers would represent the best assessment of possible approaches. There is a great deal of validity to this method in that the experiences of real people in real situations are invaluable. However, the use of SMEs is largely qualitative and does not provide a basis means for comparison to more detailed materiel analysis. Therefore, this option is out and a more quantitative option that does not put people in harm's way and is reasonable in cost is needed.

3.1.1 Wargaming

In addition the techniques above, another tool that is commonly used to experiment with changes in doctrine is the wargame. According to Air Force Instruction 10-2305, a wargame is “a simulation, by whatever means, of a military operation involving two or more opposing forces using rules, data and procedures design to depict an actual or assumed live situation [143].” Essentially what this says is that conducting real operations is obviously out of the question, so the military needs to conduct wargames to gain insight into a potential present or future state. The AFI goes on to state that, in general, the purpose of a wargame is to examine friendly and enemy concepts, experiment with new ideas, and support the development of long-term military vision planning.

While the preceding definition is extremely broad, there is one particular class of wargames which is particularly relevant to how doctrine is analyzed today. The class of wargames used to assess doctrine is virtual, interactive wargames. These games bring together subject matter experts from a variety of backgrounds and leverage the expert knowledge to assess potential strategies or future states.

To get a better feeling for what actually constitutes a wargame consider several examples chosen from hundreds of possibilities to provide a sampling of the extent of wargames. First, consider one of the largest wargames in the U.S.; the Global Wargame Series. Started in the late 1970's, the purpose of these games was to investigate broad strategic changes and new ideas that would not be possible to explore otherwise [46]. By simulating possible future scenarios, it was hoped that better plans could be made and military planners could be better prepared. It is worth noting that this series is classified as a 'research game' as opposed to an 'educational game'. Educational games are for training purposes while research games are meant for concept exploration.

The first series of Global Wargame was run from 1979 through 1983. It simulated various crises throughout the world and each year the scope of the effort grew to involve more and more players from the military. The second Global Wargame Series took place every summer for three weeks at the Naval War College between 1984 and 1988. At any one time, over 600 people were actively involved in the game. The participants came from the United States, the United Kingdom, and Canada. The players represented a broad spectrum of agencies including everything from the Department of Defense, to the Department of State, to the Federal Emergency Management Agency, to organizations less associated with defense planning such as the Bureau of Mines [46].

Each successive year in the second Global Wargame Series built upon the results of the previous year. In 1984, the goal was to establish that the gaming environment was realistic and that a large game with a large number of players could be run. After that, scenarios involving protracted conventional war between the US/NATO forces and the USSR/Warsaw Pact forces took place. The objectives of this study included: examining political and economic factors in wartime, the impact of strategic changes on mission success, the extent to which logistic plans were sufficient, and many others [46]. The results of this game were critical to military understanding of the possibilities in a protracted war situation as well as identifying the potential of future concepts.

Global Wargame did not end with the series in the 1980's. Global Wargame 2000 was used to study the effects of network-centric warfare. Participants were not only located at the Naval War College but also through the use of a distributed environment players were located at remote facilities including onboard the USS Coronado, an amphibious transport ship. The goal of this series was to put participants in different situations and see how novel technologies and techniques were employed in a network-centric environment. The effectiveness of the new concepts was recorded quantitatively through empirical results and qualitatively through questionnaires provided to the participants.

There are a large number of other wargames being conducted every year. In 2001 the Army's Transformational Wargame 'Vigilant Warriors' was held at the Army War College to investigate how a future Army force would perform under different scenarios in the Persian Gulf. Like the Global Wargame, Vigilant Warriors is a series that executes each year to build information about strategies and technologies. In June 2005 the

National Defense Industrial Association (NDIA) and the Marine Corps conducted a wargame to examine distributed operations [98]. Wargames are conducted by all corners of the defense world, looking at all types of problems and leveraging the expertise of a wide array of experts.

Thus far, it has been shown how wargames operate and the kinds of insights they can provide. However, there are some additional points about wargames to consider. First, the traditional output of a wargame predominately is made up of qualitative statements of performance and issues for further study. For example, in the NDIA distributed operations wargame, statements such as “needs further analysis and experimental verification” or “Are we getting our money’s worth from [distributed operations?]” appear in the results. It also lists infiltration tactics, distributed operations in urban environments, and operational tempos as items for further study [98]. Additionally, the Army Vigilant Warriors report states that it is not possible to tackle every aspect of a military operation with a single wargame because of the complex nature of the problems at hand [98].

Another drawback to wargames is the magnitude of resources required to complete one. While wargames offer a cheaper, faster, and most importantly, safer method than actual combat for analyzing potential concepts, it is still necessary to assemble large numbers of people for long periods of time. The Global Wargame took three weeks every year and required the time commitment of senior people in many agencies [46]. If these people are expert enough to be called as authorities in wargames, they cannot spend all their time in simulation as they are needed in the real world. Furthermore, consider that the scheduling takes place every year or every few years. In

between that time radical changes can be occurring in terms of everything from technologies to political realities for both friendly and enemy forces. Since so many resources are needed to complete a wargame, it is not possible to examine a large number of possibilities rapidly.

Also, consider that at the heart of wargames are people performing actions in a virtual simulation. This means that the results of a wargame are somewhat dependent on the players involved. Biases or experiences could warp study results. Combined with the resource limitations on the number of cases that can be run, this means that a single wargame likely cannot provide the final solution to a problem. Wargames should be considered a single data point as part of an overall trend. However, if wargames cannot provide enough information to definitively and objectively answer study questions, then they definitely cannot be used as a design tool to perform the large number of trades and iterative procedures that accompany an acquisition decision. Therefore, while wargames provide invaluable insight and a powerful realistic simulation capability, they cannot be the only approach relied upon by the Defense Department. Accordingly, wargames do not fit the requirements for this study and another approach is required.

3.1.2 Discussion on Modeling and Simulation

Another option for evaluating doctrinal concepts is modeling and simulation. Modeling and simulation is a phrase that appears often in many engineering communities including aircraft design and defense acquisition. However, before establishing if and to what degree modeling and simulation will play a role in this research it is important to understand fundamental concepts and definitions that govern the field of modeling and simulation (M&S). First, the term model is defined by the Department of Defense as a

“physical, mathematical, or otherwise logical representation of a system, entity, phenomenon, or process [102].” While the exact wording of definitions for the term model can vary, this is an accurate description that applies to a wide range of fields and is accepted by many. Reinforcing this definition is that given by Papadimitriou et al that a model is “an abstract representation of a system, usually containing structural, logical or mathematical relationships, which describe a system in terms of state, entities, and their attributes 0.” Furthermore, a state is all the variables needed to totally define the system at any given time. An entity is defined as anything that needs to be directly characterized in the environment and the elements that describe an entity are called attributes. Finally, an event is anything that modifies a part of the system state [102][70].

Now that ‘model’ and its accompanying terms have been defined, the next major term to investigate is simulation. The DoD defines a simulation as the application or execution of a model over a time period [72]. A simple definition, but as it turns out, simulation is a vast field that contains many options for an analyst. Law decomposes a system and the means to study it so a relation can be drawn between simulation and the accompanying system. An adaptation of Law’s diagram is shown below in Figure 7 [84].

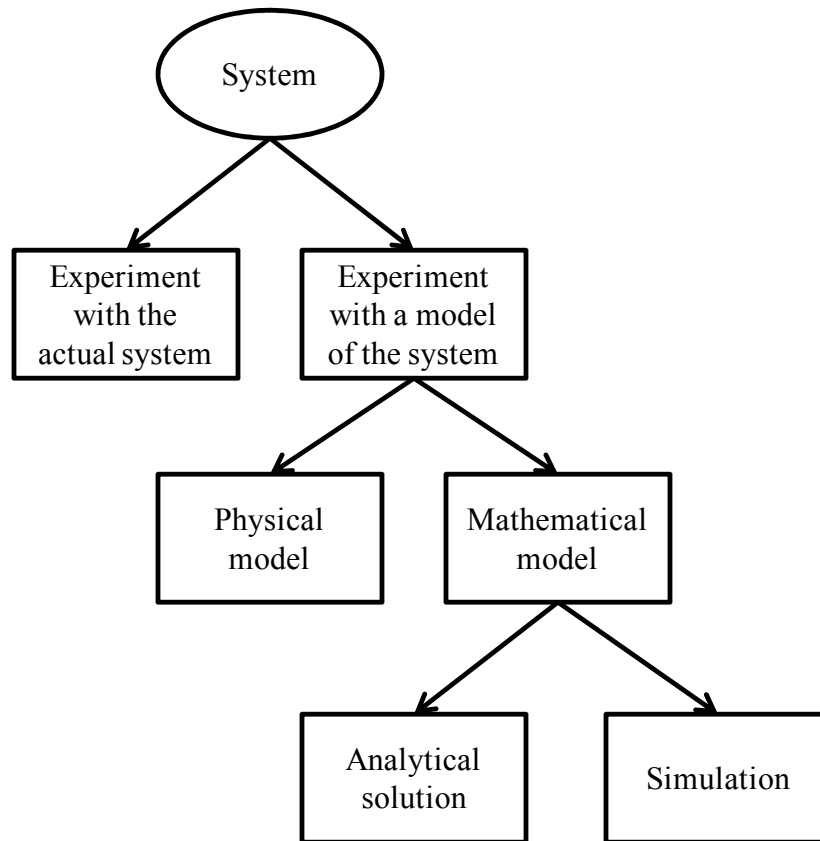


Figure 7: Decomposition of Experimentation Types

Law contends that in order to investigate a system it is possible either to perform experiments with that system itself or create a representation of that system (i.e. a model). While it may be preferable to experiment with the actual system, it is not hard to deduce that the cost of such a plan could become prohibitive and in the case of a conceptual design study such as this one, the actual system does not even exist yet. Given that the choice has been made to experiment with a model of the system, the next choice is whether to pursue a physical (also known as iconic) model or to create a mathematical model. Here mathematical means either logical or numerical relations. The problem with physical models is that they can also be too expensive and, similar to the argument against experimentation with the actual system, in a conceptual design where the physical

system is not yet determined an inordinately large number of models would have to be created. Now given that a mathematical representation is preferred, the next question is should our model be an analytical representation or is simulation required. An analytical solution is something that can be framed in a closed form equation that can then be easily computed. For many systems, no closed form solution exists and even if one does exist, the computation resources needed to compute it are extreme and prohibitive. For the capability analysis portion of this study, no closed form solution exists. Therefore, given Law's decomposition of system evaluation and the fact that this research centers on design space exploration of a conceptual system not yet in existence, it can be proven that other options besides simulation are impossible given the time and monetary constraints imposed here.

Now that modeling and simulation have been defined and it has further been argued that simulation is essential to the research of this dissertation, the next step is to address the manner of simulation required. With respect to interaction or interface there are three types of simulation: live, virtual, and constructive. These types describe how the user or users interacts with the simulator. A live simulation is one in which the entities in the simulation are physically present such as a live fire exercise at a military base. A virtual simulation is one where people interact with the simulation. These are also called "human-in-the-loop" simulations. An example of a virtual simulation would be a flight simulator for pilots. Finally, a constructive simulation is designed to have no interaction with a human during the execution of the model. An example of a constructive simulation is an aircraft design code that takes inputs from the user and then executes an aircraft design without any further input from the user.

Based on this brief description, it should be clear that live and virtual simulations, which require human interface, are almost guaranteed to run in real-time. Additionally, for a live experiment, the equipment available is the only equipment that can be used, precluding a design space exploration for new concepts. There are also limits on how much design space can be explored in a virtual simulation due to the limits of human endurance. The constructive simulation does not suffer from these shortcomings. While it is possible to have a constructive simulation that runs slower than real-time, no such simulation will be considered for this study and in fact there is a wide selection of simulators that execute much faster than real time. With no human constraints, a constructive simulation can be run many times allowing a more thorough exploration of the design space.

Recall that one of the motivating challenges for this research is that the current methods of analyzing doctrine do not explore enough designs and are not appropriate at the conceptual design stage. The obvious drawback to a constructive simulation is that it is likely more simplified than the other simulation types. However, for a conceptual design study, the fidelity required is such that a constructive simulation can be viable. More details will be provided later about model construction and verification. For now, based on the reasons of reduced run time, reduced cost, and the potential for greatly increasing the design space covered it is clear that a constructive simulation is what is needed for this study.

The next choice about the simulation type is in regards to a static versus a dynamic simulation. Basically, a static simulation does not account for time changes; it can be thought of as a snapshot. A dynamic simulation, which accounts for time changes,

is the clear choice for the mission effectiveness portion of this research project. Next, consider if the simulator should be deterministic or stochastic. The term stochastic basically means that there is randomness in the simulation. A deterministic simulation is the opposite, that is to say for a given set of inputs, a deterministic simulation will always provide the same output values [84]. For the mission effectiveness piece, it is proposed that many of the mission and environment parameters such as enemy dwell time, weapon kill probability, and so forth will be random. Therefore, the mission effectiveness simulation will need to be stochastic.

3.1.3 Selecting an Effectiveness Simulation Environment

3.1.3.1 Criteria for Selection

Before an appropriate simulation environment can be chosen, it is first necessary to establish the guidelines for selecting the environment. As was already stated, it is desired to have a constructive, dynamic, and stochastic simulation. Therefore, all of the options presented here will have all of those qualities. Accordingly, there must be some additional properties of the simulation that are desirable which can differentiate between candidate environments.

First, the simulator must be accessible. This sounds like an obvious notion but if it is going to take a lot of effort to acquire the simulator then that particular option may not be worth pursuing unless there is a particular feature that justifies it. The next criterion is whether or not the simulator can effectively handle agent-based modeling. It was hypothesized that agent-based modeling can be used to assess the impact of doctrine therefore the environment must be capable of supporting agent-based modeling. Third, it is important to consider what level of effort is needed to obtain the relevant results. Some

simulations might have a specific programming language that requires a great deal of effort to learn, for example, and therefore are not as desirable as a simulator that produces similar results with less effort. This is an important criterion because this research effort has a limited scope and timeframe as well as being primarily about methods development so creating the perfect simulation is not necessary. Thusly, the simulator selection will proceed with three criteria that boil down to: can the simulator be obtained and if so, can it produce the necessary data with a reasonable level of effort?

3.1.3.2 Candidate Environments

There were seven simulation environments that were considered for this study: Arena, FLAMES, MATLAB, Netlogo, OPUS, SEAS, and STORM. The selection of these seven was conducted through a literature review of simulators that might fit this problem as well as be readily available for the research. The following paragraphs will discuss the strengths and weaknesses of each environment as well as address the above criteria.

First, Arena is a software package from Rockwell Automation that was developed to study logistics and processes primarily for manufacturing, supply chain, and service systems. Generally, the Arena package is employed as “an enterprise business analysis and productivity tool.” A classic example is a warehouse that sells to customers while balancing supply levels. Arena would allow the user to change the cost, the rate of customers, and so forth, and determine overall business effectiveness. While on the surface this seems radically different than a military effectiveness simulation, a hunter-killer mission could be modeled as a process. Arena contains a graphic interface that allows rapid construction of models but also gives the user access to the SIMAN

programming language allowing for a great deal of customization [80]. However, referring again to the above criteria, working against Arena is the level of effort required to construct such a simulation. This program is not set up to allow agent-based modeling and trying to put behaviors into the program would require a significant investment of resources. Finally, while this simulator is accessible, there is a caveat. The readily available version is the student edition which has limits on the number of entities and number of steps in a given process. Therefore, even if it were possible to construct the agent-based effectiveness simulation, there may not be enough pieces to fully assemble the game board.

Next, FLExible Analysis and Mission Effectiveness System or FLAMES is a simulation framework developed by Ternion Corporation. The key thing to note about the above statement is the word 'framework'. FLAMES is not a simulation environment in and of itself, but rather provides the flexibility and connectivity necessary for a user to develop and execute their own simulation environment [127]. This program is used widely in the military and industry and has even been used in academia. It is available to the author through Georgia Tech's Aerospace Systems Design Lab. This tool is capable of modeling agents and their behaviors and interactions with other systems. Furthermore, FLAMES can be run parametrically in batch mode allowing for a thorough exploration of the doctrine and materiel space. Even though FLAMES can provide all of the functional capability needed for this work, the key remaining issue is effort required. Creating the pieces to go into the framework must be done in the C programming language. However, there are example models provided with this package that could aid in the creation of the

simulation for this study. Therefore, for now, FLAMES remains an effective and potential option.

MATLAB is a technical computing language created by The MathWorks Corporation. It is not a simulation environment or a framework but rather a programming language similar to C or FORTRAN [128]. What makes MATLAB unique is that it is specifically geared to mathematical tasks that can be laborious in other general programming languages. MATLAB contains the means for readily handling differential equations, matrices, and many other mathematical features. This program is readily available to the author so it meets the first criterion. However, the language is not designed to immediately accommodate agent-based models. An additional item to note is that, in cooperation with the Air Force Research Lab and the Hunter/Killer team at the Aerospace Systems Design Lab, a conditional probability simulation has been developed in MATLAB to model the hunter killer problem. While using this program would greatly reduce the level of effort required, the construction of the model precludes the use of agent-based models. The abstraction type for this model construction is aggregation and that eliminates the individual behaviors that allow agent-based simulations to work.

Next, NetLogo is a modeling environment developed at the Center for Connected Learning and Computer-Based Modeling at Northwestern University [159]. There are several attractive features of this program including the fact that it is developed specifically to accommodate agent-based modeling. The program was designed to model complex systems and therefore it should have the capabilities needed to capture doctrine changes. While the tool has been used predominately in the social sciences, the attributes of the program lend themselves to adaptation to different fields. One of the problems with

using this program for this study is that it does not have inherent aircraft, communication, or doctrine models; everything will have to be created from scratch. However, there are sample models and comprehensive documentation so required level of effort should be manageable. Finally, in regards to the accessibility of the program, NetLogo is available freely from the developer's website.

The next program being considered was developed by Operations Research Concepts Applied and is called ORCA Planning and Utility System (OPUS) [105]. This program was designed specifically for military aircraft route planning. OPUS creates paths to avoid terrain and threats, reach targets, and can be used for single aircraft or for larger fleets. A discrete event simulator executes missions and calculates metrics for various scenarios. The key feature of this program is the dynamic mission re-planning which is of current interest to military UAV planners. It appears that OPUS can accommodate, in some fashion, the behaviors of agents. However, this program is not immediately available and therefore it is also difficult to assess the level of effort required to incorporate doctrinal variables into the simulation.

Next in the list of programs is the System Effectiveness Analysis Simulation (SEAS) which is tool owned by the government and used specifically by the U.S. Air Force [1]. The first thing to note about this program is that it is billed as an agent-based simulation and can handle many agents. Since it is an Air Force tool designed to enable rapid warfighting trades it possesses many organic capabilities that are needed for this research. The question of availability is an important one for this program. It is available to users who demonstrate a need and have an Air Force sponsor to grant permission. Given that this work stems from research being performed for the Air Force, obtaining

SEAS is feasible. As for the level of effort required, SEAS has its own language that, while well documented, will nonetheless take effort to master. It does come with example scenarios which can be expanded upon for creating the necessary studies of this research.

Finally, the last program being considered is Synthetic Theater Operations Research Model (STORM). STORM was developed by the Air Force and specializes in campaign simulation with particular focus on joint operations. This program is being considered because the group responsible for its development, the Air Force Studies and Analyses Agency, provided some of the motivation for this study and therefore this tool potentially could capture the desired effects. Furthermore, like SEAS, using a certified Air Force tool is likely to add credibility to the research effort here. However, this tool was not accessible and therefore was not a contender for the final simulation environment.

Now that each simulator has been examined, the next step is to select which of them is best suited for this work. For a summary of the attributes for each simulator, please refer to Table 2 below.

Table 2: Summary of Simulator Attributes

	Accessibility	Doctrine and Agents	Level of Effort
Arena	Fair	Poor	Poor
FLAMES	Good	Good	Poor
MATLAB	Good	Poor	Good
NetLogo	Good	Good	Fair
OPUS	Poor	N/A	N/A
SEAS	Fair	Good	Fair
STORM	Poor	N/A	N/A

This discussion starts with the assumption that each simulator offers some level of potential for this study. Before getting to the specific feasibility of a given program, it is first worth investigating which are obtainable. The OPUS and STORM programs received poor marks for accessibility because neither is immediately available. That is not to say they couldn't be procured. However, given the limited scope of this study, a program would have to demonstrate greatly superior performance to justify purchasing it. Since the other programs are assumed for now to be potential candidates, OPUS and STORM are not investigated further. This is the reason for the "not applicable" ratings in the other two attributes.

Of the remaining programs, all are currently available to this study. However, it is worth noting that Arena's availability suffers in that the version available is only the student version with the limitations noted above. Since each of these five programs is accessible, each of them was evaluated with respect to how well they could capture doctrine and agent-based behavior. While the actual ability of a given program cannot be fully determined until this study is complete, it became clear that some programs were well suited for this study and some were not. Arena and MATLAB, particularly the conditional probability model mentioned above, do not readily lend themselves to a study of doctrine and materiel solutions. FLAMES, NetLogo, and SEAS all show the capacity to capture doctrine effects and incorporate agent-based modeling.

The final attribute to consider is the level of effort required to achieve the results of this study. It is worth noting that this criterion is partially objective and partially subjective. That is to say, this attribute captures the intricacy of the program, the author's

expertise with a given program, and how many steps would have to be performed to successfully implement the experiments in this study. This is important. It must be clear that no single of these programs is the ‘end-all-be-all’ of simulation and the level of effort criterion is very specific to this investigation.

With regards to each program’s score in this category, the program that scored the best in level of effort was MATLAB due to its simple interface, comprehensive documentation, and researcher’s personal experience. However, since it does not readily provide the functionality that is needed, the level of effort is somewhat irrelevant. Of the three programs which were deemed accessible and appropriate, FLAMES received a ‘poor’ rating while SEAS and NetLogo received a ‘fair’ rating. All three have comprehensive documentation and all three are supplied with sample models to build off of. All three programs have been investigated first-hand by the author so the decision for their score is not arbitrary. The key element that gave SEAS and NetLogo a higher rating than FLAMES is that FLAMES is a simulation framework. FLAMES is more of a blank slate with the capability to connect modules once the user defines the connections and therefore in order to get the same functionality as in the other two programs, more actions are needed by the user. Again, this is not to say that FLAMES is never the appropriate simulation even when compared against these other programs. For a different problem, or a more experienced user, the result could turn out different.

Now the selection is down to NetLogo and SEAS. Both have the same approximate level of effort and seem to be well suited for the research at hand. SEAS scored slightly lower in accessibility but given that the author has access, that lower score speaks to the fact that it is not readily available to just everyone. As it turns out, this will

be the deciding factor. This investigation is an academic research project that is meant to advance the state of the art in doctrine analysis during early conceptual design. Theoretically, the more that the academic, industrial, and government communities can access and duplicate the results of this work, the more valuable it is. While SEAS provides some benefits in that it is a government-owned tool developed specifically for the Air Force, the anticipated ability of SEAS to answer the research questions is not appreciably higher than the more widely available NetLogo. Therefore, this research will proceed with NetLogo as the selected simulation environment.

3.1.4 Review of Studies in Doctrine and Capabilities

Previously, several capability-based acquisition critique papers presented highlights of some of the deficiencies with the execution of CBAs. These reports are a good starting point but a more comprehensive first-hand look at research studies needs to be performed. This research will help establish recent ideas that have been applied to similar challenges. This section will address what worked well, what did not work well, what hasn't been attempted, and why previous results turned out as they did. The following literature review is a collection of studies both for doctrine and capability-based analyses in general.

First, a study by Mulgund et al used genetic algorithms to optimize air-to-air combat tactics [97]. This research examined team formations and intercept geometry and tried to build large scale tactics through the optimization of small units. This was mostly a study to look at the viability of using GA's to build up combat tactics of larger forces and the work was based on a certain aircraft formation. It was decided to keep this formation as the basis and "not deviate from established tactical doctrine." For that study,

this is an adequate assumption but the research of this paper is meant to be exploratory. Additionally, there was no assessment of variable aircraft capabilities and there was no assessment of cost.

There are some doctrine studies that employ agent-based models. One example is Woodaman who examined tactics for military operations other than war (MOOTW) security. In this study, the agent-based model was implemented in a discrete-event simulator. The research examined how peacekeepers could protect an asset from rioting individuals [161]. In the study, only two tactics were examined by the author; the tactic variables were not parametric. So while it does address a technique for analyzing tactics, it does not illustrate how to expand the solution space beyond what is currently done with wargames. Furthermore, there was no cost or materiel analysis in this study which has been identified as a need to support acquisition decisions.

Another agent-based study was performed by Pawloski who looked at land tactical level combat. In particular, this study focuses on how the organization of units affects combat outcomes so it was a combination of doctrine and organization that was analyzed [108]. While this is a non-materiel analysis, it is not parametric with respect to potential doctrine. Furthermore, it does not assess cost and does not assess materiel solution. However, the study did result in agents within a computer simulation achieving behaviors that closely mimicked that of real infantry troops.

An investigation that looked more closely into the incorporation of tactics into conceptual design was performed by Frits. In this research, the process of torpedo design was studied with emphasis placed on how tactics affected design [43]. The tactical variables chosen were the distance from target when launched and the area of uncertainty

around the target. It focused on the searching tactics of a torpedo and did not account for submarine tactics. Frits also performed a mine-hunting study as another tactics investigation. Here, spacing of the search legs and search velocity were the tactical variables. This work demonstrated the importance of tactics within conceptual design. However, there are several items that were identified as areas for future work within this thesis. First, Frits noted the importance of engagement models and how to truly capture tactics in design it is necessary to have parametric engagement analyses which remove the human from the loop. Also, the work done in this thesis focused on one level of a system-of-systems. In reality, a weapon is part of a platform which is part of a larger force. Examining these different levels is important. Finally, as has been noted in this dissertation, doctrine is more than just the tactics of a platform. Doctrine also applies to operations and strategy.

Frits' work on tactics in design began to touch on the important element of systems-of-systems. Extending this concept, it is worth examining various studies that have looked at systems-of-systems and capabilities. One such study was performed by Ender who was investigating the design of an air-defense weapon in a system-of-system context [39]. In this investigation it was shown how top level requirements can be related to lower level systems. This research had some rudimentary system cost analysis as well. One of the major contributions of this study was the data visualization capability to illustrate the system-of-systems level trades.

Another system-of-systems research effort was conducted by Biltgen who created a method to perform quantitative technology assessment in a system-of-systems capability-based context [11]. This study illustrated how technology level changes can

impact overall system-of-systems effectiveness. An excellent framework for a capability-based assessment is provided however as was noted by the author, there is work to be done accounting for tactical changes and life-cycle cost.

3.1.4.1 Conclusions from Previous Studies

While much more attention is traditionally given to materiel analyses, there does exist in the literature very good research efforts for non-materiel problems. Furthermore, while the notion of systems-of-systems and designing for capabilities are both relatively new concepts, work is being done to address the challenges. What is clear though is that for these types of large, complex problems, no one yet has all the answers. The main challenge is quantification of doctrine. Additionally, much remains to be done in the area of life-cycle cost assessment. Also, there does not yet exist a study where doctrine and materiel variables are traded off in a broad yet transparent conceptual design space. In particular to doctrine variables, most of the studies thus far focus exclusively on tactics and do not consider operational changes as well. There is very limited evidence that techniques for expediting analyses have been fully exploited in the case of doctrinal changes. The few research papers presented here are fairly typical of the state-of-the-art. Finally, there is good agreement between the studies documented here and the conclusions of the assorted critique papers on capability-based analysis, providing additional support to the technical challenges and research questions developed for this dissertation.

3.1.5 Review of Doctrine Analysis Techniques

It has been stated in the technical challenges that it is necessary to find a method for quantifying doctrine alongside of materiel approaches. Already it has been shown that

constructive modeling and simulation is needed to quickly quantify doctrine changes. Furthermore, the simulation environment must be capable of handling both materiel and non-materiel changes to allow for the most straightforward comparison between radically different approaches. Therefore, this section is meant to address the modified research question: How can doctrinal approaches be quantitatively assessed in a constructive modeling and simulation environment?

The following discussion is a brief description of several analysis techniques that have either been employed or suggested as strong candidates for this class of problems. A MORS symposium on effects-based operations identified many of these methods while other came from scouring the literature on doctrine and non-materiel assessment. Each technique will be briefly explained and then once all have been presented, a final technique will be selected for use in this research. The techniques are presented in alphabetical order. The properties of a method that were investigated included where it has been used before, if it can capture the complex nature of a non-materiel and materiel design space, and if it can provide the type of quantitative information desired here.

The first technique to consider is agent-based modeling (ABM). ABM looks at each entity in a simulation as an “agent” which is a self-contained, goal-seeking entity which interacts with its environment. By constructing a discrete-event simulation with agent entities, it is possible to simulate complex behaviors. What makes an agent-based technique different than a more traditional simulation is rather than providing the entities with specific instructions over the whole timeline, the agent has simple instructions or guidelines that it applies depending on the situation. Agent-based models are very good for complex adaptive systems that exhibit a great deal of non-linearity. Furthermore, they

can be used to identify emergent behaviors. An emergent behavior is the aggregate behavior of a large group that does not logically or uniquely follow from the lower level behaviors.

Next, conditional probability is a statistical method which tries to determine the probability of event B occurring given that event A has already occurred. This type of model could be used to decompose the mission of an aircraft. For example, the probability that an aircraft destroys its target is the conditional probability that it finds its target, then identifies its target, then fires its weapon, and then that the weapon impacts the target and so forth. The positive aspect of such a model is that it quick to implement. However, conditional probability aggregates the assets together and thus eliminates the capability to identify emergent behavior. This could possibly limit the amount of doctrinal insight to be gained from this method.

Next, influence networks are a technique used in the social science to illustrate how members of a group or population are influenced by other members. This process involves collecting weighted averages of views from the members and using that to determine the next state of the views of a member of interest. This type of analysis has been used to model how nations make decisions or how terrorists can be deterred. While this technique does account for human factors and behaviors, it does not do so in a way that is conducive to doctrine analysis. This technique is better suited for, once doctrine and materiel approaches have achieved a capability, measuring how that capability then in turn affects nation will.

Petri nets are another analytical technique to consider. Petri nets are a valuable technique for modeling systems with concurrent processes. They are well suited to

dynamic and non-linear systems. Petri nets also deal with the states and transitions of entities. Some of the areas where Petri nets have been successfully employed include reliability, software and information systems, and data or product flow. The primary concern for the use of Petri nets is the relative effort required to adequately model the systems of interest for this study. Additionally, while Petri nets have a graphical depiction of the flows of elements within a system, the graphics do not show an aircraft flying through an airspace.

Finally, the last doctrine analysis technique being considered is system dynamics. This technique represents elements in the system as stocks and flows. The transitions between the states are represented by equations. System dynamics is well suited for non-linear problems and can also capture emergent behavior. Furthermore, this method captures very well the effects of feedback on a system. If there are two states and a transition that captures the flow from state one to state two, feedback means that the conditions in state two influence the transition between one and two.

At this point, it is fairly straight forward to eliminate influence networks from consideration. This technique applies more to quantifying the impact of military effects on other factors as opposed to evaluating the military effect itself. Another technique that can be eliminated at this point is conditional probability. While this is a method capable of quantifying some doctrine changes, the method of abstraction prevents the generation of emergent behavior and also requires more extensive modification of the models between analyses.

That leaves only agent-based modeling, Petri nets, and system dynamics as options. All three of these analysis techniques handle non-linear problems and could

potentially quantitatively capture doctrinal changes in a military system-of-systems. However, just because each could model doctrine does not mean that the three methods are equivalent in terms of effort and applicability. There are concerns about the complexity of a Petri net model to capture the necessary behavior. Also, both Petri nets and system dynamics require closed form equations to represent the transfers between different states, so called flow rates. In a system such as the one under study, finding those equations may prove to be prohibitively time consuming with most of the effort going into developing the equations rather than addressing the research issues.

Finally, the deciding factor may be in how each method represents doctrine. Note that in the description for agent-based modeling the agents are given guidelines for behaviors. Furthermore, recall that in the definition of doctrine, it was stated the doctrine is authoritative guidance that requires judgment in application. Agent-based modeling most closely mimics how doctrine is applied in real life. Additionally, if agent-based models approximate entities in a military simulation, then by examining individual behaviors or events within a simulation, it could be possible to assess the accuracy of the models. Therefore, due to the way that agent-based models capture doctrinal variables, that technique is chosen as the one to be employed in this research.

Hypothesis 3a: Agent-based models, incorporated into a constructive simulation, will permit the quantitative evaluation of doctrinal variables.

3.1.6 Further Discussion on Agent-Based Modeling

According to the Department of Defense Modeling and Simulation Master Plan, modeling is simply the creation of a representation of another object or event. Likewise,

simulation is executing of one or more models over a period of time [102]. So where does agent-based modeling (ABM) fit into the broader field of modeling and simulation and where is it used today? In order to answer these questions, it is first important to understand what an agent is. Macal and North [88] identify several characteristics that define what constitutes an agent. First, to qualify as an agent, the object must be “self-defined”. That is to say, the agent must be discrete and identifiable from other objects around it. There should be a very clear set of properties that belong to this agent. Additionally, an agent must have some sort of objective that it seeks to attain. As unconventional as it may sound in the context of modeling and simulation, the agent should have goals and desires. This requirement ties into the next aspect of agents which is that any agent must be capable of autonomously performing actions within its environment that will aim to satisfy the aforementioned objectives.

There are other characteristics that could be used to define an agent and in fact, given that the field of ABM is relatively new, agreement is not universal. However, these three requirements (namely identifiable, goal-seeking, and autonomous) are generally accepted and for the purposes of this research will be used to define an agent.

There are other characteristics that could be used to define an agent and in fact, given that the field of ABM is relatively new, agreement is not universal. Cares et al state that agents are autonomous and make their own decisions according to guidelines [18]. Obaidat and Papadimitriou see agents as many individuals that can interact with each other [106]. Kewley and Larimer emphasize the ability of agents to lead to emergent behavior [81]. They claim an agent-based model is one where the overall performance of a group of agents is vastly more important than any single agent.

While there is no single definition that is universally regarded by the community, it is possible to identify some recurrent and important characteristics that can be generally accepted for use in this study. First, all the definitions either directly state or allude strongly to agents acting in such a way as to satisfy some goal or criteria. The second major trait of agents seems to be that they are autonomous, or make their own decisions without interference from the user. Finally, there is a recurrent theme that agents are one of many. That is to say, each agent is discrete and while part of a larger group, you should still be able to pick out a particular agent from its environment even if you are interested in the overall emergent behavior of the system. Therefore, for the purposes of this work, the requirements of goal-seeking, autonomous, and identifiable will be used to define an agent.

Agent-based modeling arose out of the realization that engineering problems were becoming increasingly more complex and interdependent.[88] As various fields including complexity science, management science, social sciences, and system dynamics evolved, so too did the need for a modeling and simulation capability to help quantify the answers to questions being asked. Across a wide range of disciplines, agent-based modeling is used in situations where the investigator is looking for overall patterns or behaviors from a collection of basic components. For a simple example of why this concept is useful, consider an ant colony. Physics can describe the mass of an ant, how fast it can go, and how much it can carry. Consider though if the investigation involved how the ants can rebuild their hill if it is destroyed. Physics is limited to describing how much work each ant can do by itself. However, observing ants doing their work shows that there are well defined patterns in how the group behaves and a seemingly random collection of

activities results in the construction of a new anthill. Think of each ant as an agent that has some simple rules, such as carry dirt, or stack dirt. Physically, one could determine how much work each ant could do but the aggregate of what each ant does and how the ants relate to each other is really why such a simple creature can construct very complex dwellings. The physics of the problem is insufficient and agent-based modeling would allow for the simulation of the complex interactions between all the agents.

It is easy to see how the concept of identifying emergent behaviors could be of immense use once it is extrapolated to science and engineering. ABM is used, for example, to assess crowd control problems. Each person in a large crowd is an agent who has a goal, for example get out of a burning room. The pattern of how people leave under duress aids in improving design of buildings. In the field of transportation, traffic can be modeled with the use of agents. Every automobile, bike, or pedestrian is an agent that is trying to get to a destination safely. The physics determines how fast each object goes, but by treating them as agents and modeling characteristics such as aggressiveness or curiosity, it is possible to see patterns such as accidents or “rubbernecking” emerge over a large population.

3.2. How to Accelerate Analysis

One of the challenges now that a means for quantifying doctrine has been found, is how to explore doctrine. Part of the deficiency in current techniques is that only a handful of options are ever considered. The first step in quantifying many possible approaches is to accurately describe and bound the design space. One technique for this is referred to as a morphological matrix, or matrix of alternatives. This tool allows the potential settings for every entity or property in a model to be articulated in a concise fashion.

The question now becomes how all of those combinations can be assessed. The brute force answer is to run every case. However, depending on the number of variables and settings, the total number of cases can be astronomical. What is needed is a more intelligent way to sample points in the design space.

The answer to this problem can be found in a field of statistics known as design of experiments. A concise definition for design of experiments is given below,

Experimental design is a body of knowledge and techniques that enables an investigator to conduct better experiments, analyze data efficiently, and make the connections between the conclusion from the analysis and the original objectives of the investigation. [162]

This definition of experimental design (or design of experiments) clearly captures why this set of techniques is so well suited for this problem. In most cases, the design space is far too large to do every experiment so design of experiments can be used to choose the right experiments and then analyze the results properly.

The field of design of experiments (DoE) got its start with R.A. Fisher in the 1920's. During this time, the agriculture industry was trying to determine what factors

most influenced crop production. However, since the crops took a long time to grow and there was a limit on resources available for testing, it was not possible to run every combination of seed, soil, and sustenance that was desired. Using design of experiments it became possible to test different combinations of factors and ascertain the optimal settings. The characteristics of this research, while different in purpose, mirror those of the original need for design of experiments. Essentially, the number of experiments desired is greater than what current resources allow and therefore an enabler in the form of experimental design is used.

Design of experiments is one aspect of conducting a good and valid study. Wu and Hamada present the following seven steps for experimentation:

1. State objective
2. Choose response
3. Choose factors and levels
4. Choose experimental plan
5. Perform experiment
6. Analyze data
7. Draw conclusions [162]

The objective is why the study is being conducted in the first place; it is a statement of purpose. Responses are the parameters which, if properly measured or calculated, can provide insight into the objective possibly by confirming or refuting a hypothesis. While responses are the dependent variables, factors are the independent variables. The levels of the factors indicate how many settings or values each input variable can take. Step four is where design of experiments creates a set of cases or combinations of input variables. Once the experiments are performed, the next key step is how to make effective use of the information gained by executing a design of experiments. The next section will discuss a set of techniques to do precisely this: surrogate models.

The field of surrogate modeling (also known as meta-modeling) focuses on the creation of accurate representations of more complex analysis codes. Through careful statistical planning, a design of experiments is run and the results of the analysis are crafted into a surrogate for the original code. The surrogate model can be evaluated instantaneously, is accurate to within a specified tolerance, transparently shows the impact of variables on responses, and yet protects the integrity of the code that was used to create it.

There are several types of surrogate models commonly used today. Examples include polynomial response surface equations (RSE), neural networks (NN), Gaussian processes, and Kriging. Each method is suited for different types of problems. For example, RSEs work well for quadratic type functions and are easier to create. However, they fail to capture non-linearities and discontinuities. For this problem where there is anticipated to be a complex interaction between the materiel and non-materiel variables, a neural network surrogate model is more appropriate since it can account for more complex behavior. This hypothesis will be tested following the execution of the design of experiments.

Hypothesis 3b: Once the design space has been articulated through a morphological analysis, a design of experiments will permit intelligent sampling of that design space and surrogate models will enable rapid and accurate computations therein.

3.3. How to Incorporate Cost

If non-materiel approaches provide a lower cost means for closing capability gaps, then determining how much cost was incurred to achieve that change is essential. The cost assessment will combine with the effectiveness calculation to compare approaches across multiple acquisition criteria. This will establish the “bang-for-your-buck” that comes from both materiel and non-materiel changes. This section will establish basic cost definitions, outline methods for conducting the cost analysis, and provide assumptions used in the cost assessment.

3.3.1 Cost Analysis

There are two terms to define concerning how the cost will be handled in this research: cost analysis and cost estimation. According to the GAO report on best practices for costing, cost analysis is defined as creating and reporting cost estimates through analytical means. Cost estimation is the assemblage and manipulation of data, tools, and methods to predict a future cost. The guidebook further states that cost analysis is an important part of decision milestones within an acquisition [155]. This notion of cost as a key criterion in decision making is reinforced by the Operation of the Defense Acquisition System (DOD 5000.2) which lists the requirements for an acquisition process at every milestone. Among the many criteria at each milestone are one or more cost analysis requirements.

Given that cost is an important criterion throughout acquisition, the next question becomes what is needed in a cost analysis to provide decisionable information? There are two terms in the Defense Acquisition Guidebook that merit explanation with regards to

which costs are covered in the analysis: lifecycle cost (LCC) and total ownership cost. LCC is everything that can be reasonably attributed to the acquisition at hand including direct and indirect costs [151]. Total ownership cost includes not only LCC but also additional infrastructure and business costs that may not be attributable to the acquisition. Now, as far as which category of cost needs to be considered, the Guidebook says “In general, traditional Lifecycle cost estimates are in most cases adequate in scope to support decisions involving system design characteristics [151].” Basically, an analyst at a particular stage of acquisition has a limited amount of knowledge and resources and the direction for performing the cost analysis recognizes these limitations. The level of detail required to execute the cost analysis should only be what is needed to differentiate between options and only what can realistically be estimated at a given point in time.

Given that LCC is the broad category of cost being considered in this research, further attention needs to be paid to what goes into a LCC assessment. There are four stages of LCC: Research and development, Investment, Operating and support, and Disposal [151]. Research and development is the cost of concept refinement and technology maturation. Investment costs are those associated with fabricating the initial equipment, spare parts, etc. Operating and support costs are those expenses for deploying, using, and sustaining a viable system. Disposal, as the name suggests, is the category of cost that covers decommissioning a system so it is safe, then getting rid of the materiel.

Even though an LCC can be broken down into only four stages, it should be clear to see that there exists a potential for many cost estimations within each stage of an LCC cost analysis. This research is not meant to be an exhaustive LCC cost study nor is it meant to change how the DoD performs cost analyses in the future. Therefore, it is

imperative to establish the scope of the LCC analysis both in terms of what is to be analyzed and the required fidelity to analyze it.

First consider the stages of lifecycle cost and their applicability to an investigation into the analysis of non-materiel solutions against materiel ones. For a non-materiel solution, the R&D costs would be how much it costs to come up with the proposed change in doctrine or training or personnel and so forth. The investment costs would be what it takes to get the non-materiel change to be operational in the military. Operating and support costs are what it would take to maintain a viable operation with the appropriate non-materiel solution. Disposal is not as straight-forward. For non-materiel solutions like doctrine and training, there is no physical system to dispose of as there would be in a traditional materiel solution or in a facilities disposal.

Now consider the fidelity at which the stages must be analyzed. The GAO guide cites eight different levels for costing. They are shown in the list below[155].

1. Independent Cost Estimate
2. Total Ownership Cost Estimate
3. Analysis of Alternatives and Cost Effectiveness
4. Economic Analysis
5. Order of Magnitude
6. Independent Government Estimate
7. Estimation Completion
8. Independent Review

Independent Cost Estimate and Total Ownership Cost Estimate are very detailed analyses meant to cover the complete lifecycle. Analysis of Alternatives and Economic Analysis are business oriented, i.e. does the proposed product or process make fiscal sense. Numbers 6 through 8 are review assessments designed to either monitor the original cost analysis or adjust the analysis based on new information for future budgets. The costing type “Rough order of magnitude” is applicable to any portion of the

acquisition life cycle. The level of information required to generate this type of assessment is in line with the conceptual phase of acquisition and thus this GAO category of cost analysis is most applicable for the problem at hand.

Since doctrine is basically the “how” of military operations, it is reasonable to conclude the tracking operating and support costs is essential. Additionally, the impact of doctrine on new technologies is hypothesized to be quite strong. Therefore, including acquisition cost into the estimate is also necessary.

3.3.2 Cost Estimation Techniques

The next step in the cost analysis process is to determine the method that will be used for conducting the cost estimations. The Defense Acquisition Guidebook specifies four classes of methods for estimating costs: parametric, analogy, engineering estimate, and actual costs [151]. A parametric method employs statistical techniques in the development of Cost Estimating Relationships (CERs) of variable complexity. Important to note is that for parametric methods, the relationships are only applicable for the class of problem they were created for and the bounds of data that were available. Next, the analogy approach, as the name implies, draws a comparison between the current system and an operational system of which the amount of information is complete or almost so. The analogy can be derived either from quantitative analysis or from expert opinion. An engineering estimate is a more detailed approximation that is created by breaking a system down into components which are then costed individually and then rolled up into a total cost estimate. The actual cost method requires fabrication of parts or early designs of the system. From the information available after the early stages of production are complete, improved estimates can be made about the remaining costs of the system.

Before proceeding, it is worth examining other authoritative sources to see if other methods exist that would apply to this problem. According to the Department of Energy Cost Estimating Guide, there are six cost estimating methods: bottoms-up technique, specific analogy technique, parametric technique, cost review and update technique, trend analysis technique, and expert opinion technique [152]. The bottoms-up technique specifies that the total system be decomposed into smaller subsystems and then aggregated for a total estimate. This is a match to the ‘engineering estimate’ referenced in the DoD guide. Similarly, the DoE’s analogy and parametric techniques match to the DoD’s methods. Cost review and update as well as trend analysis both deal with modifying existing cost estimates. The ‘review and update’ technique is a logic check on what was originally quoted while ‘trend analysis’ compares what was projected to what has been done so far. The final technique is expert opinion. This method is to be used when no data can be found and brings in subject matter experts to create a cost estimate.

The Department of the Army Cost Analysis Manual lists parametric, analogy, expert opinion, and engineering/bottom-up as methods for cost analysis [153]. Yoner in his thesis on Major Weapon Systems Acquisition (2001) cites analogy, parametric, expert opinion, engineering, and extrapolation as techniques [164]. He further states that the first three methods are “gross estimates” and are most useful in the early stages of design where there is greater uncertainty regarding the system.

To meet the challenges of this research, a combination of analogy and parametric techniques will be used. These provide the appropriate level of detail for the stage of design corresponding to a capability-based assessment. Additionally, it should be

possible to obtain publically available historical data which can be applied to the approaches being investigated herein.

Hypothesis 3c: Acquisition cost and operational costs will provide sufficient information to differentiate between doctrinal approaches, materiel approaches, and combinations thereof while maintaining a fair comparison for cost-effectiveness calculations.

3.3.3 Costing Assumptions

There are several key assumptions which bound the costing space for this study. First, it assumed that acquisition cost and operating cost will capture the differences between candidate approaches. In actuality there will be other costs associated with developing any potential system. For example, when a new doctrine is created, there is a cost incurred with vetting that doctrine, developing guidance, and training personnel to the requisite level of proficiency. It assumed that this cost is negligible relative to either the acquisition or operating costs.

Also, it is assumed that the missions executed by the system are uniform across its life cycle. In actuality different systems could be used to perform a variety of missions. However, it is assumed that changes in operational tempo over time could apply to any of the systems and therefore more detailed computations regarding these factors would be negligible in regards to differentiating between candidate approaches.

3.4. *How to Present and Compare Doctrine and Materiel Results*

Recall that one of the motivating documents for this dissertation from the Air Force Studies and Analyses Agency spoke to the importance of comparing different approaches, particularly comparing the non-materiel to the materiel. Thus far it has been shown that a constructive agent-based simulation will be used in conjunction with cost estimation to evaluate potential materiel and non-materiel approaches. There are many things that need to happen in order to enable a side-by-side comparison of materiel and non-materiel approaches. Fortunately, the steps up to this point have accomplished many of the shortcomings of the current methods.

First, the biggest problem with comparing materiel and non-materiel approaches in the past has been that there is a wide discrepancy with the fidelity of two separate analyses. Now, instead of a high-level materiel simulation and a qualitative survey of non-materiel approaches, there is a simulation that captures both materiel and doctrine solutions at the same level of fidelity. Furthermore, by conducting the analysis in a common evaluation environment the assumptions used in the study are consistent. If a materiel study was done in a simulation environment then there are assumptions as the real world system becomes abstracted to the models. At the same time, surveying experts as to the impact of doctrinal changes carries with it assumptions inherent in the experts' experiences and interpretation of the survey. Now the assumptions are constant and transparent setting the stage for an equal comparison of approaches.

Just because it is now possible to compare approaches does not mean that the task is easy. One of the goals of this research is to dramatically open up the space of solutions considered in a study of doctrine. This means that there will be a large numbers of cases

and a great deal of information to convey to the decision maker. The question now becomes: how do we convey that information?

3.4.1 Visualization Tools

Capability-based assessments are a great analytical challenge due to the complexities of the scenarios, technologies, interactions, behaviors, and perhaps more than any other reason, the enormous potential space of solutions. A great deal of effort has been put into developing methods for quantifying the pieces of the CBA as well as accelerating the process to enable extensive design space exploration. However, if all of this data is not efficiently conveyed, then all of the hard work will be lost.

There are several tools that are employed to aid in the visualization, reduction, and analysis of data. First is a tool called the prediction profiler, part of the JMP statistical software package. The prediction profiler is a tool capable of parametric sensitivity derivatives. The terms prediction profiler and dynamic sensitivity derivatives are used interchangeably. To best explain this tool, a notional set of equations will be used. Suppose as an example that there were two outputs of an analysis, Y1 and Y2. Further suppose that there are three variables that changed these responses, X1, X2, and X3. After creating surrogate models, there is a functional relationship between Y1 and the three inputs, and Y2 and the three inputs. The next step in the process is to explore the solution space and determine the impact the variables, and their interactions, have on the responses. To illustrate how this is done with the prediction profiler, see Figure 8 below which contains the notional example variables mentioned above.

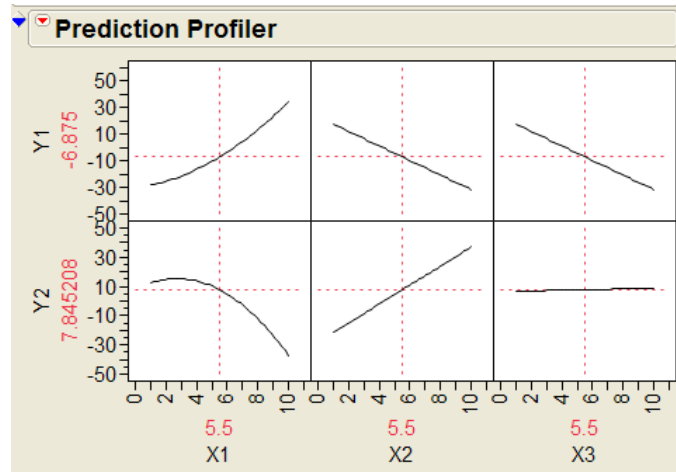


Figure 8: Notional Dynamic Sensitivity Derivatives

Along the horizontal axis in the profiler is shown each input variable in black and just above the variable name in red is the current value of that variable. In this case all three x variables have a value of 5.5. The vertical axis in the profiler displays the responses, in this case Y1 and Y2. Right next to the name of the variable there is a number, in red, that represents the current value of the response. So in Figure 8 there are six subplots which contain all of the combinations of responses plotted against the input variables. Within each subplot there is a curve of the function for a given response versus a given input *assuming all other variables are held constant*. In this way, it is possible to think of each curve as a partial derivative.

With this understanding of the graph in mind, there are several pieces of information which can be gleaned immediately from the profiler. First, it is possible to determine the degree and direction of the variables on the responses. For example, in Figure 8, it can be seen that X1 has a quadratic type effect on Y2 while X2 and X3 have a linear influence on Y2. Also, between X2 and X3, the slope of the curve (the partial derivative) for X2 is

greater than X3. Similarly for Y1, X1 has an increasing quadratic effect while X2 and X3 both have negative linear effects.

One of the most important things to remember regarding the prediction profiler is that although the graphics presented here are static, in actuality the tool is dynamic. The values of any of the input variables can be changed and immediately a new value of the response is calculated. For an example of this, see Figure 9 below.

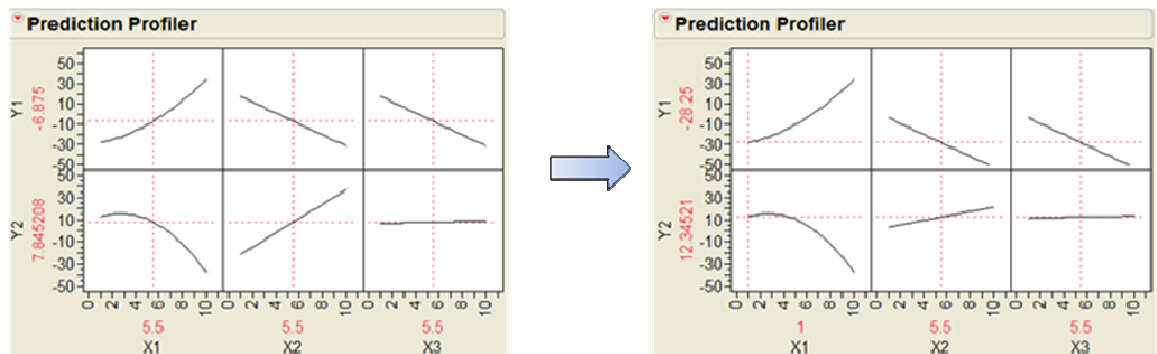


Figure 9: Changing Parametric Sensitivity Derivatives

In the left-hand profiler of Figure 9 the settings for all of the input variables are the same as what was shown in Figure 8. Now the value of X1 is going to be reduced from 5.5 to 1. The result of this change can be seen in the right-hand profiler of Figure 9. The first change to recognize is that the values of the two responses have changed. It is important to emphasize that this calculation happened instantaneously. Whereas before, changing a variable meant changing a setting in the analysis environment and running an entirely new case, the surrogate allows this instantaneous computation over the ranges investigated and to the fidelity determined during construction of the surrogates.

One other item to note in Figure 9 is that when the value of X1 was changed, the slope of the curve of Y2 plotted versus X2 decreased. Recall that each curve in each subplot was an instantaneous partial derivative of one variable on one response given that all other variables were held at their current values. Given that the partial derivative changed as it did, one can infer that there is an interaction between X1 and X2.

There are many benefits to using the prediction profiler to visualize and execute the surrogate models. The rapid analysis speed mentioned above is one of the key benefits. Additionally, since there is a graphical representation of the functions, the prediction profiler provides exceptional transparency. What transparency means is that the user can see exactly what is happening within the analysis environment over the ranges investigated. Therefore, bringing subject matter expertise into the picture, it can be determined if the behavior matches expectations; a form of verification and validation.

The prediction profiler is not the only powerful visualization aid available. The next such tool approaches the issue of design space exploration from another perspective than the prediction profiler. Where the profiler gave detailed information about any particular combination of input variables including partial derivatives about that point and values of the responses, the multivariate plot shows a large number of cases across all dimensions of the design space at the same time.

Creation of the multivariate plot relies once again on the surrogate models. Recall how with the profiler it was possible to immediately change the combination of input variables and see the new response values. Taking that to the next level, with such rapid analysis speed, a Monte Carlo simulation can be employed to randomly sample a range of input values and use the surrogate models to instantaneously evaluate the responses.

Think of it as changing the values as was shown in Figure 9, only done thousands of times in a known statistical way and in a matter of no more than seconds.

Once the Monte Carlo simulation is performed the results can be plotted in the multivariate matrix. An example plot based on the example set of inputs and responses from the profiler explanation is shown below in Figure 10. At first glance this graphic can be overwhelming. However, powerful insights can be gained from this type of display and therefore a detailed explanation of how to read and use such a chart will now be presented.

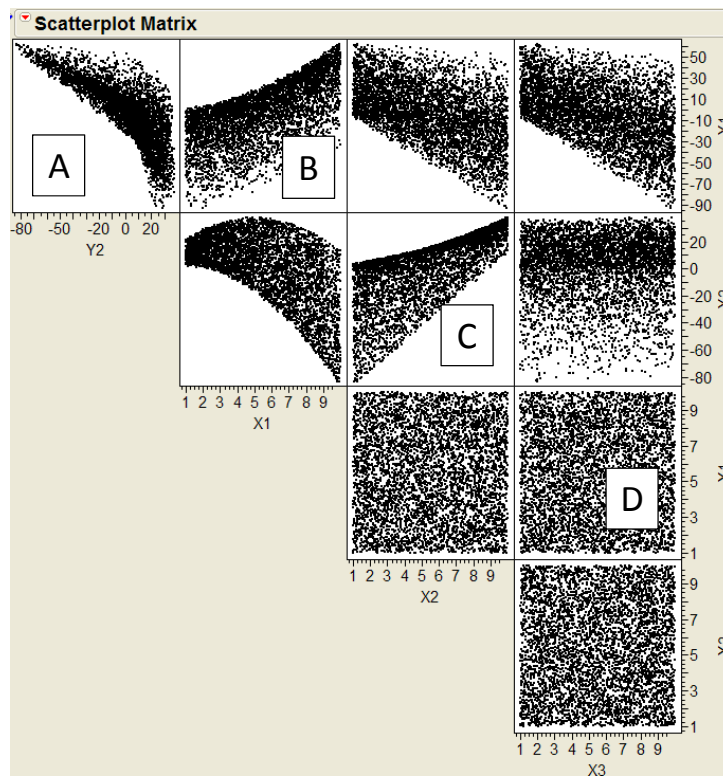


Figure 10: Notional Multivariate Matrix

A single multivariate matrix is composed of several subplots that show each response and input plotted against every other term. For the example profiler there were 6 subplots as two inputs were plotted against three inputs, now however, in addition to those 6 combinations, there is also a plot of Y1 versus Y2 and three plots showing the input variables plotted against each other for a total of 10 subplots.

To read the multivariate matrix, first examine the extreme right edge of the figure. From top to bottom the terms Y1, Y2, X1, and X2 can be seen next to a corresponding axis. Next, following the diagonal from top left to bottom right, there are labels for each column Y2, X1, X2, and X3. In other words, the subplot labeled 'A' is a plot of Y1 on the vertical axis and Y2 on the horizontal axis. Subplot 'B' shows Y1 versus X1. Subplot 'C' shows Y2 versus X2 and subplot 'D' shows X1 on the vertical axis and X3 on the horizontal axis. Note that the three plots of the input variables against each other appear as solid squares of uniform density while the other plots have varying shapes and densities. This is because the Monte Carlo simulation ran uniform random cases across the range of input variables and thus the multivariate matrix can function as a diagnostic to make sure the complete design space has been explored.

There are many pieces of information that can be learned from this plot. First, subplot 'A' in Figure 10, which plots Y1 versus Y2, shows the limits of performance for the two metrics. That is to say, if the goal is to maximize both metrics, then there is a boundary beyond which no solutions exist. This limiting set of solutions is called a Pareto frontier.

The next way to use this multivariate is to employ a technique called filtering. Given that a Monte Carlo simulation was first used to population space, this entire technique is referred to as filtered Monte Carlo. To see this process in action, examine Figure 11. On

the left hand graphic is the complete set of points just as in Figure 10 only now there is a solid red line drawn horizontally across the top row where $Y1 = 0$. This is a constraint line where for this example only solutions greater than $Y1 = 0$ will be considered. The right hand side of Figure 11 illustrates how the solution space looks once all of the points with $Y1 < 0$ have been removed. What makes this filtering so powerful is that the points are removed in all dimensions and the user can see what remains of the design space. Comparing the two sides of the graphic in Figure 11 it is possible to see how the number of points has been reduced. In fact, looking at the plot of $X1$ versus $X2$ shows that there are no points with both a low value of $X1$ and a high value of $X2$. That entire corner of the design space has been removed through filtering.

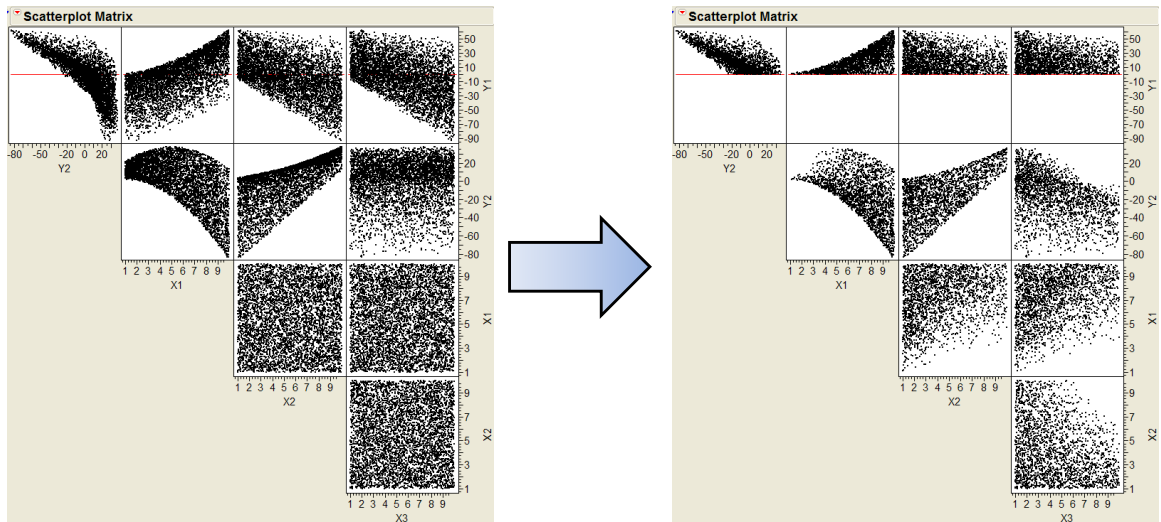


Figure 11: Filtering Y1

Next, putting in an additional constraint that $Y2$ must be greater than 0 can filter the design space even further. Graphically, this filtering step is shown in Figure 12. This time a solid blue line is drawn at $Y2 = 0$. From the left hand side to the right hand side of

Figure 12 all the points which do not satisfy this constraint are removed. There is now only a fraction of the total number of cases from the original multivariate.

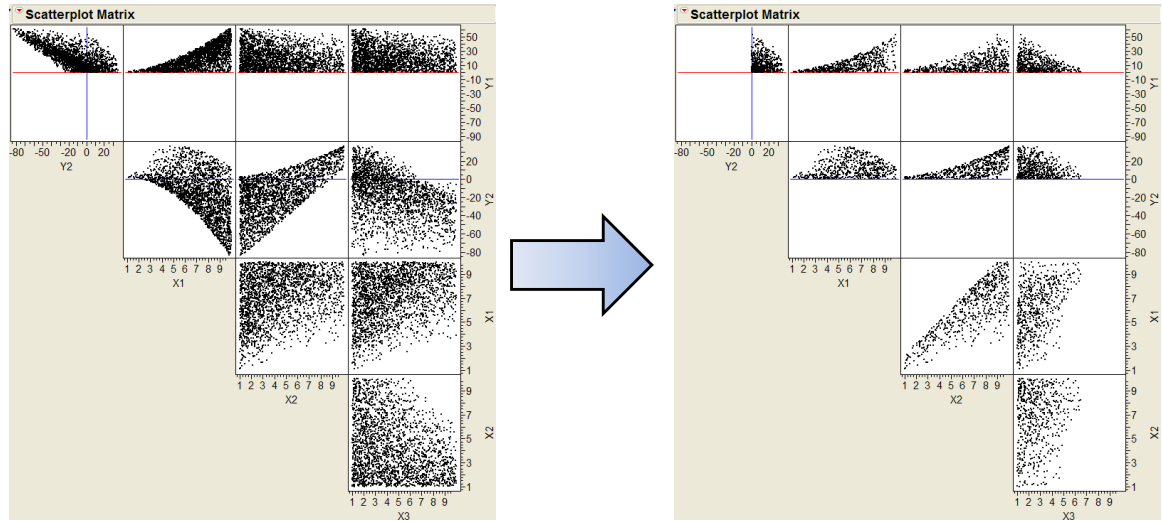


Figure 12: Filtering Y2

Once all of the constraints have been imposed, final analysis of the data can be performed. For an example what kind of information is now available, see Figure 13 below. First, in the bottom-most plot, X2 versus X3 it is clear that there is a limit on what values X3 can take. What this plot is showing is that in order for Y1 and Y2 to both be greater than zero, X3 cannot be greater than 6.5. Also, there multivariate shows that there is a tradeoff between X1 and X2 with a very defined line beyond which no designs exist to satisfy the constraints.

Finally, once the user has imposed all constraints and filtered the design space, it is possible to query any point and ascertain every property of that particular case. In Figure 13 a red X marks a point on the Pareto frontier of Y1 and Y2. If the user wishes to further investigate this point, the values can be seen in every other dimension.

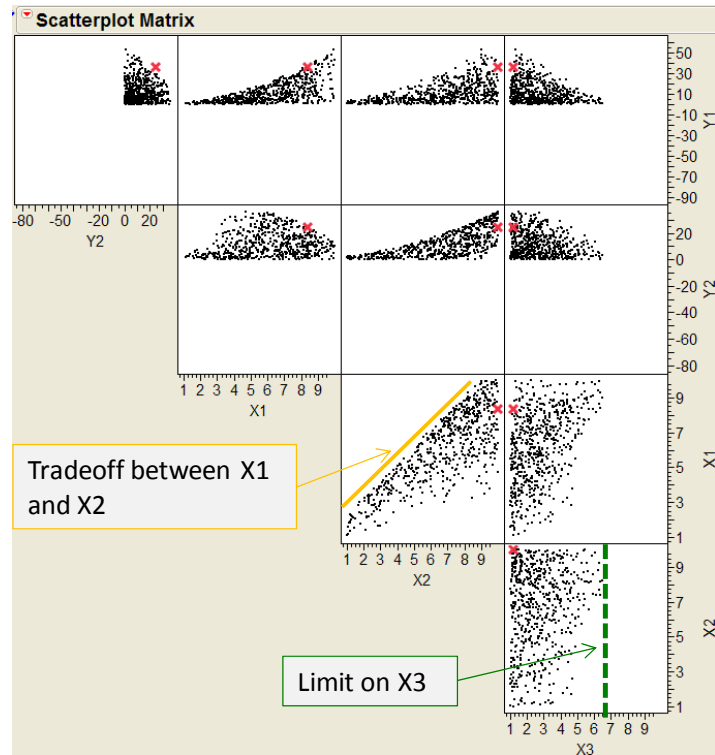


Figure 13: Results of Filtering Process

With the prediction profiler, it was possible to input any combination of input variables and rapidly evaluate the responses. This type of evaluation is referred to as bottom-up evaluation where the individual cases are supplied and then calculated. The multivariate matrix on the other hand looks at all possible combinations and based on requirements allows the user to evaluate which alternatives are feasible. This technique is referred to as top-down analysis.

Recall that a capability-based analysis is supposed to output a prioritized list of possible approaches. Traditionally most engineering analyses use the bottom-up method. But this begs the question of: how are the approaches computed by the bottom-up method selected? The power of the top-down method is that the complete space of possible

approaches, as defined in the problem definition stage, is explored. This eliminates biases in the input variables and may identify new configurations that wouldn't have otherwise been explored. Furthermore, since the surrogate models were necessary to speed up the bottom-up analysis, extending their functionality to enable the top-down analysis is a minimal investment of resources for the corresponding added richness of information.

Hypothesis 3d: A combination of prediction profiler for visualizing surrogate models and multivariate plots for visualizing the design space will permit a significant advancement in the scope and clarity of information output from a CBA.

3.5 Hypothesis #3

The hypothesis to research question 3 could not be addressed until a comprehensive knowledge of doctrine, defense acquisition, and quantitative techniques was established. Therefore, in summary of the preceding sections developed four sub-hypothesis, the final overall hypothesis for research question #3 is presented now:

Through the use of

- a. Agent-based modeling in a constructive simulation
- b. Morphological matrices, design of experiments, surrogate modeling
- c. Incorporation of acquisition and operating and support costs
- d. Filtered Monte Carlo and Multi-attribute Decision Making

this method will rigorously compare doctrinal approaches to materiel approaches for a capability-based assessment. This method will be advance the state of the art in doctrinal analysis by increasing the number of cases assessed and improving transparency of analysis.

CHAPTER IV

RESEARCH PLAN

4.1 *Example Problem Elaboration*

Earlier the notion of exploring a Revolutionary Hunter/Killer UAV problem was presented to help scope the literature search the needed to be performed. However, now that the research has progressed to the point of developing the example problem, it is imperative to gain a richer understanding of the problem in question. This section will go through the history of UAVs as well as define the state of the art and finally challenges for the future.

4.1.1 History of UAV Development

While UAVs are thought of by many as modern-day, cutting edge developments, the truth is that unmanned aircraft have been around as long as manned aircraft. Langley's Aerodrome became the first heavier than air object to fly when it was launched in 1896 [73]. In the following years, particularly in World War I and II, unmanned air systems were predominately weapons. The next logical step was to transform UAV's into target drones [7]. However, starting in the 1960's and in particular during the Vietnam War, unmanned aircraft were used for reconnaissance. Examples include the Lightning Bug and Firebee aircraft. However, the use of UAVs reached a plateau in the United States following the Vietnam War until the Israeli Bekaa Valley campaign of

1982. In this campaign, Israel employed UAVs as intelligence, surveillance, and reconnaissance (ISR) platforms in coordination with their manned aircraft to devastating success. Since that time, investments in UAV programs have increased dramatically and resulted in a wide array of platforms including the Predator, Reaper, Global Hawk, and families of smaller UAVs. Of particular interest to this study are the Predator and Reaper and each will be addressed next.

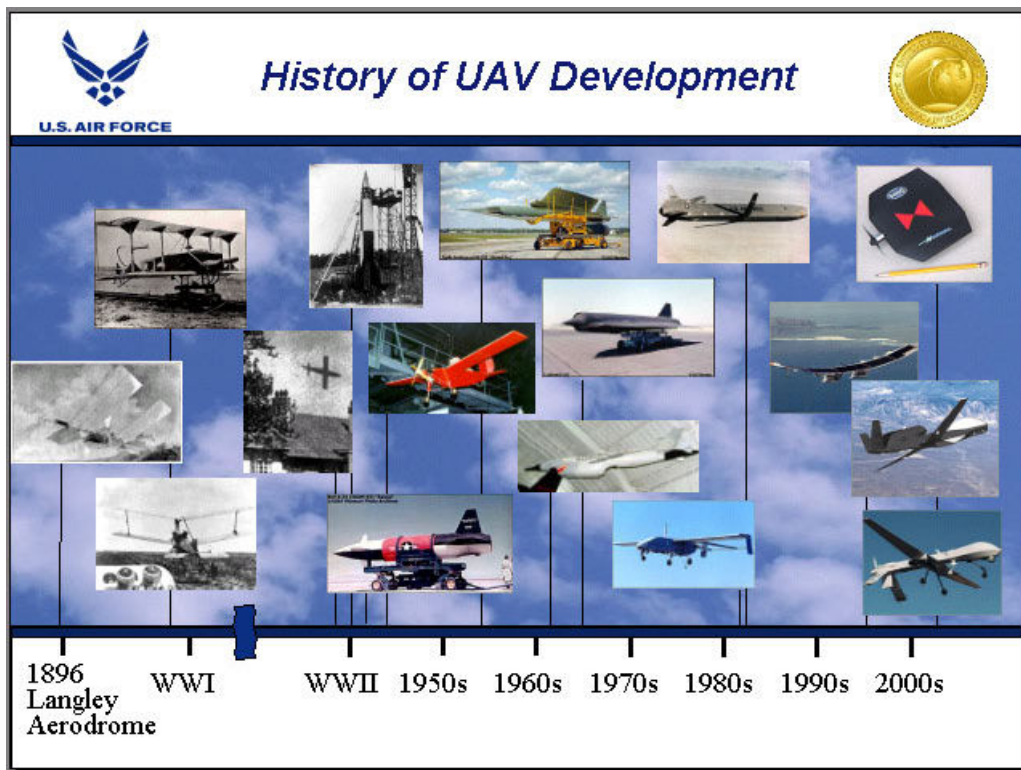


Figure 14: UAV Timeline [73]

4.1.2 MQ-1 Predator

The Predator aircraft was developed in the mid-1990s as a technology demonstrator, formally called and advanced concept technology demonstrator or ACTD

[100]. The system, which consists of four UAVs linked to a single ground station, was developed to showcase advances in unmanned technology [148]. However, it quickly became apparent that these systems could be employed in missions around the globe. Predator was a valuable ISR platform during operations in Bosnia in the 1990s and began its service in the global war on terror in ISR missions.

In 2002, a dramatic step forward was taken as Predator aircraft were equipped with Hellfire air-to-surface missiles and became a strike asset. They were used to loiter over remote areas of interest and attack high value targets such as convoys of al Qaeda members in Afghanistan and elsewhere including neutralizing one of the terrorists responsible for the bombing of the USS Cole in 2000 [68]. While it may seem ordinary to think of UAVs as carrying weapons, this leap forward by Predator has been compared to when biplanes were first equipped with machine guns.

The MQ-1 is described as a medium altitude, long endurance (MALE) aircraft whose primary mission is armed reconnaissance. The vehicle is remotely piloted, that is to say a human on the ground directs the flight path as well as sensors and weapons on-board the aircraft. Included in the sensor package are a visual camera and an infrared (IR) detector. The weapon load-out for a Predator consists of two AGM-114 Hellfire missiles [148]. Below in Figure 15 is a detailed list of specifications for the MQ-1.

Contractor	General Atomics
Thrust	115 horsepower
Wingspan	48.7 feet
Length	27 feet
Height	6.9 feet
Weight	1,130 pounds
Maximum Takeoff weight	2,250 pounds
Fuel Capacity	665 pounds
Payload	450 pounds
Cruise Speed	84 mph
Range	up to 400 nautical miles
Ceiling	up to 25,000 feet
Armament	two laser-guided AGM-114 Hellfire missiles
Crew (remote)	Two
Unit Cost	\$40 million (\$1997)



Figure 15: MQ-1 Predator Specifications [148]

4.1.3 MQ-9 Reaper

After the initial success of the MQ-1, it became clear that an improved UAV would be of great value to the Air Force. Accordingly, the MQ-9 Reaper (originally the Predator-B) was created to provide greater speed and payload capacity. As opposed to the MQ-1 which is limited to carrying only Hellfire missiles, the MQ-9 can carry the Hellfire, GBU-12 Paveway II, and GBU-38 Joint Direct Attack Munition (JDAM) [149]. Similar to the Predator, the Reaper is remotely piloted by video camera and also carries IR and SAR sensors [149]. Below in Figure 16 are more detailed specifications for the MQ-9 Reaper.

Contractor	General Atomics
Thrust	900 shaft horsepower
Wingspan	66 feet
Length	36 feet
Height	12.5 feet
Weight	4,900 pounds
Maximum takeoff weight	10,500 pounds
Fuel Capacity	4,000 pounds
Payload	3,750 pounds
Cruise Speed	230 mph
Range	3,682 miles
Ceiling	up to 50,000 feet
Armament	Combination of AGM-114, GBU-12, GBU-38
Crew (remote)	Two
Unit Cost	\$53.5 million (2006)



Figure 16: MQ-9 Reaper Specifications [149]

4.1.4 UAVs Today

Both of the above systems, as well as the RQ-4 Global Hawk and various small UAVs have been extremely successful in recent military operations. No less than nine types of UAVs were employed in Operation Iraqi Freedom (OIF). Global Hawks flew only 5% of the high altitude reconnaissance missions in OIF yet accounted for 55% of the time-critical targeting information against enemy air defenses [139]. Between 1996 and 2004, Predator aircraft flew over 100,000 flight hours and nearly 70,000 of those areas were in some kind of deployed situation [139].

The level of success for these unmanned systems has been extraordinary and very encouraging for the Air Force. What makes the success particularly amazing for Predator and Global Hawk is that both were an ACTD and not the result of a formal requirements process. The Air Force is planning to buy an additional 170 Predators and at least 50 more Reapers by 2012 while phasing out a comparable number of F-16s. In 2002, there were approximately 25,000 flight hours by UAVs across all branches of the military. In

2006 there were 175,000 flight hours. At the end of the 1990's, the total yearly Department of Defense expenditures on UAVs including development, procurement, and operations was around \$300 million. Today that number is approaching \$3 billion [139].

This huge upswing in investment is necessary to meet the growing demand that the next generation of unmanned systems will have to meet. Given that Predator and Global Hawk were an ACTD, their ability to fulfill an ever growing set of mission demands is minimal. Specific problems with ACTDs include lack of integration with other systems, limited operation and support plans, and inadequate stockpiles of spare parts. All of these drawbacks increase the operation and support costs. The next UAS must be designed specifically to meet challenges presented by the global war on terror as well as the retirement of so many of the traditional “workhorses” of the modern military. It was already mentioned at the beginning of this chapter that the Revolutionary Hunter/Killer (Rev H/K) program would serve as the test case and that this air craft will need to replace the MQ-1 and MQ-9. *It is worth mentioning at this point that NO classified, sensitive, or for official use only (FOUO) materials are used in this research.* Additionally, the baseline Rev HK program does not investigate non-materiel approaches. Therefore, it is necessary to search for pressing challenges facing the next generation of unmanned air vehicles, in particular those which can be addressed through a non-materiel analysis. These challenges will then be correlated to the hypotheses of this research. That will help determine the experiments to be performed.

4.1.5 Challenges for Future Unmanned Aircraft

The arena of unmanned aircraft development is extremely active and there are many players each with their own game-plan for how the future of UAVs should evolve. As such, it is necessary to survey a variety of sources for key challenges and then try to assimilate a list that would be generally agreeable to UAV experts from different fields. From that list, the next step is to downselect based on feasibility and relevance to the problem at hand. This section will focus on the formulation of that list.

The Unmanned Aircraft Systems Joint Analysis Team (JAT) was put together to assess the current state of UAS capabilities and recommend guidance for how future UAS should be planned. One of the first items mentioned in the JAT brief is that there needs to be an exploration of the tradeoffs between endurance and weapons payload. For example, an F-16 or a B-52 may carry a great deal of weaponry, but fuel and crew needs limit endurance and how many of those weapons can be employed. Also included in the challenges for UAVs in regards to weapon systems, the JAT states it is important to understand if the addition of weapons degrades the ISR capability of a UAV and to what extent. If putting too much capability on an aircraft makes it perform poorly in many areas, perhaps multiple aircraft with different abilities should be investigated. The final weapon system related challenge is ensuring that the system can attack targets of opportunity; the so-called “pop-up targets”. This brief recommends performing a DOTMLPF analysis on current and projected UAS capabilities in order to address these issues [121].

Beyond weapons and strike issues, the JAT cites bandwidth as a major issue for the next generation of UAVs. Only so much information can be transmitted per unit of time and there is a finite amount of frequencies available to transmit across. Also listed as a major challenge is understanding the tradeoff between directional line-of-sight systems, omni-directional systems, and satellite communications (SATCOM). Each of these communication methods requires a different piece of hardware and therefore it may first be considered a materiel solution. However, consider that the operational pattern of the aircraft will change based on what they are capable of doing. Finally, this brief discusses the importance of integrating UAVs into general airspace. As the number of unmanned aircraft enter the manned airspace, it is imperative to make certain that unmanned systems do not cause an increase in accidents. The JAT recommends developing CONOPS and doctrine to allow real-time command and control of unmanned systems in manned battlespaces [121].

Another brief that discusses unmanned aircraft issues is a presentation by the then Vice Commander of the Air Armament Center, Brigadier General Chris Anzalone. This brief describes how unmanned systems began as unrecoverable weapons, then became target drones, then systems which could only attack the simplest of targets, then the current state of the art with aircraft such as Predator, and finally the vision for the next generation. A theme throughout this brief is that in the beginning, these systems could only be used on targets that were known days in advance and yet the goal is to have the next generation be able to execute targets within minutes of actionable intelligence. That is to say, future aircraft must be reactive and adaptable. Another of the points of this brief is the degree of positive control that must be exercised with unmanned systems and

where that control comes from. This gives rise to several issues regarding the autonomy of unmanned systems. Next, this brief describes a concept known as persistent area dominance (PAD) where an aircraft deploys a loitering weapon that is able to hold an area at risk for an extended period of time and react quickly. Finally, the brief discusses the need to transmit information rapidly to allow real-time assessment of attacks, which then feeds back to the earlier point of an adaptive air system [7].

The U.S. Air Force Remotely Piloted Aircraft and Unmanned Aerial Vehicle Strategic Vision [139] is a document that provides comprehensive detail about some of the most pressing challenges facing the UAV community. First of all, the document states that the qualities of unmanned systems should lead to new CONOPS; a statement that highlights how little work is being put in to actively developing new systems and doctrine concurrently. Next, the issue of net-centricity is addressed. A network-centric system is one in which the different components communicate and the power of that communication acts as a force multiplier making the entire force stronger than the individual pieces.

Of particular concern is the level of autonomy that should be provided to the UAV. Not only are UAVs well suited to handle mundane tasks that can improve overall mission effectiveness, but providing the UAV organic capability to process information speeds up the kill chain and can increase performance. This document goes on to state that the Air Force “must address the doctrinal, tactical, and policy issues of allowing unmanned aircraft, especially those that are armed, to operate without direct human oversight.” Related to the issue of autonomy is bandwidth. According to the Strategic Vision, “A major challenge is to maintain maximum operator situational awareness and

mission flexibility given autonomous, net-centric vehicle operations and limited bandwidth availability [139].” One of the important new issues in that statement is the mention of the operators specifically. There is an interdependency between how much information comes to an operator per plane and the number of aircraft an operator can control. Also cited in this document is the need for dynamic mission re-planning.

The mission re-planning and autonomy issues are related to another issue addressed, namely integration into the airspace to ensure flight safety. Next, survivability is addressed in the form of low observability and self-defense weapons. The level of survivability for unmanned aircraft needs to be designed specifically to match the mission and airframe importance. That is to say, if the airframe is designed to be expendable, then it is not worth the cost to make it stealthy, for example. On the issue of cost, the Strategic Vision notes how important it is to accurately address cost, in particular how the cost compares to manned systems. For example, it may cost about the same to buy an unmanned aircraft but because it does not suffer from human endurance constraints, it has a lower sortie rate, thereby reducing wear and tear and reducing operating cost over the life of the system [139].

Additionally, it is important to consider the fleet mix when designing a new system. Is it better to purchase a large number of small and inexpensive UAVs or does the military get more capability per dollar by purchasing a few larger and more expensive aircraft? Finally, this document discusses the logistics behind using UAVs. Included in this discussion is autonomous refueling, self-deployment, and pre-positioning. The point being, that unmanned aircraft cannot be created and become a burden on the existing manned infrastructure if the aircraft have to be ferried all around the world [139].

The Congressional Research Service (CRS) identified several investment priorities were identified for UAVs [12]. First among these priorities is examining the cost per capability for unmanned systems and if they can in fact be considered expendable. Of particular interest is the relative cost between manned and unmanned systems. Also, the report questions how much protection should be given to unmanned aircraft. As with the previous reports, the Congressional report is concerned about integrating unmanned aircraft into manned airspace as well as general bandwidth constraints. However, one particularly interesting new challenge is the cultural aspect of UAVs. The armed services have had a difficult time convincing pilots to operate aircraft from the ground and the way this mindset influence the future of UAVs is an important point to consider.

A final document worth examining is the Unmanned Systems Roadmap for 2005-2030 put out by the Office of the Secretary of Defense [100]. There are nine goals for development in unmanned systems during this time period. First, it is desired to develop unmanned systems that can operate in high threat environments to expand the missions that can be accomplished by today's UAVs. Secondly is the requirement for secure communications and data links. Third, the video and data coming from the unmanned systems must conform to already accepted standards and be available in near real time. Fourth, it is the goal of the UAV community to create methods for integrating into all types of environments, manned and unmanned, safe and hazardous. The next goal is relatively straightforward: improve the collaboration between military branches in the development and employment of UAVs. Also, by 2030 it is important to develop alternative propulsion systems. The next goal, like several of the above sources, calls for

all-weather capability in unmanned systems. The eighth goal seeks to standardize weapon practices on unmanned systems including control of weapons release. Finally, the last goal requires that unmanned systems more rapidly find their way into the hands of the warfighter in such a way that they are integrated with existing equipment.

4.2 Selection of Study Problem

Based on the above UAV resources a master list of future challenges is compiled below.

- Fleet mix
- Autonomy
- Bandwidth
- Integration into airspace
- Attacking pop-up targets
- Endurance vs. payload
- Weapons vs. ISR
- Operator information load
- Persistent area dominance
- UAV culture
- UAV logistics

Across the board, fleet mix, autonomy, and bandwidth were recognized as critical issues that UAV's will need to address in the future. Fleet mix describes the problem of how many aircraft and of what type are needed for a given mission; should they be expendable and cheap or is survivability important. Autonomy refers to how much of the mission the aircraft can perform on its own and bandwidth recognizes the challenge of dealing with all the information passing between UAV's and other assets.

There are several challenges that will be eliminated from consideration immediately. The challenges of UAV culture and UAV logistics are beyond the scope of this study and will not be addressed here. Also, examining how much information the operators can process is a human factors problem that, while important, is not considered here. Persistent area dominance is a weapons tactic but involves weapon design that is not considered here. Finally, although, integration into airspace was a recurring need for future UAV's, that problem will deal heavily with safety and adherence to FAA regulations and is beyond the effort available for this research.

Examining the remaining challenges reveals something very interesting: there is a strong interdependence between the challenges. The fleet is made up of aircraft with certain values of endurance and payload as well as certain amount of lethality and sensing capability. Depending on how much capability is given to each unit, the fleet mix will look different. If there are a lot of aircraft each with a large amount of sensing capability then a lot of bandwidth is required to transmit all the information. However, changing the autonomy reduces bandwidth demands and can allow the aircraft to more quickly prosecute their missions, allowing so called pop-up targets to be engaged. Clearly a very complex and interrelated design space is emerging and it is obvious how important these issues are to future UAVs.

From the preceding list, a subset must be selected to scope the problem. Given the overwhelming prevalence of fleet mix as a high priority, that will be the first aspect of the problem which will be investigated. Additionally, examining endurance/payload tradeoffs and weapon/ISR tradeoffs should allow for comparisons between doctrine and

technologies. These will be combined in a scenario designed to examine attack of pop-up-targets.

4.3 Verification and Validation

Verification and validation are two of the most challenging parts of any simulation-based study for the following reason. Recall the earlier discussion where it was stated that, while ideally it would be best to experiment directly on a system of interest, practical limitations dictate that a representative model be created for experimentation. The question becomes then, how accurately does the model represent what the study requires? If the model is not an accurate abstraction of the system of interest, then the results of the study are not valid or useable.

With each step away from the original system, there is greater and greater uncertainty about how closely the model represents reality. As an example, consider aerodynamic testing on a commercial airliner. The best way to go about this study is to build a working aircraft and measure the aerodynamic forces to see how the aircraft will perform. However, this can be expensive and at a level of abstraction, a small scale model might be made. The tests can then be performed on the model but if it is not shown that the small model behaves the same as the full size aircraft, then the experiments do not help with the overall goal of the study. Furthermore, assume for a moment that even a small scale model is too expensive. Perhaps a computer model is made and virtual tests are performed on the computer model. Now it needs to be determined if the coding of the computer model mimics the reality of the full size aircraft. A simple thought experiment such as this shows how quickly the process of vetting the model becomes difficult.

As was alluded to above, there are two means for assessing the veracity of a model: verification and validation. First, consider verification. Verification is the process of establishing that a model is properly executed by the simulator [72]. Said another way, verification looks to see if the computer model is properly coded; a process sometimes known as debugging. In order to complete the verification, it is important to know what aspects of the model need to be represented, how they are to be represented, and then how they are currently implemented. Verification can be a demanding task that requires significant time investment and multiple iterations. However, through careful documentation and manual tracing of the code, verification can be completed for most any computer model.

The other means for assessing model authenticity is validation. While verification is the process of making sure the model does what it is supposed to do, validation is the process of making sure that the model is the right one to represent the system. Put more formally, validation is “the process of determining the degree to which a model or simulation is an accurate representation of the real world from the perspective of the intended uses of the model or simulation [72].” Clearly, verification and validation are intertwined and iterative. If the chosen model is not appropriate, then a new one must be assumed, which then must undergo verification, and then tested again for suitability.

According to Zeigler there are three forms of validity. First, replicative validity is the ability of the model to match the observed behavior of the real system. That is to say, for an observed set of inputs run through the simulator, the results should match the observed metrics to a specified tolerance. Next, predictive validity requires not only that the model replicate prior observations but also accurately predict future observations.

Predictive validity is a stronger case than replicative validity. Finally, an even more detailed level of validity is structural validity. In this case, beyond just being able to generate the proper input-output pairs, the model must realistically represent the state transitions of the real system [165].

Clearly these types of validation are all highly desirable, but how realistic is it to achieve any form of validity? For all three types mentioned above, it is required that a real system exists and observations can be taken from it. However, in conceptual design and acquisition, evaluations are being made about systems not in existence. Furthermore, for a complex system-of-systems there are large numbers of heterogeneous assets that act in very complicated ways. Creating these systems in the real world and performing the necessary missions is prohibitively difficult. While validation is an ever popular arena of research, there currently does not exist a sound and established method for validating, to traditional standards, a conceptual system-of-systems.

If no such method exists, does that mean that nothing can be done to validate the model? According to Alexopoulos and Goldsman, there are three means for establishing validity. One of them is to validate the input-output relationships as was discussed above. The next method is to validate the model assumptions [3]. Any model has some type of abstraction and the formulation of the model abstractions is based on assumptions made on aspects of the system that can be simplified. Often the choice of assumptions stems from a literature search or an expectation about how the model will run. However, by coordinating the assumptions with subject matter experts, insight into the validity of the assumptions can be gained. A subject matter expert could provide information about the

operation of the real world system to either support or revise the assumptions in the model.

The last means for assessing validity according to Alexopoulos and Goldsman is referred to as face validity [3]. This type of validation again leverages subject matter expertise to examine the conceptual framework of the model and determine if it makes sense. Additionally, simple sensitivity experiments can be performed to see if the behavior and trends of the inputs and outputs are logically sound at least from a qualitative standpoint.

Neither assumption validity nor face validity are as strong as even replicative validity. However, it is possible to survey subject matter experts either directly or through their published works to find insight as to whether the assumptions, constructs, and trends of the model are sound and valid. In the case of complex system-of-systems problems this is the extent of the effort which can be effectively performed.

CHAPTER V

IMPLEMENTATION

This chapter describes in detail the implementation of the methodology for quantifying doctrine and materiel approaches in a capability-based assessment. One important distinction is that the functional area analysis portion of the CBA is not performed from scratch. Instead, open source information is used to construct the example problem and generate a notional capability gap. However, while the heart of the analysis and quantification lies in the steps corresponding to the FSA, the procedure outlined for the FAA and FNA is critical to performing sound systems engineering. The problem definition and baselining of capabilities set the game board on which the FSA will eventually be played and doing so in a consistent and traceable manner will improve the outcome of a capability-based assessment beyond the current state-of-the-art.

5.1 Step I: Problem Definition

Earlier in Chapter II, it was stated that the example problem for this research would be the Air Force Next Generation UAS program, also known as the Revolutionary Hunter/Killer. The first step in this methodology is to more rigorously define the problem. In this particular case, the problem is selecting a future unmanned aerial system to execute hunter/killer type missions for the United States Air Force. According to AFDD 1, the Air Force specializes in six distinct capabilities: Air and Space Superiority, Information Superiority, Global Attack, Precision Engagement, Rapid Global Mobility,

and Agile Combat Support. Of these, the capability most closely associated with the Next Generation UAS is precision engagement. This capability entails using measured force exactly where it is needed, including bringing to bear command, control, and force application to cause specific effects [142].

Within precision engagement, there are a variety of missions that the Air Force is responsible for. Examples of these include close air support of ground troops, suppression of enemy air defenses, interdiction operations, and so forth [77]. For this example, the mission is to find and prosecute time-critical, high valued, mobile, unobscured targets in a minimal threat environment.

The next piece of problem definition is description of scenarios. The scenarios are based on an analysis of the missions that the Next Generation UAS will perform. A sample mission within the subset of precision engagement was described above. However, details about the specific performance of USAF assets executing that mission are not available. Since the impetus for the Next Generation UAS program is an anticipated inability to meet more challenging future missions such as the one described above, two scenarios will be described. The first is meant to represent one of the simplest precision engagement missions possible as a benchmark for system performance. Second, there is a more advanced mission which more closely represents the precision engagement mission of prosecuting time-critical targets.

The baseline mission is one in which the targets are stationary and are non-threatening. There is no terrain or atmospheric interference. Targets remain exposed indefinitely. The aircraft stage from a base and transit to an area of operations (AOO)

where they conduct the hunter/killer mission. In order to account for the time-criticality of the mission, an overall time limit is imposed on the simulation.

For the advanced mission which represents future UAS missions, the time-criticality of targets is captured with individual dwell-times. A dwell-time refers to the amount of time that a target is exposed; it can be thought of as a window of opportunity in which the target can be attacked. This mission is significantly more demanding than the baseline mission as the targets may disappear before an asset has a chance to find them. Additionally, the targets will no longer remain stationary and shall be capable of moving within the target area. However, as in the baseline mission, the targets are incapable of returning fire.

There are two steps remaining in the problem definition phase: define the tasks and define the metrics. While all portions of the problem definition step are related, these two in particular go together and will be addressed concurrently.

The authoritative source on military tasks is the Universal Joint Task List (UJTL). This document contains a listing and explanation of every possible task the military could perform as well as the various criteria associated with each task. Tasks apply to all levels of military operations. At the highest level there are strategic national tasks, beneath those are strategic theater tasks, then operational tasks, and finally tactical tasks. Notice how the levels of tasks mirror the levels of doctrine: basic, operational, and tactical. For a sample of a UJTL task, refer to Figure 17 [77].

OP 3.2 Attack Operational Targets

To engage operational level targets and to shape and control the tempo of campaigns using all available joint and multinational operational firepower (includes ground, naval, air, space, and SOF conventional and special munitions) against land, air, and maritime (surface and subsurface) targets having operational significance. Operational targets can be conventional or CBRNE weapon related. (JP 1, 2-01, 2-01.1, 3-0, 3-03, 3-09, 3-54), 3-60) (JP 2-0, 3-55, CJCSM 3500.05)

M1	Minutes	To get ordnance on target after initiation of task.
M2	Percent	Execution of missions requested by components.
M3	Percent	Of high priority missions executed within specified time.
M4	Percent	Of maneuver forces secure assigned objectives.
M5	Percent	Of missions flown/fired achieve desired target damage.
M6	Percent	Of operational fires on time in support of maneuver forces.
M7	Percent	Of planned targets successfully attacked during operation.
M8	Percent	On time of missions with given times on target.
M9	Percent	Of enemy NBC delivery systems engaged/destroyed by friendly forces.
M10	Percent	Of immediate targets successfully attacked during operation
M11	Percent	Of attacks assessed to have greater collateral damage/effects than planned/expected.

Figure 17: Sample Task List From UJTL

At the top of this figure, there is a heading providing the name and reference number for the task. In this case, the heading OP refers to an operational task (SN relates to strategic national, ST to strategic theater, and TA tot tactical). Within the set of all operational tasks, all tasks with a number 3 deal with the employment of firepower. In this case, the number of the task is 3.2, or ‘Attack Operational Targets’. These tasks can be further decomposed to a finer level of detail. For example, OP 3.2.2.2 is titled ‘Employ Electronic Attack in the Joint Operations Area’.

Regardless of the level, all tasks contain a description paragraph under the heading, just as can be seen above in Figure 17. Beneath the description is a table containing all of the metrics which correspond to the given task. Each metric is labeled with a number, contains a unit of measure, and a specification about how the measurement should be made. For example, within OP 3.2 metric 1 (M1) measures how long it takes “to get ordnance on target after initiation of task” in minutes. The universal joint task list is both

comprehensive and authoritative making it an ideal starting point for the selection of tasks and metrics.

In addition to the universal joint task list, each of the services has their own individual task list which extends the concept of the UJTL to service specific missions and concepts. The Air Force Task List (AFTL) contains tasks which correspond to the six capability areas, including Precision Engagement which is the segment of operations most relevant to the Revolutionary Hunter/Killer UAS. A sample excerpt from the Air Force Task List is presented below in Figure 18 [133].

AFT 2.1.1 Perform Lethal Precision Engagement Functions. To cause discriminate strategic, operational, or tactical effects through the use of lethal force. (AFDD 1, AFDD 2, AFDD 2-1, AFDD 2.1.1, AFDD 2.1.1.1, AFDD 2-1.2, AFDD 2-1.3, AFDD 2-1.4, AFDD 2-1.5, AFDD 2-1.7, AFDD 2-3, AFDD 2-7, AFDD 2-9)

M1	Time	From the desired timing for lethal force to cause desired effects.
M2	Distance	From desired impact point for precision weapons.
M3	Distance	From desired location for force placement or position.
M4	Percent	Desired strategic effects achieved.
M5	Percent	Desired operational effects achieved.
M6	Percent	Desired tactical effects achieved.
M7	Cost	To perform lethal precision engagement.

Figure 18: Sample Task From AFTL

Similar to the format for the UJTL, the AFTL contain an identification number, task title, description, and a list of metrics. For this example, AFT 2 and all subtasks relate to precision engagement. Task 2.1.1 Perform Lethal Precision Engagement Functions contains very similar metrics to OP 3.2 and also includes some additional pieces of information such as the cost required to perform lethal precision engagement.

With an understanding of the types of tasks prescribed by the Air Force, the final step is to compile and select the most relevant tasks and metrics to guide the capability-based

assessment. The list below shows the UJTL and AFTL tasks with their corresponding metrics that were selected for the hunter/killer type mission.

- OP 3.2 – Attack Operational Targets
 - M1 – Minutes – To get ordnance on target after initiation of task
 - M7 – Percent – Of planned targets successfully attacked during operation
- OP 3.2.5 – Interdict Operation Forces/Targets
 - M6 – Percent – Of enemy operational targets engaged
- AFT 2 – Provide Precision Engagement
 - M4 – Percent – Of successful precision engagements
- AFT 2.1.1 – Perform Lethal Precision Engagement Functions
 - M1 – Time – From the desired timing for lethal force to cause desired effects
 - M5 – Percent – Desired operational effects achieved
 - M7 – Cost – To perform lethal precision engagement
- AFT 2.1.3 – Equip Forces with Lethal Precision Engagement Capabilities
 - M4 – Cost – To equip forces with lethal precision engagement capabilities

It should be noted that there are many tasks in both the UJTL and AFTL which could relate to some stage of precision engagement or a hunter/killer mission. However, finding then ultimately prosecuting the target at a given cost is the ultimate measure of success. Given that this implementation is designed to highlight a methodology only most critical tasks and metrics were selected. Across both task lists and within different levels of the task lists, time to achieve effect and percent targets prosecuted were the primary measures. Additionally, metric M7 of AFT 2.1.1 is interested in the cost to perform precision engagement while M4 of AFT 2.1.3 measures the cost to equip forces with precision engagement capability. These two costs are operation cost and acquisition cost respectively, and represent the two cost metrics that will be tracked during this investigation.

This first step of problem definition corresponds to the functional area analysis. Rather than performing a unique FAA, it was shown how the information required for an FAA can be gathered for a known area of interest to the Air Force. The next set of steps will pertain to the functional needs analysis beginning with the development of an analysis environment.

5.2 Step II: Create Analysis Environment

5.2.1 Introduction

In this section, the agent-based modeling and simulation environment shall be described in detail. There are two main goals of this section. First, a clear understanding of model architecture, code modules, and assumptions is essential for a rich and thorough understanding of the experiments and results that follow. Secondly, providing guidance and detail on the analysis procedure enables future doctrine-materiel studies to readily build off of the results obtained herein.

This section will read in many ways as a user manual for a piece of software. The simulation overview describes the inputs, outputs, and most importantly, the analytical means for converting between the two. First, details will be provided about the simulation tool, NetLogo which was chosen for this research. Second, an overall architecture of the analysis code will be provided. Next, each module of code within the architecture will be elaborated and explained including linkages to other code elements. Finally, the wrapper which automates multiple runs of the code will be discussed.

5.2.2 NetLogo Overview

Earlier it was decided that the best simulation environment to employ was NetLogo, developed by the Center for Connected Learning and Computer-Based Modeling at Northwestern University. All simulations within NetLogo were executed with version 4.0.2, released December 2007. NetLogo is flexible program which the user can customize and is most applicable for problems involving complex systems. By instantiating entities in the simulation with properties and behaviors, models can be developed that yield new insights into a wide variety of problems.

Some examples of the classes of problems NetLogo has been used for include: biological systems such as predator-prey, social science problems such as spreading disease or traffic congestion, and many others. In each of these cases, as well as the research performed herein, entities called “agents” were created in the simulation and were given simple behaviors. It is then possible to observe patterns or performance over time based on the simple interactions of all the different agent behaviors.

One of the aspects of NetLogo that makes it so powerful is the flexibility in the type of problem that can be solved. While there is a great deal of inherent capability in NetLogo, the programming environment is essentially a blank slate, permitting a high degree of customizability. However, one caution is that since the environment is blank, the user must create all of the instructions, behaviors, and entities needed for the problem at hand.

A NetLogo environment starts out with a simple window displaying the user interface as well as a command line and several tabs and buttons which are used to construct the model. Figure 19 below is a screen-shot of the view the user sees upon first opening the program. The black area occupying the majority of the view represents the “world” for a particular model and provides a visualization capability to the events that take place during the simulation. There is a command center beneath the NetLogo world which allows the user to issue commands or queries to the environment. In the empty white space around the “world” it is possible to place input GUI’s for variables or output displays such as charts. The area above the world display contains menus and buttons for customizing the interface including: defining the properties of the world, controlling the simulation/visualization speed, and adding variables or outputs. Throughout the

remainder of this section and the in the results section later in the document there will be screenshots of this interface illustrating the various aspects of the Next Generation UAS model to the reader.

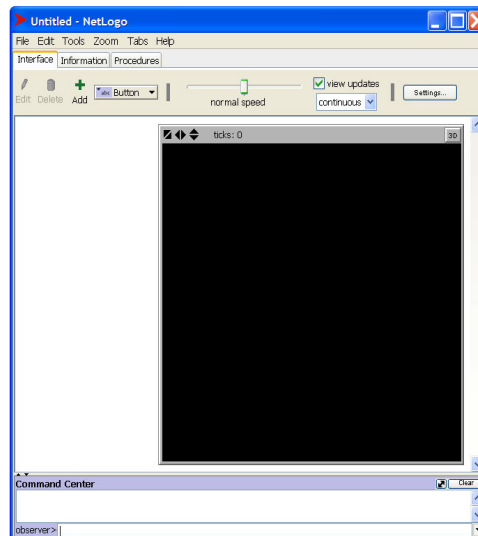


Figure 19: Basic NetLogo Display

Thus far, the elements of the graphical interface have been explained. However, note that in Figure 19 there are two additional tabs just above the visualization of the simulation world. The leftmost tab is the interface tab. The middle tab, ‘Information’, provides a location for the developer to enter comments and descriptions of the code. Finally, the third tab, ‘Procedures’ is the area where the code for describing the model and executing the simulation is entered. As was mentioned above, this area is empty when NetLogo is first opened but can be modified for whatever needs the user has.

Please note that the above discussion of the NetLogo simulation environment is not meant to be an exhaustive tutorial for the program; there is documentation to that effect in the program. The description of the general environment is meant to aide in the

understanding of subsequent explanations or illustrations involving the NetLogo program. Furthermore, such a discussion will assist the unfamiliar user with the basic knowledge needed to examine the results presented in this dissertation on their own.

5.2.3 Model Overview

Now that the program used for simulation in this study has been explained it is possible to detail the elements of the model that was created. The first step along this path is to present a complete overview of *all* the modules contained in the model. This will outline the proverbial forest for the reader before examining each of the trees, so to speak.

A complete diagram of the code structure is shown below in Figure 20. The main file is titled ‘doctrine.nlogo’ and is represented by the topmost blue rectangle. There are three other files that complement the main file and they are ‘allpatterns.nls’, ‘divide.nls’, and ‘targetprocedures.nls’. Note that there is only one file with an extension ‘.nlogo’. This file is the primary executing file and the other ‘.nls’ files are included within it.

Written within ‘doctrine.nlogo’ are the main parameters and behaviors needed to execute the simulation. There are seven categories of code in this file and each category is represented in Figure 20 by a red oval. First, there is a segment of initialization or setup code that lays the foundation for all the other instructions. Secondly there is an execution block of code that calls and directs all the remaining modules in the model. The remaining five categories of code that are created by the setup commands and run by the executable commands are: code for the hunter aircraft, killer aircraft, weaponry, controllers, and targets. Note in Figure 20 how the five types of entities are created by the setup commands and then feed the execution of the code. All of this takes place within

‘doctrine.nlogo’ which is denoted by the dashed line. Finally, outside of the dashed line, the three ‘.nls’ files are shown supporting their respective modules of code within the main file.

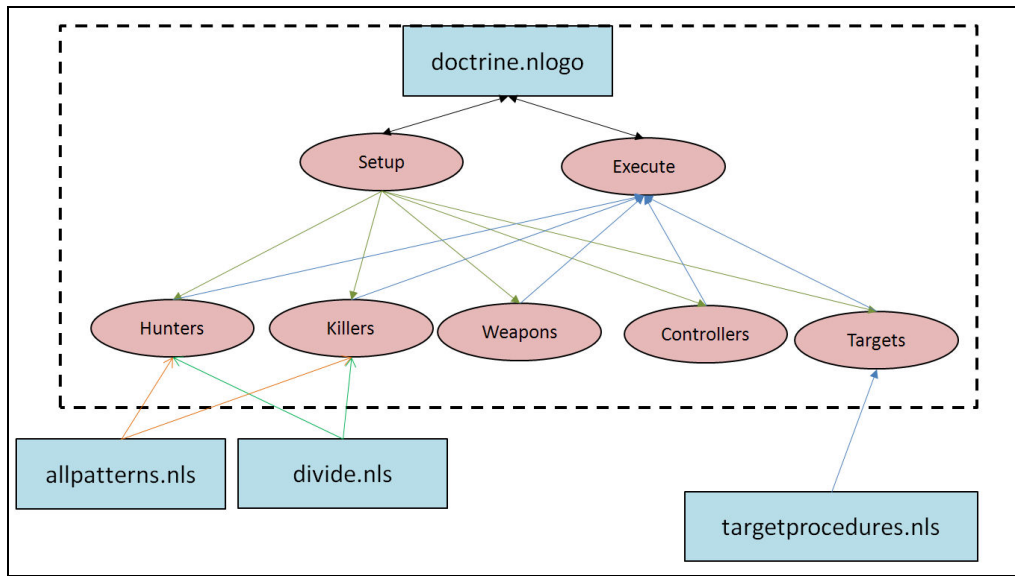


Figure 20: Simulation Architecture

With all of the boxes and arrows in the above figure, it may appear that the code structure is complicated. In actuality, it is quite simple. Only four files are used to execute all of the simulations described by this research. The rationale behind creating three supporting files was a matter of streamlining. Some of the code contained in the ‘.nls’ files is quite detailed and given that the commands in these files only interact with selected elements of the simulation, this architecture is efficient to run and to develop. Now that the NetLogo program and the overarching structure have been presented, it is appropriate to detail each of the elements in the model.

5.2.4 Modules and Procedures of M&S Environment

First, within the realm of initializing the code there are two distinct portions: definition and setup. Definition takes place at the very start of the code, before any procedures are run. It is here that the classes of agents are defined along with the variables that each agent class will have. Creating these classes, also known as breeds in NetLogo, involves defining the name and corresponding variables but not yet assigning values. For example, a block of code such as

```
breed [hunters hunter]  
hunters-own [fuel objective numWeapons]
```

defines the class, or breed of hunters in which a single agent is a hunter and each agent has properties of fuel, objective, and number of weapons. Complete sections of code are provided in Appendix D. However, this brief example illustrates basic definition procedures that apply to all of the classes of agents and how variables such as those being investigated in this research can be assigned to an agent class.

There are a few other points to make about definition. In addition to defining variables for each agent class, several global variables are defined including the metrics listed in earlier chapters. Also, this is the part of the file where references to include other files must be made, it cannot take place later in the model.

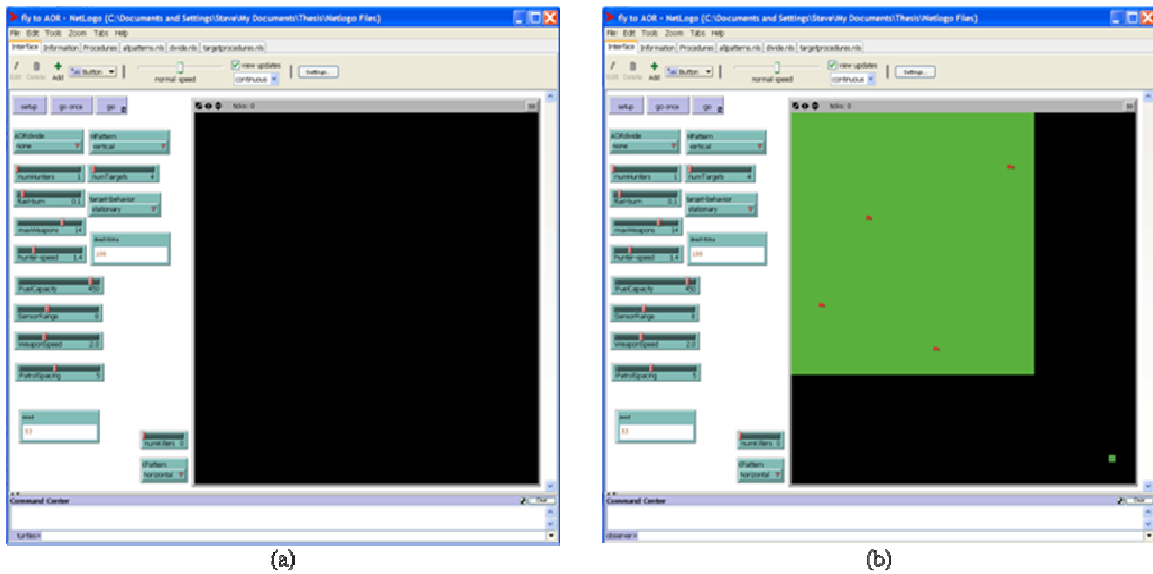


Figure 21: Setting up the Simulation Environment

Now that the definition has been completed, the other portion of initialization is setup. There is a procedure coded in the doctrine.nlogo file called ‘Setup’ and this procedure executes all of the code necessary to prepare the environment for a simulation. In Figure 21 (a) above, a NetLogo screen capture shows a blank world that differs from the previous example in that a series of variables occupies the left-hand portion of the window. When the Setup procedure is called, these variables are read into the program and a sequence of other procedures constructs the modeling environment. Figure 21 (a) shows the program prior to Setup while Figure 21 (b) shows the environment after Setup. Specifically, the procedures called within Setup are discussed in the list below.

- Clear – Removes information from any previous runs
- Random seed – Sets the random number seed so the user can repeat “random” cases

- Create airbase – The airbase is the small green dot in the simulation world of Figure 21, in the lower-right hand corner. This is the location where all blue aircraft stage from and where they must return in order to refuel and rearm.
- Create area-of-operations – The area of operations is the large green square in the simulation world. This area contains the targets and represents the area that the blue aircraft must perform their missions in.
- Make controllers – The controller, called breed CAOC (short for combined air operations center) in the model, is the entity that receives information from the aircraft and assigns targets. There is only one controller created currently and it resides at the same location as the airbase.
- Make hunters – Setup reads in the number of hunter aircraft specified in the input variables along with other parameters such as speed or fuel capacity and creates the appropriate number of hunters and places them at the airbase. At this point, a shape is provided for the hunter aircraft. Also, a set of internal variables are defined that will be used during the aircraft’s mission. Finally, during the creation of the hunters, Setup calls one of the supporting files, ‘divide.nls’ which is used to divide the area of operations into smaller segments for the corresponding number of hunters. This division process is one of the key doctrinal variables and will be explained in more detail later.
- Make killers – If the doctrine calls for multiple aircraft such that some vehicles search while others destroy, then Setup will create the killer aircraft. The process for doing so is virtually identical to that for the hunter aircraft. Key differences in

the two aircraft emerge during the operations and will be discussed in a later section.

- Make targets – The targets, visually represented as trucks in the NetLogo display, are created such that they are located randomly throughout the area of operations. There are two distinct scenarios that the Setup function must take into account. For the basic mission, the targets are fixed so the initial setup completely defines the targets. For the more difficult scenario, the targets appear at random times, move, and disappear some time later. In this case, setup defines when the targets will appear and where then stores those values in a list to be read later during the execution of the model.
- Create responses – Setup creates global variables that will be used to compute metrics that will determine the cost and effectiveness of the candidate approaches.

The second main procedure in ‘doctrine.nlogo’ is the executable procedure, called simply ‘Go’. When this procedure is called, all of the agents will execute their respective commands in sequence and the scenario advances one time step. NetLogo offers two options for calling a procedure: a single call or what is referred to as a “forever” call. The forever call means that the procedure runs automatically at every time step and will continue to do so repeatedly unless stopping criteria are specified. In the case of this model there are two stopping criteria. First, if all of the targets in the scenario are destroyed the simulation stops. Secondly, if the simulation hits 2000 time steps then the code stops. This second criterion was created to prevent unexpected behaviors from locking the simulation in an infinite loop. The Go procedure is used to march the

simulation forward and calls the entities created by the Setup function. The next step is to examine each of the classes of entities and describe how they were constructed.

First among the classes of entities are the hunter aircraft. The name ‘hunter’ is partially misleading in that when the doctrine calls for only one type of aircraft, the hunter class is really a hunter/killer class. The simplest way to describe the functionality and construction of the code is to initially treat this as a hunter/killer and then later describe how the functions of the hunter/killer system can be divided between hunter aircraft and killer aircraft.

The main hunter function is called ‘fly’ and directs the flight of the aircraft through each of the stages of the mission. For the purposes of this code, there are 5 distinct mission stages and the code specifies which stage the aircraft is in at any given time. The first stage is encapsulated in a function called ‘ingress’. As the name implies, this function guides the aircraft from base to the area of operation. The specific location within the area of operation was determined during the setup procedure with the ‘divide’ function. Each aircraft orients their self towards the ingress point and moves forward per unit time according to their speed. When the ingress point is reached, the code indicates that the particular aircraft should enter stage 2, patrol.

During the patrol stage, the aircraft executes the patrol pattern it has been assigned and searches for targets. The search procedure is straightforward. A sensor range is input from the main interface screen as is a sensor angle or sensor field of view. For example, if the sensor range is 3 and the sensor angle is 90 degrees, then the aircraft will detect any target within the cone 45 degrees off the nose in either direction and out to 3 units of distance. A diagram of this is shown below in Figure 22, where the blue cone

represents the sensor field of view and the dashed grid lines represent units of distance in the area of operation.

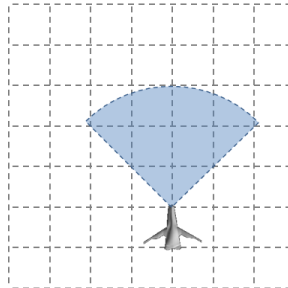


Figure 22: Sensor Diagram

If a target appears within the sensor field of view, then that information is passed to the controller and assigned to a prosecuting asset. In the hunter/killer case, the aircraft that found the target will automatically be assigned to engage however the allocation is more complicated when there are both hunter aircraft and killer aircraft and will be discussed later. Once an aircraft has found a target and been assigned to prosecute it, then that asset enters stage 3, prosecution.

The goal of the prosecution stage is clearly to engage the target and successfully deliver weapons. This stage of the mission begins with the hunter agent determining if the target is within weapon range. If so, then the aircraft launches a weapon immediately. Within the code, the launch process takes place by “hatching” a missile from the hunter. The hatch command in NetLogo creates a new agent that takes certain properties from the parent agent. In this particular case, the parent is the shooting aircraft and the weapon takes the values of location, heading, and objective from the hunter. When the weapon is dispensed the aircraft’s state is changed from searching to engaged. This means that the

aircraft cannot search or fire upon another target. Also at this point, two counters are activated. One counter is designed to keep track of the total number of weapons fired by all hunters and the other tracks the number of weapons still being carried by the aircraft. If the aircraft has zero weapons remaining it will enter stage 4 and return to base, which will be discussed later.

Once the weapon is released it begins to fly towards the target. The range and speed of the weapon are specified as input variables by the user. Each time step the weapon moves forward a distance equivalent to its speed. When the weapon reaches the target, the weapon kills the target and both entities are removed from the simulation. At this point, another counter in the program tracks that another target has been killed. Also, there is a counter which tracks not only how many targets have been killed, but the number of targets killed plus the number of targets which have finished their movement and hiding procedure. It is this counter which serves as the stopping criteria mentioned above in the 'Go' procedure section.

If the target is not within the range of the weapon, then the agent activates its 'pursue' function. This function orients the hunter towards the target and tries to chase the target down to the point where the weapon is within range of the target. When that occurs, all of the events mentioned above, including weapon launch, change in aircraft state, and activation of counters, take place.

Once the target has been destroyed, provided there are still weapons remaining, the aircraft resumes its patrol in stage 2. The process of searching, engaging, and patrolling continues while there are targets and hunters still in the area of operations. Throughout the hunter patrols and pursuits the aircraft is burning fuel. The total fuel capacity and fuel

burn rate are specified as input variables. During the flight, the aircraft checks its current fuel state against the fuel required to return to base. The aircraft enters stage 4, egress, according to the relation shown below in Equation 1

$$fuel = d_{RTB} \times \frac{SFC}{V_{cruise}} \quad (1)$$

where,

fuel = amount of fuel remaining

d_{RTB} = the distance to return to base

SFC = specific fuel consumption

V_{cruise} = cruise speed.

The egress stage begins with the aircraft storing information related to its last location and last heading. This is so that after the aircraft has finished refueling, it can resume the patrol where it previously was. Then the aircraft orients itself towards the airbase and flies until it reaches the base, burning fuel along the way but not engaging any targets. Once the airbase has been reached, the aircraft enters stage 5, refueling.

When the aircraft reaches the airbase, the first action is to restore the fuel level to full capacity. Note that there is no limit to how much fuel is at the airbase. Similarly, if the aircraft has less than the full complement of missiles, it is fully rearmed assuming that there are always more weapons available at the airbase. Also during refueling, a counter is activated that keeps a record of the total amount of fuel dispensed which will be used as a measure of cost.

In summary, the hunter aircraft are created by the Setup function. During the execution of the ‘Go’ function, the aircraft ingress to their area, patrol until a target is found, prosecute that target, iterate the hunt and kill segments until fuel is low or weapons are expended and then egress to base to refuel and rearm. The process repeats while the stopping criteria are not yet fulfilled. A pictorial representation of the algorithm is shown below in Figure 23.

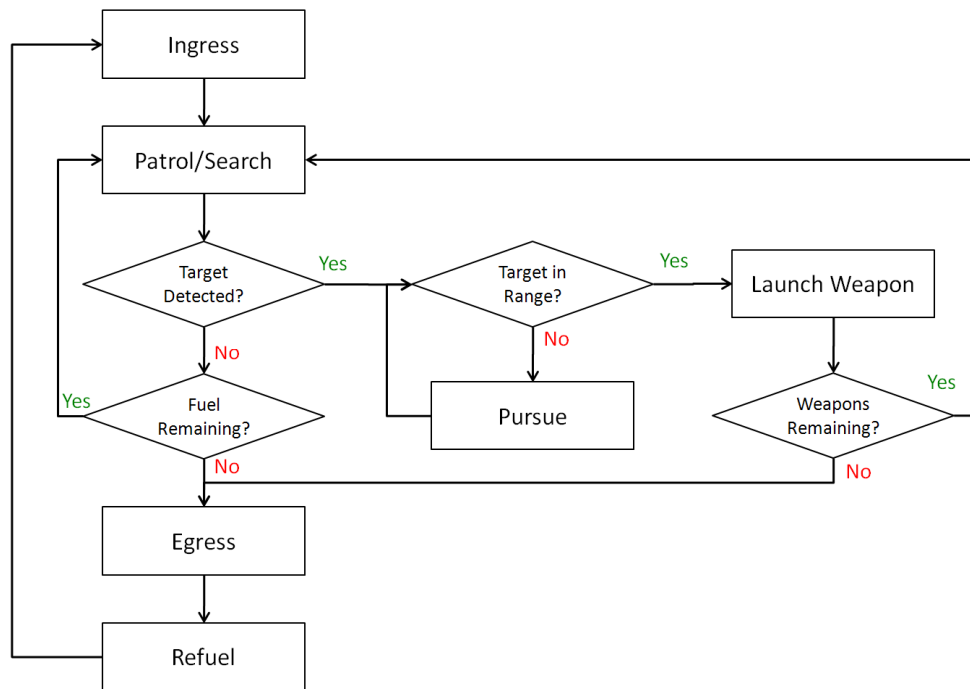


Figure 23: Diagram of the Hunter Algorithm

Now that procedures for the aircraft agents have been described, the next class of entities in the simulation to address is the target class. Targets, represented in the simulation as red vehicles, are the objective of the blue force and successfully destroying the targets is how a successful mission is achieved. There are several different target

behaviors that are modeled in this analysis. The purpose of the different behaviors is to make the mission more challenging thereby resulting in a capability gap and therefore necessitating the execution of a Capability-Based Assessment.

First, the targets can be stationary. This is the most basic target behavior. After the setup command at the beginning of the simulation creates the blue agents and the area of operation, a user supplied number of targets is generated. These targets, also called TCTs, or time-critical targets in the code, are randomly distributed throughout the area of operation. The information on their location is NOT supplied to the blue forces meaning that there is no prior intelligence about the location of these agents. The targets are created with a variable called “priority”. When the aircraft are searching and reporting the contacts back to the combined air operations center, there needs to be a mechanism for prioritizing the targets. At the simplest level, this priority is instantiated by recording the time of detection and the earlier a target is found, the higher its priority. Basic changes to include location information, target type, or threat level could also be included.

Secondly, the targets are capable of moving and hiding. In the previous behavior description, the targets were all created at the beginning of the simulation and stayed in their original location. For this behavior, during the setup procedure, the code randomly determines the “start time” (the time when an agent will be created) for every target. This information is stored in an array. Then, as the simulation progresses, at the specified times, a target creation algorithm generates a new target at a random location within the AOO.

Part of the behavior of these targets is that they do not stay exposed indefinitely. Rather, the user supplies a dwell time, or a length time that the target will remain visible,

at the start of the simulation. When the agent is created, it is assigned a variable called “die time” which is the time in the simulation when the agent will go back into hiding. The value of this time is simply equal to the start time plus the constant dwell time. It is important to note that once an agent has completed its dwell time it will not reappear. This agent behavior mimics that of several real world targets. Consider first a ballistic missile launcher. Once the system begins to setup to fire its weapon, there is a certain amount of time until the weapon is fired and after that time, it does not matter if you kill the launcher since the missile is already gone. Another example could be a terrorist in an urban environment. The dwell time could represent the amount of time it takes a terrorist to transit from a safehouse to a hospital or a cave. Once the terrorist is hidden or protected, it is no longer possible to prosecute him.

The other aspect of the limited dwell time targets is the ability to move. The movement commands are located in the ‘targetprocedures.nls’ file included in the overall ‘doctrine.nlogo’ file. A procedure called “evade” moves the targets in a random path around the area of operations. Similar to the random-walk procedure for the hunter agents, the evade procedure randomly determines how often and in what direction the agent shall deviate from its current path. The agent does not try to specifically evade any hunter asset.

The next type of agent possible in the simulation is the killer class of blue aircraft. This class of agents contains the same set of variables as the killer aircraft including their own speed, fuel burn, number of weapons, and so forth. Additionally, the killer aircraft can employ the same patrol patterns and be divided according to the same algorithms as the hunter aircraft.

However, when aircraft designated as killers are used in the simulation there are several key differences to the functionality of a single class of hunter/killers. First, where before the single aircraft type carried sensors and weapons, now the hunter aircraft carry sensors while the killer aircraft carry weapons. Therefore, when a hunter aircraft senses a target, it will not pursue and it will not engage. Rather the information will be eventually fed to the killer aircraft to carry out the prosecution of the mission. In order to communicate the information between hunters and killers there is one more key step required: the air controller.

The final type of agent in the simulation is the combined air operations center, or CAOC. Represented as a breed of agent in NetLogo, the CAOC receives information from the various blue assets about which TCTs were found and the priority assigned to them. For the case where there are only hunter/killer aircraft in the simulation, the queuing performed by the CAOC is trivial as each aircraft is assigned to attack the target it just found. However, for the case where there are both hunters and killers, the CAOC performs a queuing procedure.

First, when a new target is identified, the hunter aircraft forwards this information to the controller where it goes into a list containing the names and priorities of each target. Then, the CAOC will identify which killer asset is closest to the target without already being engaged with another target. Once the assignment is passed on to the killer, the CAOC removes that entry from its list of targets. If there is no killer aircraft available when the CAOC receives the information from the hunter aircraft, then the controller stores that information and will dispense orders to killer aircraft as they become available according to target priority.

The preceding section detailed the procedures in the analysis environment being used in the study of doctrine and materiel changes. However, the challenge now lies in finding a way to automate the process of executing cases. There were several options for running the necessary cases. First of all, NetLogo possesses an inherent capability to run variables at multiple settings. The benefit is that all the analysis happens inside NetLogo resulting in a very simple execution. However, NetLogo only runs full-factorial sets. That is to say, if there are 20 variables to investigate, each at 3 levels, NetLogo would try to run all 3 billion possible combinations! Clearly this is impossible and inefficient so another way was needed.

Another option for automating the process is to use the ModelCenter environment, developed by Phoenix Integration. ModelCenter is specifically designed to wrap other analysis codes and provides an intuitive graphical interface to facilitate the execution of those analyses. The drawback to the ModelCenter approach is that additional coding would be needed both in ModelCenter and NetLogo. However, the ability to quickly execute a design of experiments through this environment as well as the accessibility of the program made ModelCenter an ideal choice for this problem.

The basic premise for implementing this wrapper is straightforward. Figure 24 below depicts the simulation architecture. Comparing this diagram to Figure 20 from earlier, the elements of the NetLogo model, enclosed in the dashed box, are all the same and have the same linkages. However, there is now an analysis wrapper which reads in sets of cases, interfaces with NetLogo, and once a case is completed, stores the results.

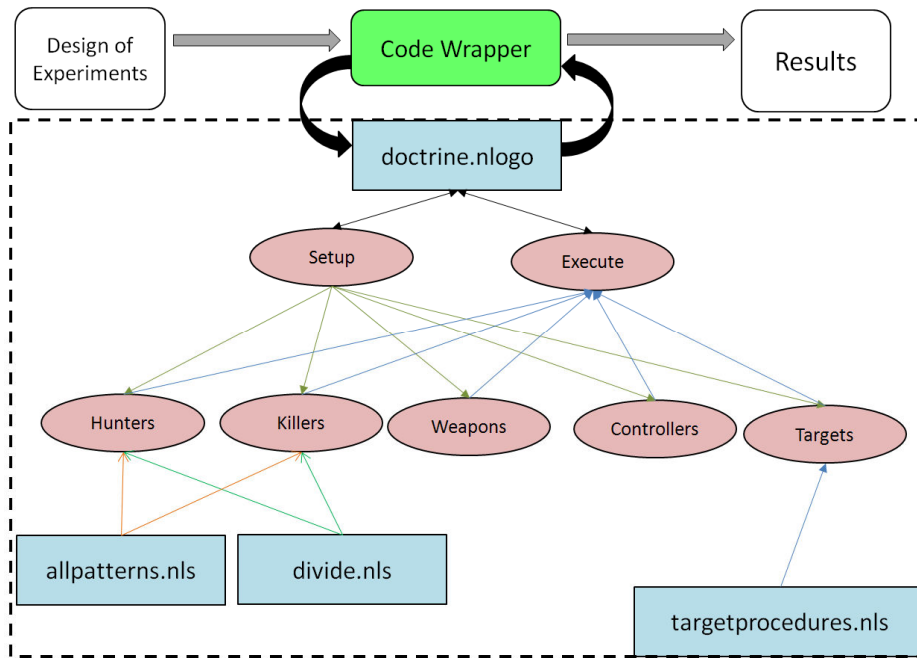


Figure 24: Analysis Code with Wrapper

5.3 Step III: Baseline Analysis

With the analysis framework in place, the next step in the process is to perform a baseline analysis to ascertain if a capability gap exists. For the baseline analysis, the simple mission of finding fixed targets in an area of operations will first be analyzed. A listing of variables and their default values are presented below in Table 3.

Table 3: Input Variables for Baseline Analysis

Variable	Value
Number of Aircraft	4
Aircraft Speed	0.4
Sensor Range	5
Sensor Angle	90
Fuel Capacity	100
Fuel Burn	0.1
Number of Weapons	2
Weapon Range	5
Weapon Speed	2
Patrol Pattern	Parallel
AOO Division	Basic

These default values represent the notional MQ-1 Predator aircraft. It is worth reiterating that there is no classified or otherwise sensitive information in this study and these values are only meant to represent a relative estimation of performance within the example scenario.

Recall from Step II that the creation of the scenario in the simulation is done through a partly random process. For this baseline experiment, it was decided to test the performance of the system against 100 randomly generated configurations of targets. In each instance there will be exactly 10 targets and they will be fixed. Testing this many cases will create greater confidence that there is no bias due to target placement.

The 100 simulations were executed in NetLogo and the number of targets killed was output. Below in Table 4 is a table showing the possible number of targets killed in the simulation along with the number of times that outcome occurred. Eighty-five times out of 100, all of the targets in the scenario were destroyed. In only 15 instances did the time limit halt the simulation prior to completing the mission.

Table 4: Frequency of Targets Killed in Baseline Scenario

<i>Targets Killed</i>	<i>Frequency</i>
0	0
1	0
2	0
3	0
4	0
5	0
6	0
7	1
8	5
9	9
10	85
Total	100

This analysis shows that the current system performs well for its current mission. The scenario is not too demanding for the asset and under normal conditions, the system is sufficient to achieve the desired capability.

5.4 Step IV: Gap Analysis

Now that the baseline analysis has been performed, it is time to examine the more difficult time-critical-target mission to see if a gap exists. Additionally, in the case where a gap does exist, it is also important to quantify the extent of that capability gap. This is an important point and a central theme of this method. Obtaining a quantitative understanding is an important requirement for a capability-based assessment. Characterizing the severity of the gap, both in terms of performance shortfall and importance, will provide information that will guide the decision-making process once all of the approaches have been assessed. Additionally, when making acquisition decisions, it is important to identify exactly how much added performance a given system could provide. Therefore, quantifying the capability gap through the use of modeling and simulation is a vital step in the CBA process.

The new advanced scenario with targets capable of moving and hiding was simulated in a manner consistent with the baseline scenario described in the previous step. The same set of input variables for the UAS that appear in Table 3 were used for this scenario. Also, 100 replications were run to test a variety of target configurations. Just as in the previous simulation, the number of targets killed was tracked as an output. The results of this set of simulations are presented below in Table 5 with the possible values of targets killed in the left-hand column and the number of times that outcome was achieved in the right-hand column.

Table 5: Frequency of Targets Killed in Advanced Scenario

<i>Targets Killed</i>	<i>Frequency</i>
0	1
1	3
2	16
3	20
4	37
5	12
6	9
7	1
8	0
9	1
10	0
Total	100

This table shows that the performance of the baseline system in an advanced mission is significantly deficient when compared to the baseline scenario. The most common outcome from these 100 simulations is 4 targets killed. There are only two instances where the number of targets killed is greater than 6 and no cases where all 10 targets were prosecuted. On the other hand, there was also only one case where no targets were destroyed. While not up to the standard necessary for successful completion of this mission, this does indicate that the mission is one that a similar UAS would be well suited for if different approaches were considered.

To compare the two scenarios side-by-side refer to Figure 25. This chart is a histogram plotting the frequency of occurrence versus the number of targets killed. The visual contrast between the two scenarios is immediately apparent. The blue bars represent the baseline mission while the red bars represent the mission with targets capable of moving and hiding.

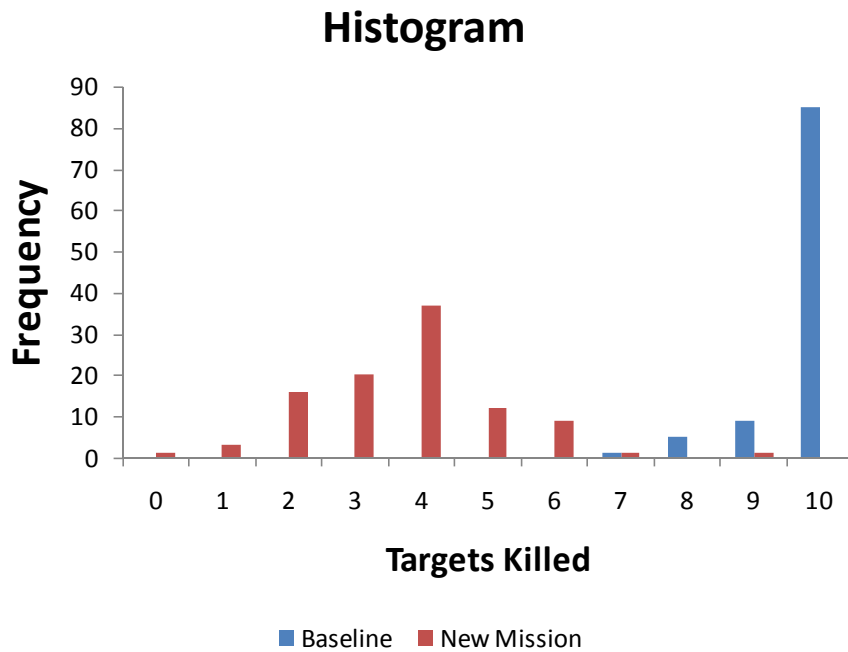


Figure 25: Performance Histogram for Gap Analysis

Finally, to obtain a firm quantification of the capability gap rather than a visual qualitative feel, a statistical analysis was performed on both sets of simulation results. The results of this comparison are shown below in Table 6.

Table 6: Statistics of Performance Gap

	Baseline	New Mission
Mean	9.78	3.73
Std. Dev	0.58	1.43
Max	10	9
Min	7	0

The average value of targets killed in the advanced mission is 6 fewer aircraft per mission. The minimum value of targets successfully killed in the baseline scenario is seven and even this minimum is three targets better than the average number of targets

killed in the moving-hiding mission. Clearly there is a performance gap in the current system capability driven by an increase in mission difficulty. However, more than just indicating the presence of a capability gap, it is now possible to quantify that gap.

The capability gap will be measured in accordance with the selected metrics from Step I: Problem Definition. Recall that the metric for this mission was derived from the Universal Joint Task List and the measure is percent targets killed. Expressing the results of this baseline and gap analysis in terms of percent, it can be stated that for the baseline mission, the current UAS prosecutes 98% of targets. Furthermore, for the advanced mission, the current UAS prosecutes 37% of targets. Therefore it can be stated based on the results of these analyses that there is a 61% capability gap for percent targets killed relative to the current average performance.

The quantification of capability gaps is an important element but this step does not end there. In order to feed the next stage of analysis, it is necessary not only to provide the extent of the capability gap but also any insight about why this gap is occurring. The reason that this additional element of the step is so important can be found by revisiting the CBA process. Steps II-IV fall into the Functional Needs Analysis portion of the capability-based assessment. In addition to answering the question of “How successful are we today?”, the FNA must feed the Functional Solution Analysis. The FSA must generate new possible approaches and then assess those approaches for relative goodness towards satisfying the capability gaps found in Step IV and meeting any other requirements identified in Step I. In order to best frame the set of possible approaches, there should be an analysis of why the capability gap exists. This will provide insight as

to which approaches should be considered, how the design space should be bounded, and so forth.

So now the issue is what could be possible causes for this 61% gap in performance? Given that this is a hunter/killer mission taking place within a constrained time frame, in order for the mission to be successful, the system must quickly find and then prosecute any and all targets. Therefore, there could be a deficiency in either the finding or the prosecuting section of the mission. For the find portion, success ultimately comes from maintaining constant or frequent coverage over any particular area. As for prosecuting the targets, that is ultimately a question of reacting to information and delivering weaponry.

5.5 Step V: Identify Materiel/Non-Materiel Approaches

The identification of materiel/non-materiel approaches is the step where, assuming a gap has been found, the design space for the upcoming analysis is defined. This is an important step that bounds the functional solution analysis.

There are several inputs to the identification of approaches, including the tasks and metrics outlined in Step I. More directly, the identification of approaches should take the gap analysis results and decompose that information into elements which can then be used to form the approaches. What this decomposition means is, identify the functions which must be performed in the mission, use the gap information and baseline analysis to make assumptions about which functions are deficient, and then find alternative means of achieving that function.

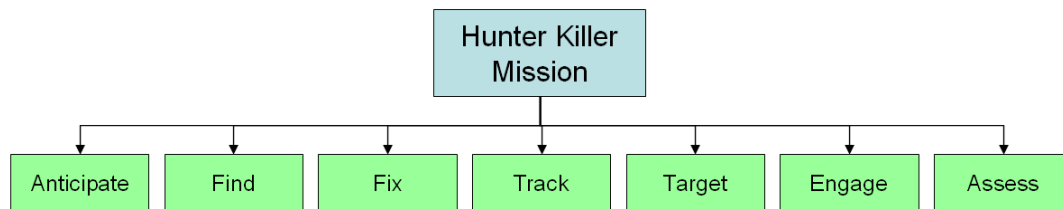


Figure 26: Functional Decomposition of Mission

For example, consider the decomposition shown in Figure 26. The overall hunter killer mission is first broken down into the steps of the kill chain. The system must anticipate that a target exists, then find, fix, track, target, and engage the enemy finally assessing the success or failure of the mission. Based on the gap analysis, the first area of concern is finding targets.

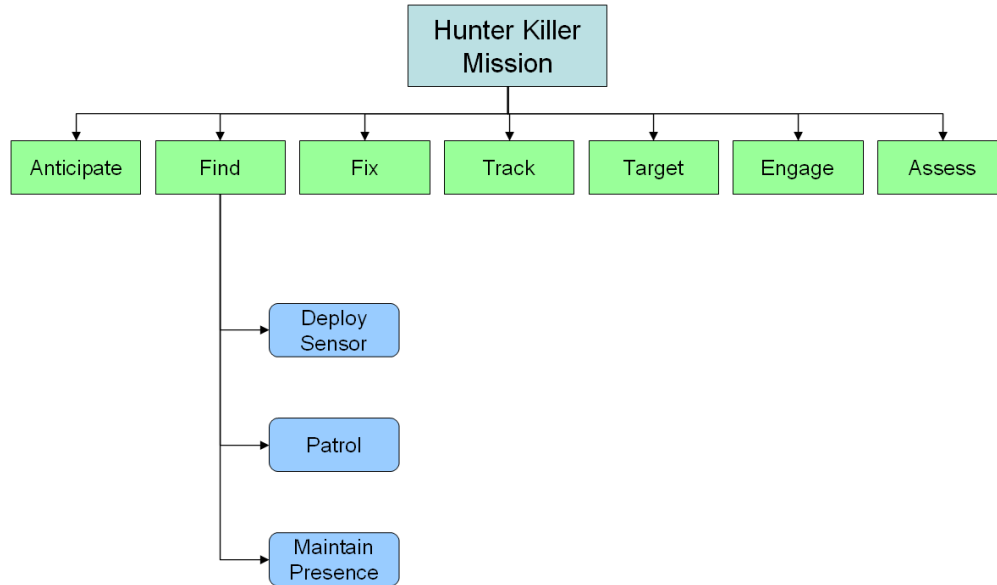


Figure 27: Functional Decomposition of Find

Decomposing the ‘find’ function into sub-functions results in a hierarchy like the one shown in Figure 27. To find the targets, a unit must deploy its sensor, patrol an area, and maintain a presence in that area. Next, each of these sub-functions can be further decomposed. The first example is show in Figure 28.

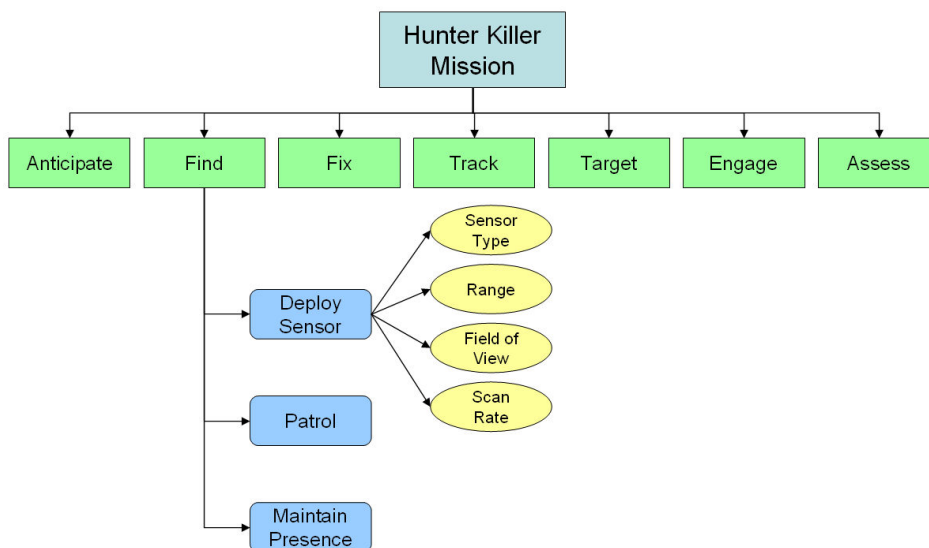


Figure 28: Functional Decomposition of Deploy Sensor

The sensor deployment depends on the type of sensor deployed, the range that sensor can see, the field of view angle, the rate at which the sensor covers area, and possibly others. From this decomposition it is possible to identify new approaches. For example, changing the sensor range affects how the sensor is deployed as well as the area covered, which in turn influences the find function of the kill chain and finally the effect propagates up to overall mission success.

Another example is shown in Figure 29. In this example, the patrol function was further decomposed into the type of patrol pattern, the speed of the patrol, and the area boundaries within which the patrol takes place. Changing the speed or the patrol pattern affects the patrol function, in turn the find function, and eventually the mission success. It is important to note that the speed of the asset is a materiel variable while the patrol pattern is a doctrinal variable. Functional decomposition is powerful because the basic elements of the higher level functions can be materiel or non-materiel.

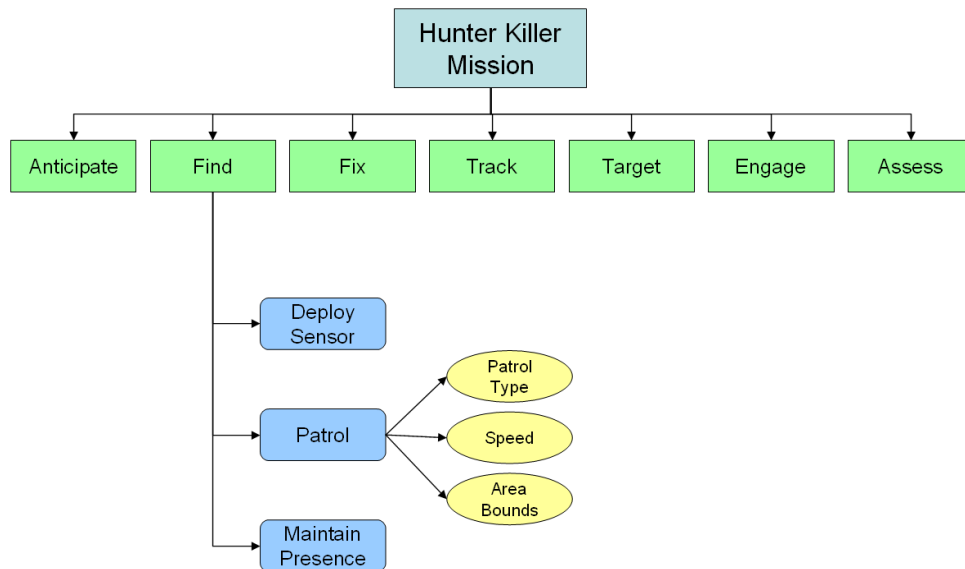


Figure 29: Functional Decomposition of Patrol

Similarly to the above examples, additional decompositions were performed on Anticipate, Find, and Engage. The logic behind the decompositions are discussed below. After those discussions, a listing of the variables and approach options selected will be presented.

Another aspect to finding targets is to maintain presence where the target will be. Therefore, the unit must be in the same area as the target. To accomplish that, the unit logically cannot be at base. If the unit wants to minimize its time at base, then it must not run out of fuel or weapons. To avoid running out of fuel, the unit must either carry a lot of fuel, or burn it slowly. Therefore, the number of weapons, fuel capacity, and fuel burn rate are all appropriate materiel variables.

Also, if the unit must cover ground with its sensor there are several ways for the system to cover more ground. First is to have more units in the fleet. Therefore number of aircraft will be a variable. Since the number of aircraft in the fleet does not define the physical properties of any particular aircraft and changing the number of aircraft does not require the defense acquisition system to generate a new materiel program, the number of aircraft is classified as a doctrinal variable.

Another way to cover more ground is to increase the speed of the platform. This will increase the amount of area covered per unit time and also increase the revisit rate to any given point which is an important property when the targets are moving and hiding. Also, by increasing the sensor footprint on the ground, more total area can be covered. At a top level, there are two ways to increase the sensor footprint. First is to increase the range that the sensor can see. Second is increasing the field of view, or the number of degrees that the sensor can see. Each of these parameters will be a materiel variable in the study.

The previous paragraph brought to light a key idea: that of revisit rate and efficient searching. There are a variety of search techniques prescribed by the military that can be used in different situations. Search pattern is definitely a tactic and an important one for a mission such as the one being modeled here, therefore patrol pattern will be a doctrinal variable. Related to the idea of patrol pattern is the concept of how much area should be patrolled by each asset. Since the number of assets is a doctrinal variable, the way those assets are allocated in the overall area of operations is also a doctrinal variable.

Assuming that the combination of assets and allocation has found the target, the next step is to actually prosecute the target. Herein lies an interesting question: who delivers the effect to the target? Should there be a single aircraft type acting as a hunter/killer platform or could the various functions be improved by dividing the activities into an aircraft designed to hunt and an aircraft designed to destroy. The choice about how many aircraft of which type to include in the fleet is another doctrine variable which will be considered.

Once the attacking asset has been chosen, the issue becomes, how well does the weapon prosecute the target? Two of the most important properties of the weapon are the speed and the range. Here, speed refers simply to the velocity with which the weapon flies towards its target. Weapon range refers to how far away from the target the weapon can be fired. These two parameters will be used as materiel variables in this study.

Theoretically, there is a nearly infinite set of variables that could be used to fully describe these scenarios. However, the objective is to develop and test a methodology for quantifying doctrine. The preceding discussion has highlighted 6 materiel variables and 6 doctrine variables. This number of variables should be sufficient to adequately

demonstrate the tradeoffs on a problem of reasonable scope without complicating the discussion of the methodology with too many degrees of freedom.

In order to fully articulate the design space a morphological matrix is used to list the variables and their possible values. Table 7 below is the matrix for this design study.

Table 7: Morphological Matrix of Approaches

		Option Number								
Variable	Units	1	2	3	4	5	6	7	8	
Doctrine	Patrol Pattern	N/A	Random	Vertical	Orbit	Box				
	Patrol Spacing	Distance	1	3	5	7				
	AOR Division	N/A	None	Entry Point	Strips	Boxes				
	# of Aircraft	#	1	2	3	4	5	6	7	8
Configuration	Hunter Type	N/A	MQ-1	MQ-9	RQ-4	F-16	F-35	New		
	Killer Type	N/A	None	MQ-1	MQ-9	F-16	F-35	New		
Materiel	Sensor Range	Distance	3	15	*					
	Sensor FoV	Degrees	60	180	*					
	Weapon Speed	Distance/Time	1	3	*					
	Cruise Speed	Distance/Time	0.4	1.6	*					
	Fuel Burn	1/Time	0.05	0.15	*					
	Fuel Capacity	#	100	400	*					

* = continuous

Each of the 6 doctrinal and 6 materiel design variables is presented in Table 7. For the materiel variables, the ranges of the variable are expressed in generic units and values. This was to highlight that the results are generic and will not represent any real system. Additionally, the default aircraft types are listed as configurations. These configurations represent a set of available aircraft from which the commander can choose to perform the mission. Selecting a different configuration or configurations addresses the fleet architecture and fleet sizing.

Table 8 below then shows the default values of these configurations with respect to the materiel properties. Again, the values are *notional* values for aircraft *based on, not equal to*, the configurations listed. The names of the aircraft are provided simply to give a

general reference for the types of assets that might be used and notional relative performance between the asset types.

Table 8: Notional Values for Aircraft Types

Aircraft Type	Speed	FuelCapacity	fuel-burn	maxWeapons	SensorRange
MQ-1	0.2	50	0.1	2	3
MQ-9	0.4	100	0.1	6	5
RQ-4	0.6	350	0.1	0	10
F-16	1	200	0.2	10	5
F-35	0.8	400	0.2	8	5

5.5.1 Definition of Variable Types

In Table 7 there are variables listed as either doctrine or materiel. However, it is important to specify in more detail exactly what kind of variables these are. A complete taxonomy of variable types is developed in Appendix B. It is important to describe the level of doctrine variable as well as identify if there is any confounding between variable types.

First, the doctrinal variables of patrol pattern and patrol spacing are tactical level variables because they address execution of asset level procedures. The area division, number of aircraft in the fleet, and configuration type(s) are all operational level doctrine variables. While none of these variables are confounded with each other, there are some assumptions behind them that should be noted. First, only one type of hunter aircraft and one type of killer aircraft (if applicable) can be used. That is to say, the commander is not able to use both “MQ-1 type” aircraft and “MQ-9” type aircraft for hunting. The other assumption is that the type of patrol pattern, division, or spacing applied to the class of

aircraft is used by all assets. For example, one hunter asset cannot use a random-walk patrol while another uses an orbit patrol.

With respect to the materiel variables, parameters such as the fuel burn and fuel capacity are not confounded with any doctrine. However, the cruise speed is a confounded variable. The speed at which the aircraft flies depends not only on the aircraft, but also on the doctrine employed at the time. An aircraft capable of flying at Mach 1 could theoretically fly at Mach 0.9 as well. The choice of speed employed during the mission is a doctrinal consideration. Therefore, the cruise speed is a materiel variable with the implicit assumption that the aircraft will always fly at maximum cruise speed.

5.5.2 Doctrinal Procedures in M&S Environment

Now that the doctrinal variables are known, the implementation in the NetLogo simulation will be discussed. First is the area of operation division, included in the divide.nls file. The basic question that this section of code tries to answer is: given that I have multiple aircraft, how do I divide them up to operate effectively in that area? There are four distinct division algorithms for the area of operations (AOO) employed in this analysis: None, Entry Division, Strips, and Boxes. The first three of these are presented below in Figure 30 while the more complex box division is presented later in Figure 31.

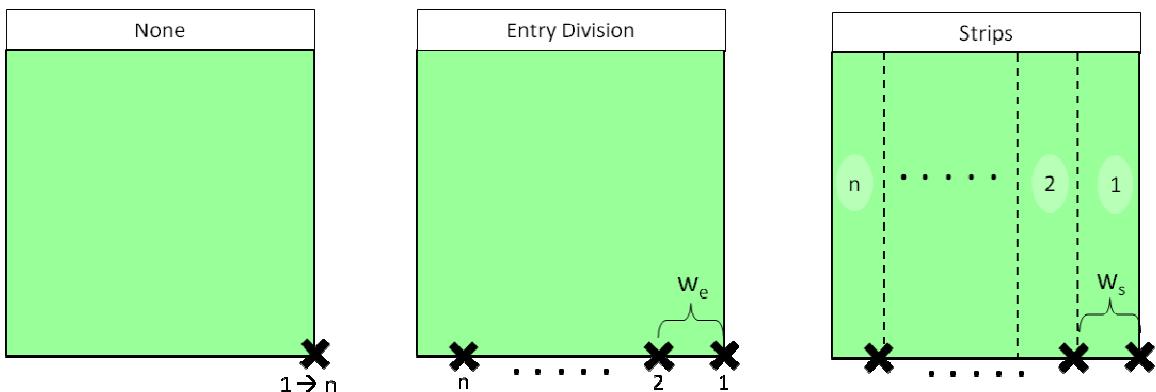


Figure 30: Three AOO Division Methods

First note that the light green area represents the AOO, the black x represents the entry point for a particular aircraft, and n is the total number of aircraft in the scenario. Examining the baseline condition, or ‘None’, it is clear that in this case, all of the aircraft have the same entry point and there is no sub-division within the larger area of operations. That is to say, all initially enter at the closest point to the base and can move anywhere in the area of operations. Over time, as aircraft expend weapons at a different rate or patrol different directions, they will reenter the AOO at different points following refueling. This is the simplest case of AOO division and represents the baseline.

Next, the middle graphic in Figure 30 depicts the next step forward in division of the area of operations, entry point division. The concept behind this technique is to still allow all aircraft to transit the entirety of the AOO yet stagger their entry points to maximize the coverage of the area. If there are n aircraft in the simulation, then the entry points for these aircraft are spaced out evenly along the bottom edge of the area of operations. The value for the spacing between the entry points is represented in the figure by W_e and is computed by the formula in Equation 2.

$$W_e = \text{ceiling}\left(\frac{(x_{\max} - x_{\min})}{n}\right) \quad (2)$$

where

n is the total number of aircraft

x_{\max} is the coordinate value of the right hand side of the AOO

x_{\min} is the coordinate value of the left hand side of the AOO.

The entry point for any particular aircraft is based on an equal number of W_e values and is computed using the formula in Equation 3.

$$x^* = (n - n^* - 1) \times W_e + x_{\min} \quad (3)$$

where

x^* is the entry coordinate of the current aircraft

n^* is the identifier of the current aircraft.

Two items to note about these equations. First, the ceiling operator is included so that the coordinate of the aircraft's ingress is a whole number. Secondly, the n^* term identifies the aircraft but in NetLogo, the convention of identifiers begins at 0. That is to say, if there are 5 hunters in the scenario, the set of their identifier tags are [0, 1, 2, 3, 4]. This identifier scheme is what mandates the inclusion of the n^*-1 term in Equation 3.

Finally, the last of the algorithms in Figure 30 is the Strips division technique. This technique takes the idea of spacing the entry points a step further and limits the area that an aircraft can patrol to a fraction of the total AOO. In Figure 30 these subdivisions are denoted by dashed lines. The spacing between entry points is marked by W_s and is the same value as W_e that was computed by Equation 2. The key difference between Strips and Entry division algorithms is that for the Strip division, the x coordinate of the entry point becomes the maximum coordinate that the aircraft can reach in that direction. The minimum coordinate is computed according to Equation 4 below.

$$x_{\min}^* = (n - n^*) \times W_s + x_{\min} \quad (4)$$

Note that unlike Equation 4, there is no n^*-1 term. This means that that lower bound, or left-most x-coordinate that an aircraft can fly to, is the same value as the maximum point of the next aircraft. This ensures complete coverage of the area of operations.

Finally, the last division algorithm is the Box divide method. The basic principle here is that as the number of aircraft gets larger, the strips become narrower and eventually they could become too narrow. Therefore, dividing the AOO in half horizontally and then assigning the aircraft similarly to the strip fashion above makes the individual areas more square, hence the name Box division. Additionally, it is possible that this configuration provides better spacing under other operating conditions.

For a graphical representation of the Box divide algorithm examine Figure 31. There are two graphics, the left hand representation applies when the number of hunters is an even number while the right hand representation applies for odd numbered fleet sizes.

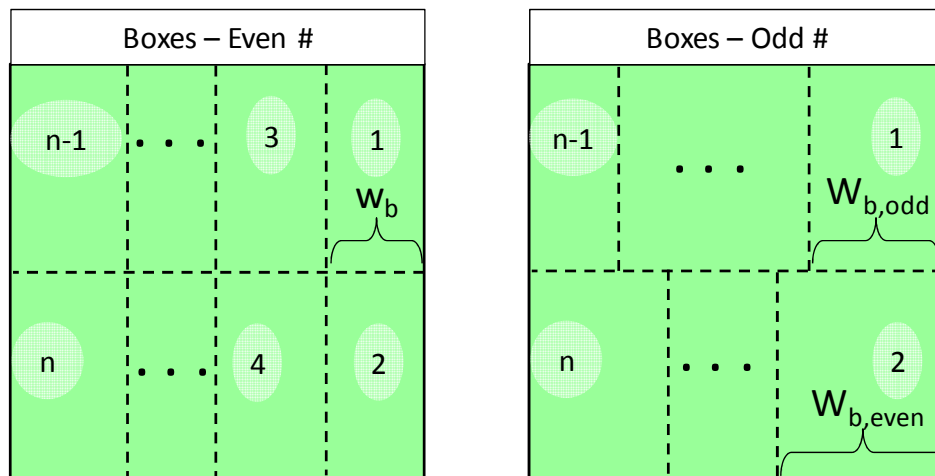


Figure 31: Box Division of AOO

The first step in creating the boxes is to divide the AOO in half across the centerline. This step takes place regardless of the number of aircraft. Then, for the case where n is an even number, the boundaries of the boxes are computed by the formula in Equation 5. The computation of the box width is very similar to the computation performed for the strip calculation with the only difference being that the total AOO width is divided by $n/2$ as opposed to n .

$$W_b = \text{ceiling}\left(\frac{(x_{\max} - x_{\min})}{\frac{n}{2}}\right) \quad (5)$$

Similarly, Equation 6 computes the entry coordinate for each agent in a manner analogous to the strip method.

$$x^* = \left(\frac{n}{2} - \text{floor}\left(\frac{n}{2}\right) - 1\right) \times W_e + x_{\min} \quad (6)$$

In the case where n is an odd number, the calculation entails an extra step since clearly the top and bottom of the AOO cannot have the same number of aircraft, and thus the width of the boxes must be different. The convention for this research is that the upper portion of the area of operation will have one more agent than the lower portion. The computations proceed in a manner similar to that where n is an even number.

Beyond the division of the area of opportunity, another key doctrine variable is the aircraft patrol pattern. There are four patrol patterns that an aircraft can employ: random walk, lawnmower, orbit, and border patrol. The last three of these patrols are

based on the TTP manual for CSAR missions while the random patrol pattern was provided to gauge the relative effectiveness of the different patrol patterns.

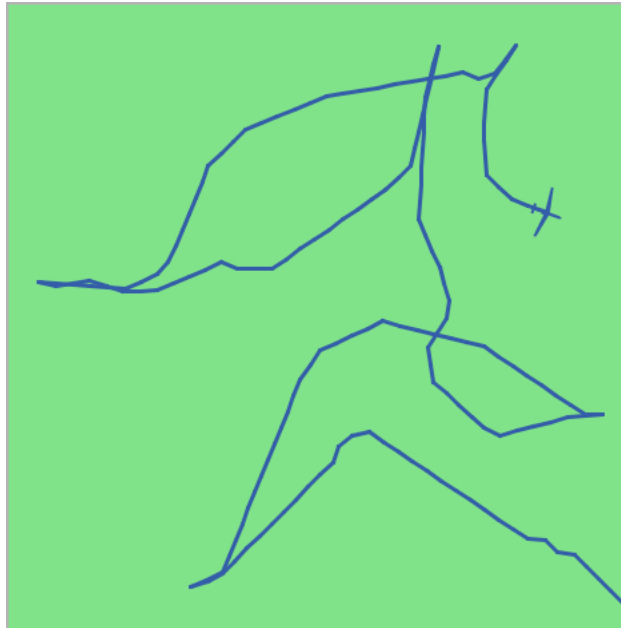


Figure 32: Random-Walk Patrol Pattern

First among the patrol patterns is the random-walk pattern. An example of this pattern is shown above in Figure 32. Here, there is a single aircraft moving within the entire AOO and the blue line traces the path of the aircraft. At each time step, the agent randomly determines if it shall change direction or continue on its current path. If the aircraft will be changing direction, another random draw determines the direction, left or right.

The second patrol pattern is the parallel search pattern, also sometimes referred to as the S-pattern or lawnmower pattern. Examining Figure 33 illustrates how this pattern got its names. The aircraft flies a straight leg and upon reaching the edge of its assigned area, turns 180 degrees and flies a leg in the opposite direction. One interesting note about the turn at the end of each leg is that the precise turning radius of the aircraft is not explicitly

calculated. Instead, the aircraft is assumed to be able to turn sufficiently tightly so that the return leg is directly adjacent to the down leg. Then the spacing between two legs becomes a doctrine variable.

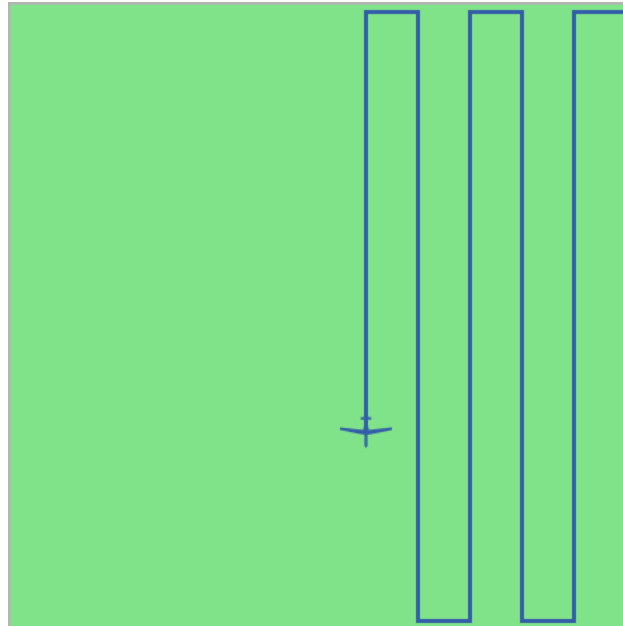


Figure 33: Parallel Search Pattern

Next, the orbit patrol pattern is designed to have the agent loiter about a point. The execution of this pattern begins by the aircraft flying into its patrol area and then computes the central point of that area. Once the aircraft reaches the center point, it flies out to the orbit radius and then performs four successive legs to form a box about the center point. The orbit radius is fed from the spacing doctrine variable that spaced the legs of the parallel search pattern. An example case of the orbit patrol pattern is shown below in Figure 34. Here there are four aircraft orbiting about the center points of their respective areas.

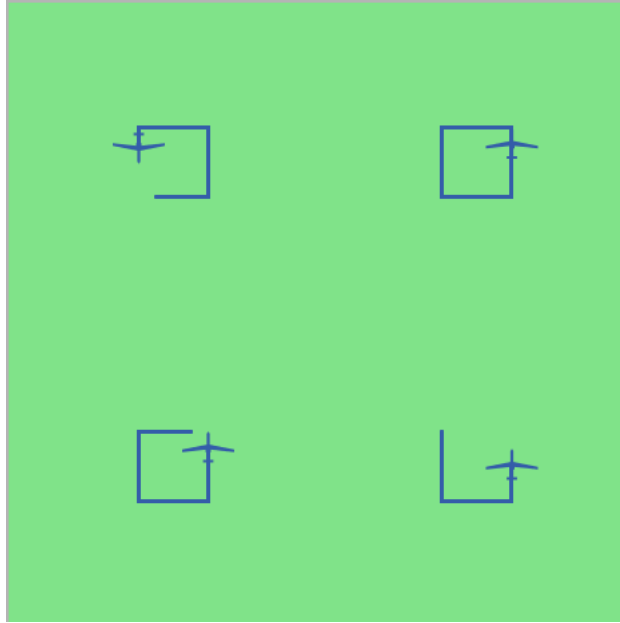


Figure 34: Orbit Patrol Pattern

Finally, the last patrol pattern is called border patrol. The basic premise of this pattern is that each aircraft will patrol the edge of its area. For example, Figure 35 shows 5 aircraft with a box division using the border patrol pattern.

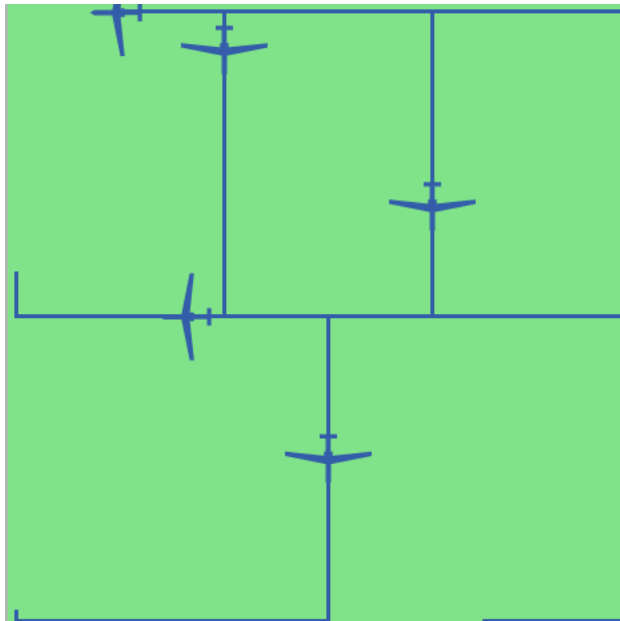


Figure 35: Border Patrol Pattern

5.5.2 More on Doctrine as Variables

Since this method is designed to aid in the quantification of doctrine a most important point of emphasis is how one can come up with doctrine variables for any particular problem. The first step in understanding how to identify a potential doctrinal variable is to reexamine the definition of doctrine. Doctrine was defined as the guidance for the employment of assets in military operations. Also recall that doctrine is a broad term that could potentially encompass national strategy, operations in a theater, or asset level tactics. So in rather informal terms, if somebody is doing something, the way they are doing it can be called doctrine.

Sometimes, the identity of a parameter as doctrine can be unclear. For example, Shaw stated that one example of an air combat tactic is speed. However, the case can also be made that speed is a design variable for a new system. If a current aircraft is capable of flying at 300 miles per hour and a next generation asset is planned to fly at 500 miles per hour, than the new aircraft is capable of a different set of tactics than the old asset. So how is this overlap between doctrine and materiel variables resolved? There is a delicate distinction here that must be highlighted. Materiel variables characterize the physical properties of an asset while doctrinal variables describe the operations, actions, decisions, and employment of an asset. So the fact that aircraft A is capable of a maximum speed of 300 miles per hour while aircraft B is capable of a maximum speed of 500 miles per hour means that the physical properties of the two aircraft are different and thus the maximum speed is a materiel variable. However, the particular speed that is prescribed for the asset during its operations is a doctrinal variable, specifically a tactic. The tactic could be to fly

at “maximum possible speed” and while the instantiation of that tactic may differ from asset to asset, the doctrine can be fulfilled in all cases. There has always been an extremely tight coupling between doctrine and technology, and yet through careful articulation of the context it is possible to distinguish between the two.

Another point to consider is the generation of ideas for doctrinal variables. Often times, evolutions in doctrine occur based on hard experience such as different actions in combat. Additionally, new doctrinal concepts can be proposed and tested in training or wargames. There are extensive publications about current doctrine, tactics, techniques, and procedures. All of these sources can also feed the quantitative methodology outlined in this research. In fact, quite possibly the best way to harness the power of this method is to employ it in conjunction with the current state-of-the art in doctrine. This technique of surveying doctrine and identifying promising variables to adjust in the agent-based simulation was one of the techniques employed in this study.

Another technique which was demonstrated earlier is functional decomposition. Knowing that doctrine is the actions or functions performed during a mission, it is reasonable to apply functional decomposition as a means to reduce doctrine down to basic elements which can then be modeled. Functional decomposition also helps account for the hierarchical nature of doctrine. For example, at the strategic level, a doctrinal statement could be ‘prosecute time-critical-targets in defended airspace’. To decompose this statement determine what functions need to be performed in order to achieve the mission. Those functions could be find, fix, track, target, engage, and assess. Now, further decompose each of those functions into finer detail. To find, it is necessary to select assets, position relative to area of operations, provide tasking information, take-off,

ingress to area, execute search pattern, and employ sensor. The final level of detail comes from taking those operational functions down to specific tactics that can be employed at each stage. For example, a search pattern could be a parallel search or a border patrol search. The ingress to area of operations could be done at maximum altitude at high speed or nap-of-the-earth to shield against radar. Some of this functional decomposition can be found in current doctrinal literature but it also is a tool which can be extremely useful to an analyst to take a fresh look at a problem.

Specific to this research, a combination of legacy doctrine survey/adaptation and functional decomposition was employed. Recall the set of doctrine variables being examined are: number of aircraft, system architecture (a single hunter-killer type, or a hunter aircraft and a killer aircraft), the type of asset used (MQ-1, MQ-9, RQ-4, F-16, or F-35 type aircraft), division of area of operations, and patrol pattern. For this problem, decomposing the mission led to questions about what number of aircraft should be employed as well as how those assets would be allocated within the area of operations. The survey of doctrinal publications led to the inclusion of aircraft type, system architecture, and patrol pattern.

5.6 Step VI: Analysis of Materiel/Non-Materiel Approaches

5.6.1 Section Introduction

The analysis of materiel/non-materiel approaches is the heart of the Functional Solution Analysis and it can be argued the CBA as a whole. It is in this step that the quantification of the approaches in terms of the scenarios, tasks, and metrics takes place. From the previous step, identification of approaches, the analysis of approaches reads in the morphological matrix which describes the design space. Based on this set of possible solutions, a set of experiments is then constructed and executed. According to JCIDS guidance, the analysis performed herein should ideally be done in the same framework as the baseline analysis executed in Step IV. After carrying out all necessary analyses, the results are analyzed and fed forward one step closer to the end of the CBA process.

This chapter will detail the creation of experiments as well as post-processing of results. Significant time will be spent examining the results as these will form the basis for arguments either supporting or refuting the research hypotheses.

5.6.2 Design of Experiments

The morphological matrix from the previous step has a total of 12 variables, 6 discrete variables and 6 continuous. A quick examination of the matrix might lead to the conclusion that there are 8 possible combinations or systems described by the matrix. However, the 8 vertical columns contain the possible settings for each variable and each variable is independent of the others. Here independent means that selecting the first setting on one variable does not necessitate selecting the first setting on any other variable.

For this particular matrix, the number of levels or settings of a variable is not constant. The number of levels per variable is shown below in Table 9.

Table 9: Number of Levels per Variable

	Variable	Number of Levels
Doctrine	Patrol Pattern (H)	4
	AOO Division (H)	4
	# of Aircraft (H)	8
	Patrol Spacing	4
	Patrol Pattern (K)	4
	AOO Division (K)	4
	# of Aircraft (K)	8
Configuration	Hunter Type	6
	Killer Type	6
Materiel	Sensor Range	continuous
	Sensor FoV	continuous
	Weapon Speed	continuous
	Cruise Speed	continuous
	Fuel Burn	continuous
	Fuel Capacity	continuous

In this table, the term continuous refers to the fact that there are not a preset number of levels for that particular variable. Instead, those variables are defined over a range of values. Assuming for a minute that the continuous variables are only evaluated at their minimum and maximum values, then each continuous variable would have two levels. Then, in order to determine the number of possible combinations simply compute the product of the number of levels for all variables. For this example, that would be $4 \times 4 \times 8 \times 4 \times 4 \times 4 \times 8 \times 6 \times 6 \times 2 \times 2 \times 2 \times 2 \times 2 = 150,994,944$ possible combinations! Now assume that all of those cases needed to be run and that the execution time was a sporty 1 case per second. In that case, running all those combinations would take almost 42,000 hours

or just under 5 years. If a higher resolution is desired for the continuous variables then the computations expense increases dramatically. For approximate figures on the length of time it would take to run all cases as a function of continuous variable settings see Table 10 below.

Table 10: Execution Times for Full Factorial Experiment

Levels for Continuous Variables	# of Cases	Execution Time @ 1 case / sec (hours)	Execution Time @ 1 case / sec (years)
2	150,994,944	41,943	5
3	1,719,926,784	477,757	55
4	9,663,676,416	2,684,355	306
5	36,864,000,000	10,240,000	1,169

With such extensive computational demands, clearly a true full factorial experiment is impossible. Therefore, as postulated in Chapter III, it shall be necessary to construct a statistical design of experiments to reduce computation demands while still isolating the impacts of individual variables.

The list of variables being investigated appears in Table 11. The top half of the table contains all of the discrete variables and all of their settings while the bottom half contains the continuous variables and the ranges over which they shall vary. In order to create the design of experiments the JMP software developed by the SAS institute was used. JMP is a statistical package for analysis and visualization of data that includes many powerful tools including a design of experiments generator.

In this particular case, Table 11 was entered into JMP's custom DoE option. Since the experiments need to determine not only the impact of the variables themselves but also interactions, the DoE was constructed to account for main effects as well as 2nd order interactions. That is to say, this design of experiments isolates the impact of hunter speed,

for example, as well as the interaction between hunter speed and patrol pattern. The second order interactions include squared terms for the continuous variables but not so for the discrete variables. There is no physical meaning for patrol pattern squared or other similar variables.

Table 11: Input Settings for Design of Experiments

		Variable	Settings and Ranges						
Discrete	Hunter Division	1	2	3	4				
	Killer Division	1	2	3	4				
	Hunter Pattern	1	2	3	4				
	Killer Pattern	1	2	3	4				
	Number of Hunters	1	2	3	4	5	6	7	8
	Number of Killers	1	2	3	4	5	6	7	8
	Spacing	1	3	5	7				
	Max Weapons	2	4	6	8				
Continuous	Hunter Speed	0.2	1.6						
	Killer Speed	0.2	1.6						
	Hunter fuel burn	0.05	0.15						
	Killer fuel burn	0.05	0.15						
	Weapon Speed	1	3						
	Weapon Range	3	15						
	Hunter Max Fuel	100	400						
	Killer Max Fuel	100	400						
	Sensor Range	3	15						
	Sensor Angle	60	180						

There were two DoE's created from the above table. One was for the combinations of doctrine and technologies for an architecture consisting only of a single aircraft type. This DoE consisted of 1024 cases. A second DoE was created to perform experiments on an architecture consisting of a hunter aircraft and a killer aircraft. Since there were more parameters in this situation the DoE totaled 2048 cases.

Each design of experiments was input into the ModelCenter wrapper and run through the NetLogo environment. The results were compiled back into JMP alongside the lists of inputs. With the data in hand, the next step is to analyze the information.

5.6.3 Analyzing Cases

Since the design space is so large and it was not possible to run all combinations, the strategy is to create surrogate models to accelerate the analysis. The design of experiments is meant to enable the creation of surrogate models capable of representing the agent-based simulation. The surrogate models were created in JMP using the Neural Network module.

5.6.4 Simple Tradeoff Environment

To begin analyzing the results, first examine the surrogate for percent targets killed for a single aircraft type. A section of the prediction profiler is shown below in Figure 36. The value of the function appears on the far left of the figure. There are six subplots each representing the partial derivative of the surrogate function with respect to a particular variable along the horizontal axis. The first three subplots are doctrinal variables; area division, hunter pattern, and number of hunter aircraft respectively. The other three subplots are materiel variables of aircraft speed, sensor range, and sensor angle. Each variable's current value appears in red at the bottom of the plot.

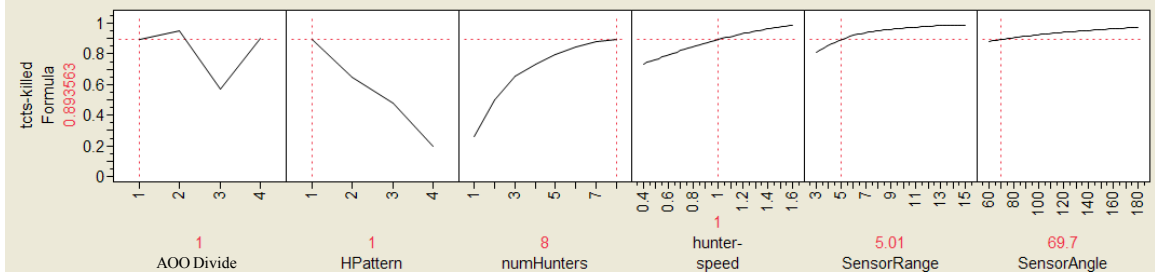


Figure 36: Dynamic Sensitivity Derivatives for Single Aircraft Type

For this plot, the division value of 1 represents no division, value of 2 is basic division, 3 is strip division, and 4 is box division. For the pattern, 1 equates to random-walk, 2 is parallel search, 3 is orbit pattern, and 4 is border pattern. Right now, Figure 36 shows a fleet of 8 aircraft in a random-walk pattern with no area of operation division. For this particular area division with this number of aircraft and the corresponding aircraft properties, if the aircraft were to use a different patrol pattern, the performance would be degraded. However, if instead of no subdivision of the AOO, there was a strip division, then the slope of the pattern subplot changes dramatically. This can be seen in Figure 37 below. The pattern subplot (second from the left) is now jagged and a pattern of random-walk is no longer the best option, rather a parallel search or border patrol option are superior.

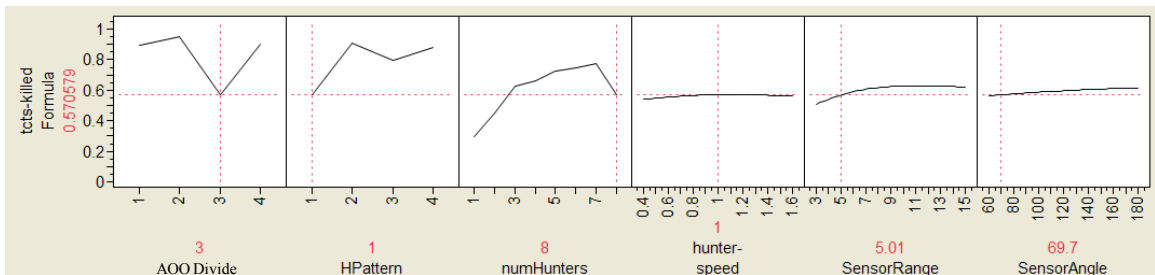


Figure 37: New Area Division Yields New "Best" Search Pattern

For another example of how doctrinal variables interact examine Figure 38. In this figure there are three dynamic sensitivity derivatives arranged from top to bottom.

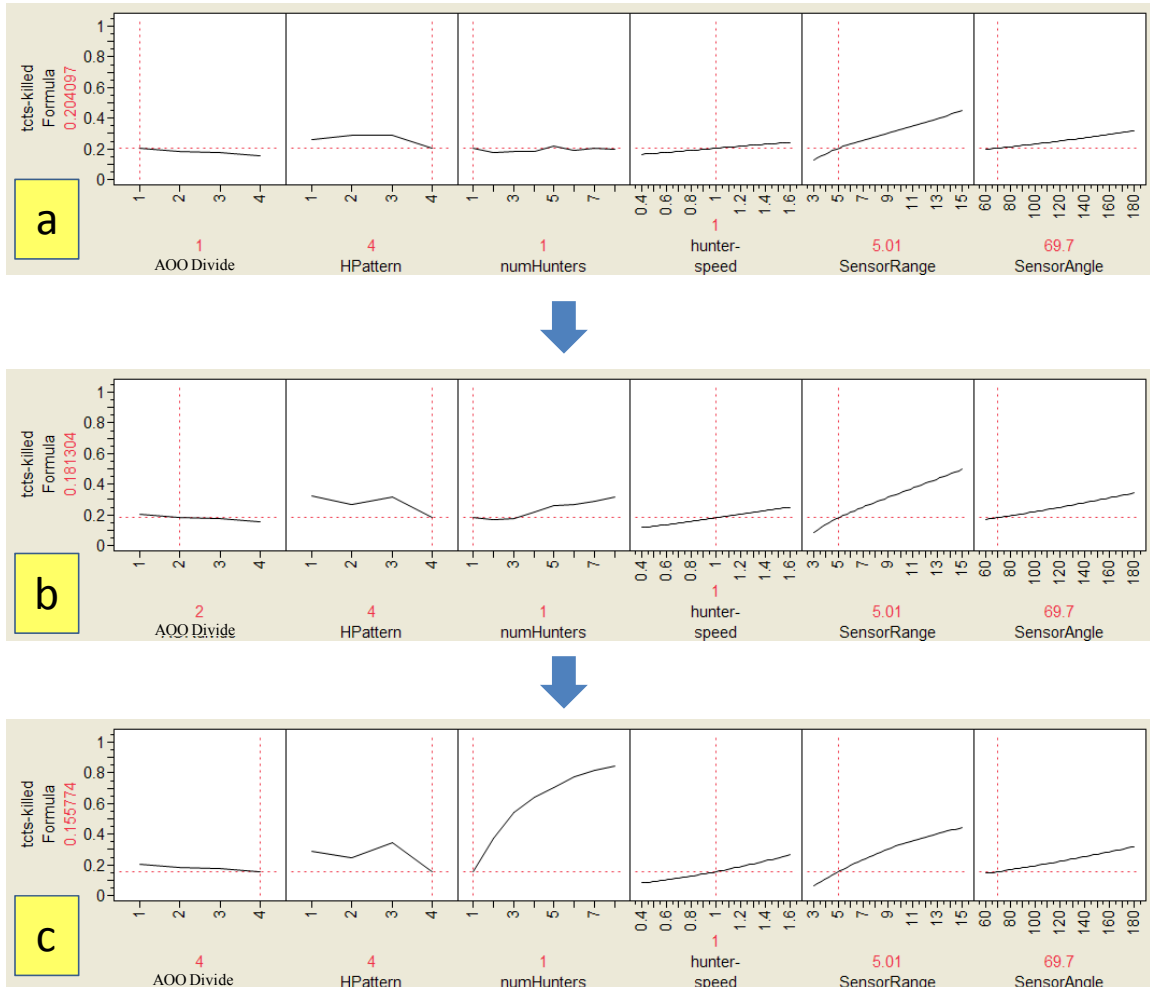


Figure 38: Using Dynamic Sensitivity Derivatives to Illustrate Doctrine Interactions

From profiler ‘a’ to profiler ‘b’ the division doctrine changes from “None” to “Basic”. There are minimal changes to the rest of the subplots between profiler ‘a’ and ‘b’. However, between profiler ‘b’ and profiler ‘c’ as the division changes from “Basic” to “Boxes” the shape of the curve representing the number of hunter aircraft changes dramatically. In the first two profilers ‘a’ and ‘b’, the overall performance was relatively

independent of the number of aircraft in the fleet. However, for this new set of doctrine, the number of aircraft is an important driver. So depending on the settings, doctrinal variables can have very little or very strong interactions with each other. These results are for only a small subset of the total model but begin to show the complex role that doctrine plays in military missions.

Now that interactions between doctrinal variables have been demonstrated, it is time to consider if and how doctrine and materiel variables interact with each other. Based on historical evidence, there is an expectation that a strong coupling exists between doctrine and materiel. After all, throughout the history of warfare, new types of weapons have always revolutionized the way war is conducted. However, knowing that there is a historical basis has only limited use for the problem of defense acquisition. What is important is being able to capture the subtleties in the interaction and also quantify the exact impact that doctrine and technology have on each other. It is only through a quantitative analysis that the extent of new tactics, techniques, and technologies can be captured, compared, and acted upon.

Figure 39 below illustrates one example of doctrine-materiel interaction. The top profiler shows a system with no area of operation division, a random-walk pattern and a single aircraft. This system is predicted to successfully prosecute 25% of targets on average. From the top profiler to the bottom profiler, the only change is an increase in the sensor field of view angle, from 70 degrees to 180 degrees. First of all this increases the predicted value of targets killed to around 75% which is a significant increase. However, where the doctrine-materiel interaction comes into play can be seen in the number of hunters subplot. In the top profiler, with each additional aircraft that is added to the fleet,

performance rises substantially. However, when the sensor field of view increases, the plot of targets killed versus number of hunters resembles a plateau. There is diminishing return on adding additional aircraft after the number of hunters reaches two.

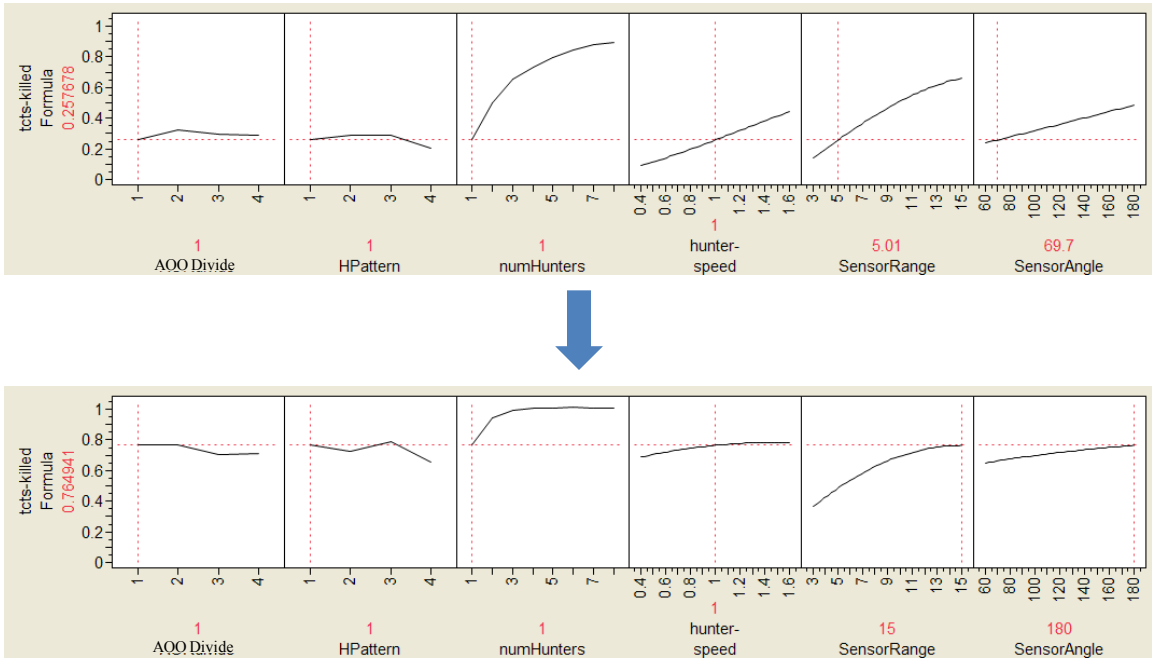


Figure 39: Interaction between Materiel and Doctrine

Qualitatively it makes sense that as each aircraft can see more area per unit time, fewer aircraft are need to cover the same total area. However, with this method, there is now a quantitative answer to when and how that tradeoff occurs. The dramatic change in shape for the curve of targets killed versus number of aircraft is a clear example of how there is a strong interaction between doctrine and materiel. Furthermore, through the agent based model of the Next Generation UAS as well as the prediction profiler it is possible to define that interaction more precisely than ever before.

From the above example it is clear how the prediction profiler can provide insight that was not possible before. However, the power of the surrogate models can extend beyond

this capacity. The next step is to use the surrogate models to execute a design space exploration. This is accomplished with a Monte Carlo simulation. Each input variable to the model is assigned a distribution, in this case a uniform distribution that samples the range of the input variable with equal probability. The simulation was performed in JMP and a total of 10,000 random cases were run. Once the Monte Carlo simulation was completed, the results were compiled and are displayed below in a multivariate plot in Figure 40.

The response variable of percent targets killed is located on the y-axis of the top row in the figure. Each of the six input variables appears on the x-axis of the plot in the same order as they appeared in the above prediction profilers. Furthermore, the rows other than the top row contain plots of the input variables against each other providing an n-dimensional representation of the design space.

Recall that the first three variables are discrete and the last three variables are continuous. In the multivariate plot, the subplots of targets killed against the continuous variables appear as clouds of points. However, for the discrete variables, there are distinct bars in Figure 40 indicating which categorical value of that variable is assigned to a design point.

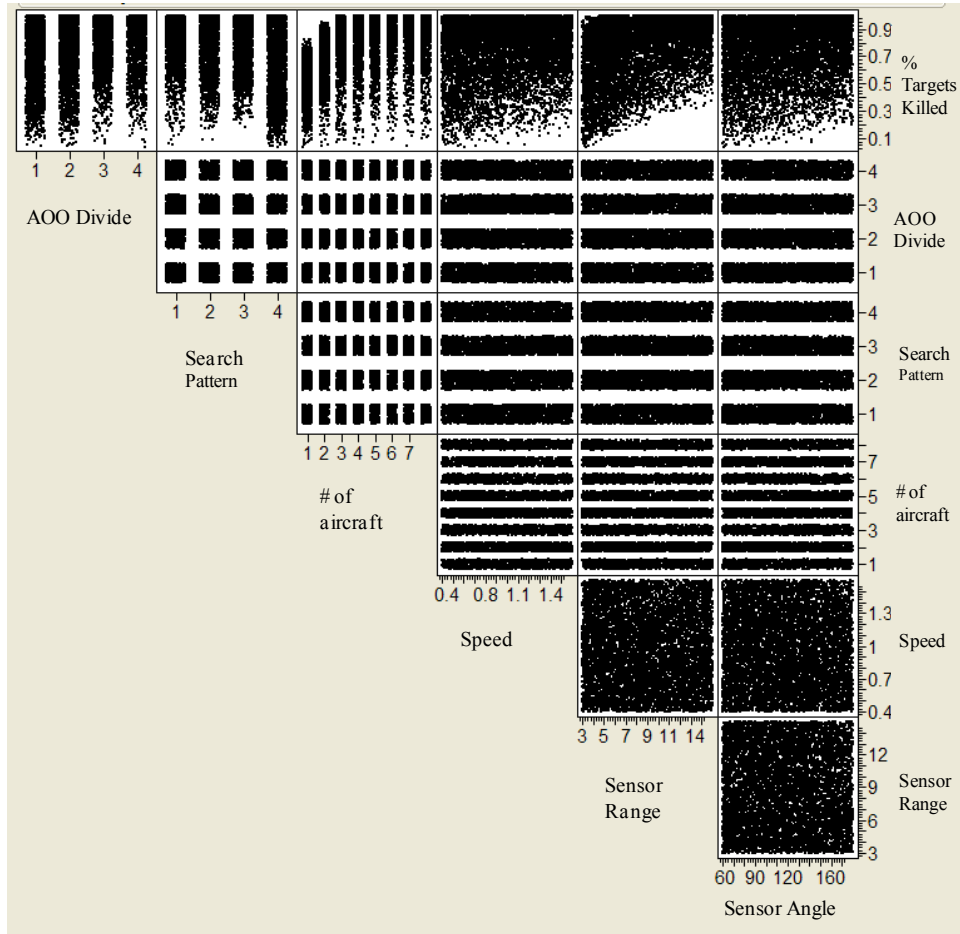


Figure 40: Multivariate Plot for Single Aircraft Type

Figure 40 shows how the design space was uniformly sampled. Additionally, it shows the possible percent targets killed values with respect to all input variables. However, the next step is to parse this enormous data set to ascertain which combinations of doctrine and materiel are most desirable. In order to accomplish this, a constraint was input to the multivariate plot that the only designs being considered are those which exceed the 98% threshold established in the baseline scenario. All other points are then filtered out. The resulting multivariate plot is shown below in Figure 41.

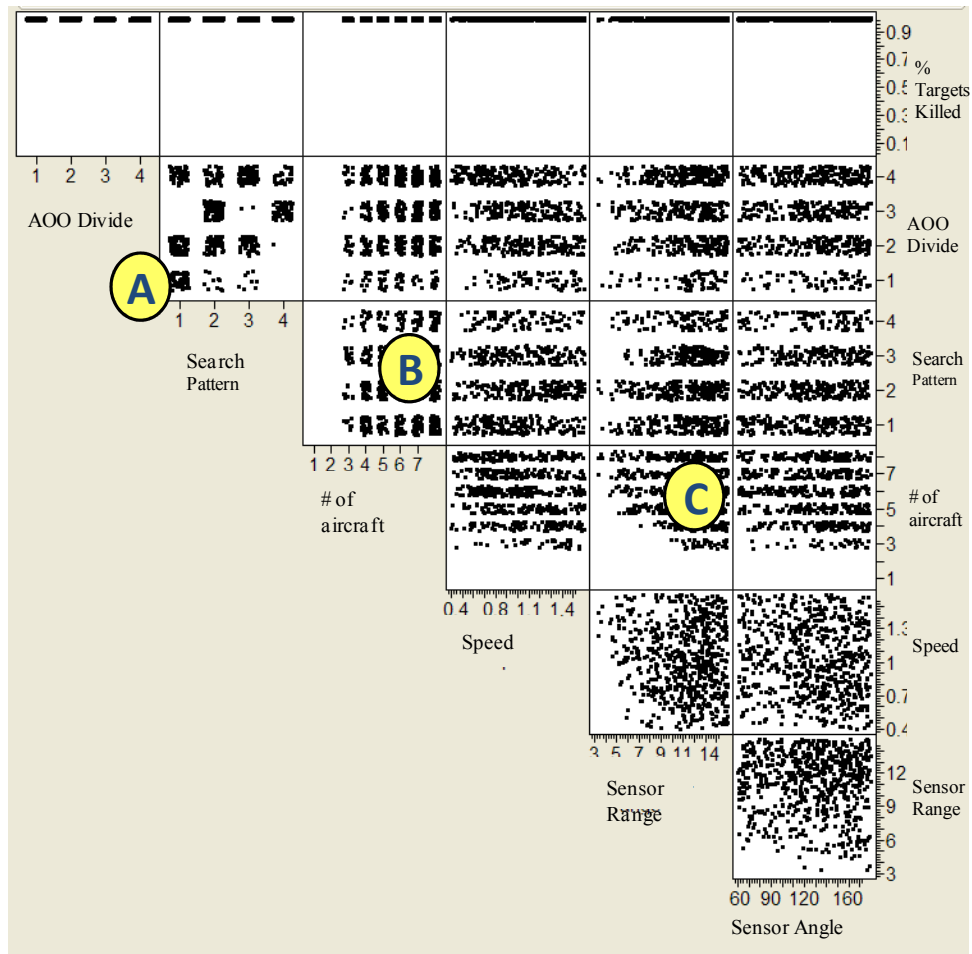


Figure 41: Filtered Multivariate Plot for Single Aircraft Type

The top row of this figure is no longer densely populated with points, rather a thin strip of possible points occupies the very top of that row. At the same time, those points which did not meet the aforementioned constraint have been removed from all the other dimensions of the plot.

The first subplot to examine in Figure 41 is the plot of area division versus pattern, indicated by ‘A’. These are two key doctrinal parameters and determining which settings are most successful is an important goal of this research. In Figure 40 there were 16 distinct blocks of points representing the 16 possible combinations of area division and patrol pattern. However, in Figure 41 two of these blocks contain no points while four

others are almost gone. It appears in this plot that the fourth patrol pattern (border patrol) is a relatively weak option unless the area of operations is divided into strips or to a lesser extent boxes. The first patrol pattern (random-walk) actually appears to be quite effective unless the AOO is split into strips. Also, across all possible AOO divisions, the parallel search (option 2) appears to be the strongest or least influenced by how the area is divided.

Continuing with subplot 'A' but now examining across the rows with respect to the area division options, it appears the strongest option or the one least influenced by patrol pattern type is option 4, the box division. The weakest option is number one, that option that does not divide the AOO at all. It is extremely interesting how there is no single pattern or division option that is universally superior and at the same time, no one setting of either doctrine variable is universally poor. This supports the supposition of strong interaction and feeds the desire for deeper exploration and understanding.

Next, examine the plot of hunter patterns versus number of hunters, indicated in Figure 41 by 'B'. This subplot is very interesting in that it established a hard and fast requirement for meeting the specified capability. Namely, according to this graph there can be no fewer than three aircraft, regardless of other parameters, in order to satisfy the 98% threshold. It is the ability to clearly identify requirements in this fashion that make the top-down design approach supported by the filtered Monte Carlo such a valuable tool.

Finally, the last subplot in Figure 41 to examine is labeled 'C' and is the plot of number of aircraft versus sensor range. This plot is interesting in that it demonstrates a tradeoff that the designer may have to make between two design variables. One corner of

this plot is empty and there is a clearly defined frontier. If fewer aircraft are desired then the maximum range of the sensor must be larger.

The preceding example was a simple case, examining the results for only a single aircraft type against only 6 variables. This example was presented first mainly to manage the level of information and gradually acclimate the reader to the type of trades being performed. Even with a small scale example, there were still some extremely profound results including understanding good and bad combinations of doctrine variables. However, this example leads to many more good and interesting questions such as, what do additional variables contribute to the design space, what role does cost play, and so for. Therefore, armed with an understanding of the analysis method and a basic understanding of elements in the design space, it is now possible to expand the analysis of materiel/non-materiel approaches to its full extent.

5.6.5 Construction of Complete Tradeoff Environment

In order to fully explore all dimensions of this design space, the two sets of surrogate models created for the different system architectures were imbedded in a tradeoff environment in JMP. By using a conditional indicator, the appropriate neural network would be selected based on the input configuration. This environment will take a preset aircraft configuration as an input, match that configuration to the appropriate values for the materiel properties and then use the surrogate relationships to compute the values of targets killed, fuel expended and weapons fired. Additionally, the technology impact matrix was embedded in the environment so that various combinations of technologies could be infused onto a platform thereby changing the values of the materiel properties and thus the values of the responses.

5.6.6 Cost Computation

An important part of the analysis for materiel and doctrine is to establish the cost of a given system. The cost measured is the cost to execute a mission with the complete set of entities, not simply a single aircraft. There are three aspects of the cost computation shown below in Equation 7: the acquisition cost of the aircraft, the technology cost for adding new technologies to the aircraft, and the operational cost of executing the mission.

$$TotalCost = AcquisitionCost + TechnologyCost + OperationalCost \quad (7)$$

Note that in actuality there would be additional costs associated with making these changes. For example, changing doctrine initiates new training which is required to bring the skill level of the aircrews to a point where they can successfully implement the new doctrine. However, these costs are not considered for this sample problem, as addressing the other aspects of DOTMLPF is beyond the scope of this research.

The remainder of this section will detail each of the three cost components. Recall that there are five notional aircraft types loosely based on aircraft available today. The goal of the proof-of-concept implementation is to show how the design methodology can solve a doctrine-materiel capability-based assessment. As such, relative cost values will provide more intuitive comparisons of systems and further de-emphasize the notional aircraft types which were chosen.

5.6.6.1 Acquisition Cost

The first aspect of total system cost to consider is acquisition cost. This cost, as the name implies, simply represents what the cost would be to buy all of the aircraft needed for the mission. The formula for calculating acquisition cost is shown below in Equation 8.

$$AcquisitionCost = n_{hunters} \times unitCost_{hunters} + n_{killers} \times unitCost_{killers} \quad (8)$$

In this equation the number of hunter aircraft and the number of killer aircraft are input variables. The per-unit costs are constants and were estimated from publically available Air Force data. A summary of the acquisition cost data is presented below Table 12.

Table 12: Aircraft Acquisition Costs

Aircraft Type	Unit Flyaway Cost (2007 million dollars)
MQ-1 Predator	\$3.8
MQ-9 Reaper	11.8
RQ-4 Global Hawk	64
F-16 Falcon	18
F-35 JSF	50

All data in Table 12 are shown in terms of millions of dollars and year 2007 dollars. The unit flyaway cost represents what it would cost for the Air Force to purchase a single airframe of each type. However, that acquisition cost is for the entire life of the airframe whereas the goal is to compute the cost to execute a single mission. Therefore it is necessary to divide the acquisition cost by the operational life cycle of the aircraft. For the sake of this study, the life cycle of the aircraft is assumed to be 20 years with an average annual usage of 360 flight hours per year. These assumptions will be uniform across aircraft type.

5.6.6.2 Technology Cost

The second cost component is the cost associated with adding new technologies. This is closely related to acquisition cost in that adding new technologies to a given aircraft increases its cost. Technologies are tied directly to materiel variables through the Technology Impact Matrix shown below in Table 13. On the far left-hand side of this table the seven materiel variables of the proof-of-concept problem are listed. Along the top of the matrix are ten notional technologies which are meant to modify different materiel variables. Within the matrix, when a technology changes a characteristic of an aircraft, the extent of the change is indicated by a percent change. Improvements in the various attributes are shown in green while degradations are shown in red. The improvements are a positive percentage while the degradations are a negative percentage except in the case of ‘fuel burn’ where an improvement actually reduces the amount of fuel burned per unit time.

Table 13: Technology Impact Matrix

		Technologies										Max +	Max -
		T1	T2	T3	T4	T5	T6	T7	T8	T9	T10		
Technology Areas	hunter speed	5%	10%		-5%			-1%				15%	-6%
	fuel burn	(+)5%	(+)1%	-5%	-10%							-15%	6%
	weapon speed					20%		-5%				20%	-5%
	weapon range					-5%	35%	-5%				35%	-10%
	fuel capacity					-5%	-1%	25%				25%	-6%
	sensor range								15%	5%		20%	0%
	sensor angle										10%	10%	0%
	Cost	High	High	Moderate	High	High	Moderate	Low	High	Moderate	Low		
	Time	Poor	Poor	Fair	Poor	Fair	Fair	Good	Poor	Fair	Fair		

Also in Table 13, the bottom two rows labeled “Cost” and “Time” indicate the technology cost incurred and the penalty in realization time respectively. The time factor will be addressed shortly so for the moment, only cost shall be examined. The cost of

each technology was randomly assigned one of three cost levels: low, moderate, and high. Then, each of these levels was assigned a cost value (low = \$100,000, moderate = \$250,000, high = \$500,000) which represents the *per-aircraft cost of adding this technology*. What this means is that if a given hunter/killer system consists of three aircraft, adding one moderate technology and one high technology increases the total system cost by \$15 million. A general formula for computing the cost can be found below.

$$Tech.Cost = n_{aircraft} \times \sum_{i=1}^{10} c_i \cdot T_i = n_{aircraft} \times \sum NewTech \quad (9)$$

where T_i is the i -th technology, c_i is the cost coefficient for the i -th technology, and $n_{aircraft}$ is the total number of aircraft in the system. Note that this includes both hunter aircraft and killer aircraft, if applicable. The technologies are assumed to be applied to all aircraft in the system. Combining that general formula with the information in Table 13 provides a set form for calculating the cost of new technologies.

$$NewTech = 0.5 \cdot T_1 + 0.5 \cdot T_2 + 0.25 \cdot T_3 + 0.5 \cdot T_4 + 0.5 \cdot T_5 + 0.25 \cdot T_6 + 0.1 \cdot T_7 + 0.5 \cdot T_8 + 0.25 \cdot T_9 + 0.1 \cdot T_{10} \quad (10)$$

where the value of each technology is zero if it is not selected, and 1 if it is selected. There are no intermediate values for the technologies.

5.6.6.3 Operational Cost

The final aspect of cost to consider is operational cost. Operational cost is a measure of the resources expended to execute the operation. For this problem, there are two

operational values being tracked: number of weapons fired and quantity of fuel burned. The equation used for calculating operational cost is shown below in Equation 11.

$$OperationalCost = 0.1 \times n_{weapons} + 5 \times 10^{-6} \times fuel \quad (11)$$

where $n_{weapons}$ and fuel are responses. They are computed using surrogate models regressed from the experiments performed in the NetLogo environment. The cost per weapon is assumed to be \$100,000. The cost for fuel is assumed to be \$5 per gallon.

5.6.7 Time

The “realization time” is defined as the time it takes to develop or modify a system such that it can be fully fielded. This time was computed by examining what changes were made to the “starting point” system. First, it is determined if there were doctrinal changes made to the system. The first change examined was architecture type. If there were no killer aircraft (a single aircraft type completes all aspects of the mission), the next parameter to check is number of aircraft. In the case where the number of aircraft in the system is less than the baseline number of 4, no additional production is needed so the number of aircraft does not influence the realization time. If the number of aircraft is greater than 4 but less than 6, this indicates that a slight increase in production is necessary. Since the aircraft types are already in existence, adding a small number of aircraft will take 0.5 years. If the number of aircraft is above 6, then the realization time increases to 1 year. If the patrol pattern or the area division are changed from the starting point values, then a value of 0.5 years is assigned representing the time it takes to train and implement this new doctrine. It is assumed that these changes can be implemented concurrently.

If the configuration architecture calls for both hunter aircraft and killer aircraft, then this increases the time by 0.5 years in order to train the pilots or operators to work together. Otherwise, the realization time penalties are the same as for the single aircraft case.

There is no penalty in either architecture for changing aircraft type since it is assumed that all of these aircraft exist and there are qualified pilots for each type. Refer to Figure 42 below for a diagram of the logic for computing doctrinal realization time penalties.

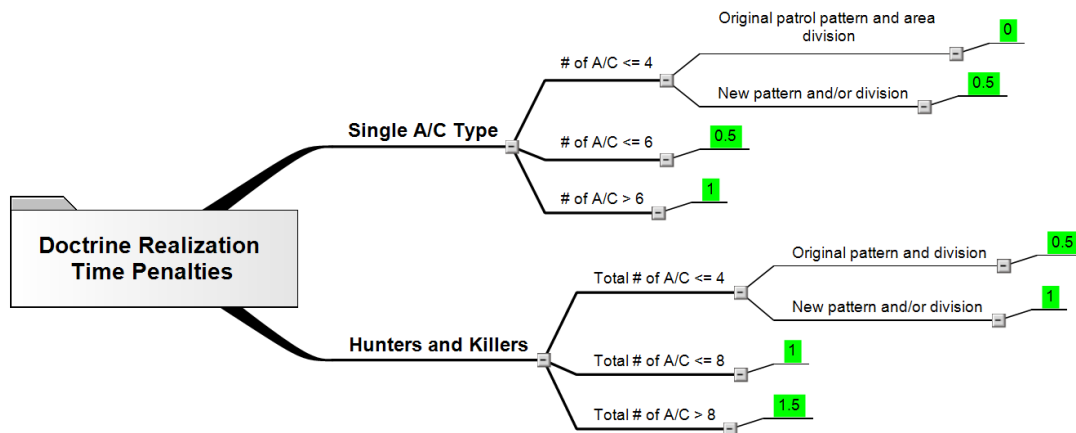


Figure 42: Logic Tree for Computing Realization Time

Once the doctrinal time penalties are accounted for, the technology development times are examined. There are three levels of technology development time, short, medium, and long and they correspond to realization time penalties of 0.5, 2, and 5 years respectively. The total time penalty is computed by determining the maximum time for any one of the technologies being selected. That is to say, if three “short” term technologies are selected, the time penalty is not 1.5 years, but rather 0.5 years. Likewise, if one short, one medium, and one long technology were selected, the time penalty would be 5 years, not 6.5. Refer to Table 13 above for all of the times.

The doctrinal realization time and the technological realization time are added together to give a total realization time. The reason for adding the two is that it is assumed for the purposes of this study that new doctrine cannot be trained and implemented until the technology has been established.

5.6.8 Complete Design Space Exploration

With the analysis environment configured to input doctrine, configurations, and technologies and then compute cost, performance, and time it is now possible to proceed to the exploratory phase of the analysis of materiel/non-materiel approaches. The first question to address is how the possible technologies influence the baseline system. In this case the starting point system is the configuration based on the MQ-1 Predator. It is worth reiterating that *NONE* of the configurations or values in this study are true values. Instead they are approximations and normalizations based on publically available information of known systems.

The following page contains two prediction profilers in Figure depicting the entire design space. In each of the two profilers, there are three responses plotted on the vertical axis: percent targets killed, realization time, and normalized cost. Along the horizontal axis of each profiler are the input variables including default aircraft type, number of aircraft, area division technique, patrol pattern, patrol spacing, architecture type, and all potential technologies. The first profiler depicts the design space at the starting point configuration while the second profiler shows the design space with a two-aircraft type architecture. These graphics are meant to provide an illustration of the design variables and the types of trades that will be performed.

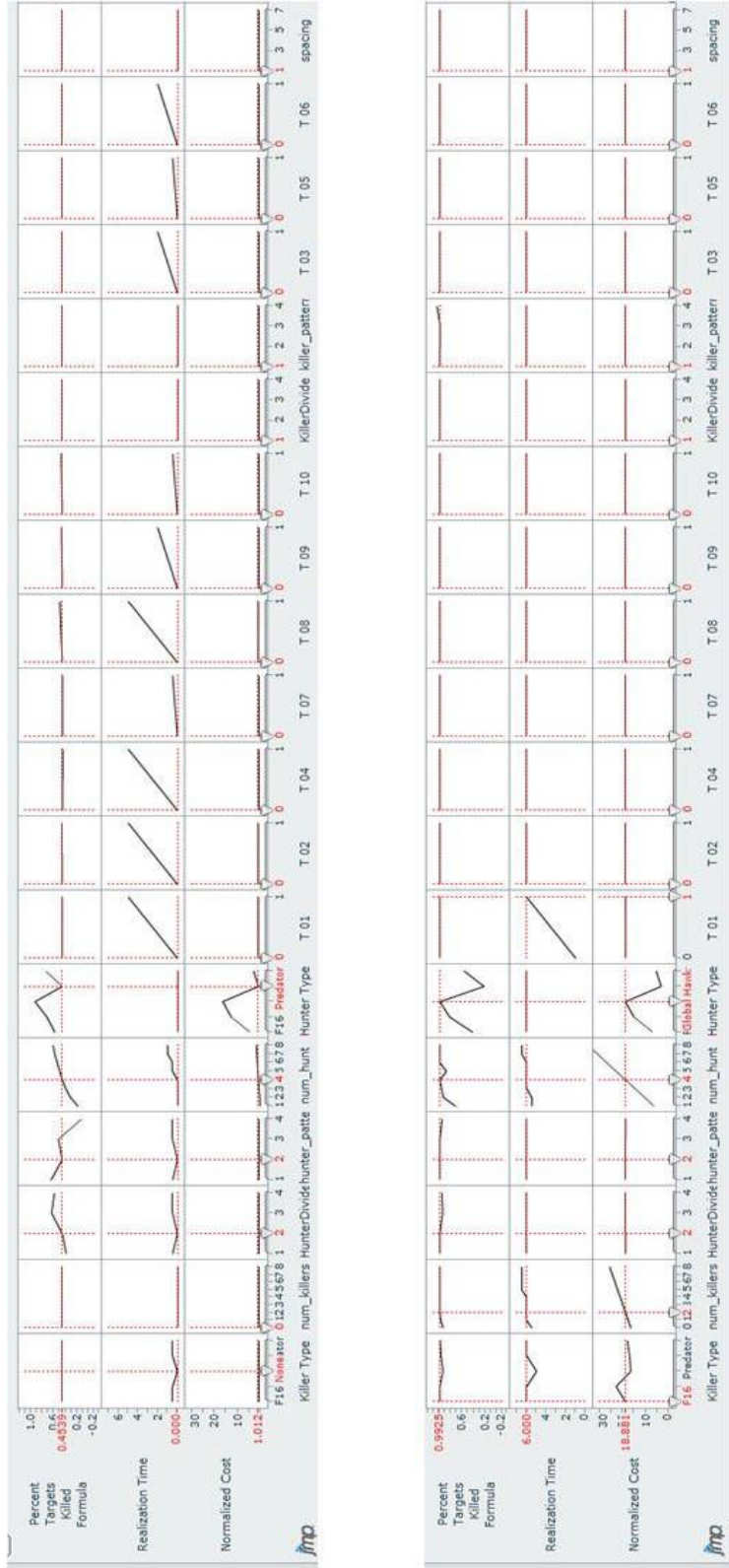


Figure 43: Dynamic Sensitivity Derivatives for Complete Tradeoff Environment

Behind the prediction profilers in Figure are the surrogate models generated from the design of experiments. These equations are parameterized and thus easily queried to analyze various design cases. Through this approach, the analysis time for a single case has been made nearly instantaneous. Therefore, it is possible to statistically sample the design space using Monte Carlo simulation and through an automated execution of the surrogate models, rapidly explore the entirety of the design space. The following plots and results are derived from a 50,000 case Monte Carlo simulation across all of the input variables in the design space.

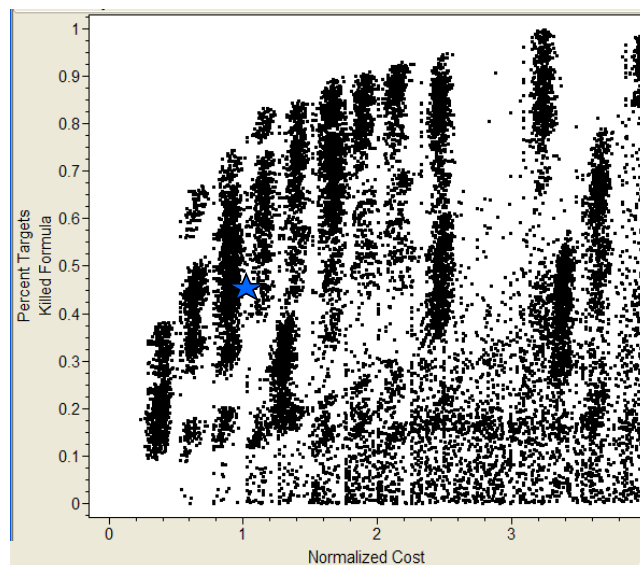


Figure 44: Results of Monte Carlo Simulation for Complete Design Space

Figure 44 shows the results of the 50,000 case Monte Carlo simulation plotted on a graph of percent targets killed versus normalized cost. In this figure, the original starting point assumption is shown by a blue star at a normalized cost value of 1 and an average percent targets killed of 45%. In this form, there is no clear answer as to what the possible approaches are for the hunter-killer problem. What is clear is that there is a

definite boundary of performance, an ultimate Pareto frontier, which runs along the edge of the design space.

In order to glean more information, the first step is to filter the design space. The filter in this case will be realization time. Figure 45 below shows the solution space of Figure 44 filtered to only those solutions available immediately. Recall that solutions available immediately will: require no additional aircraft beyond the assumed starting point value of 4; employ the assumed starting point doctrine of single aircraft type, parallel search pattern, and basic area of operations division; and contain no new technologies. This set of solutions is clearly limited however there is a distinct frontier of performance for the decision maker to choose from.

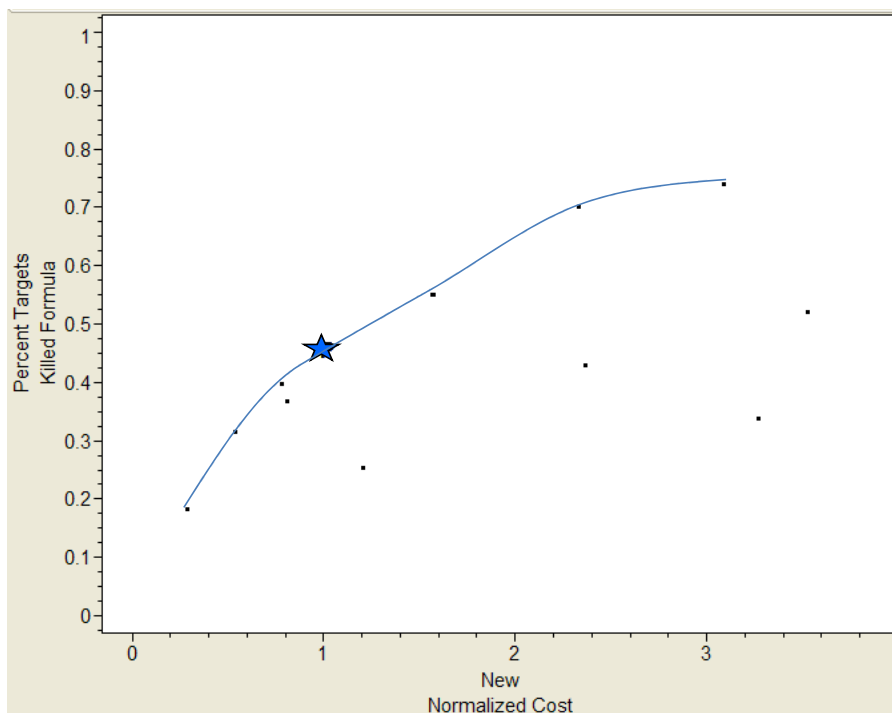


Figure 45: Pareto Optimal Among Immediate Solutions

As before, the blue star indicates the original starting point assumption but this raises the question, what do the other points in Figure 45 represent? In order to show the design space more clearly, the complete Pareto frontier has been decomposed into the frontiers for each aircraft type. This breakdown can be seen in Figure 46. As it stands now, the assumed starting point is one of the best, if not the best, options available. The blue curve depicting the notional Predator-based system is the lowest cost system in terms of executing the mission. No increase in performance is possible while still using the Predator system. An increase in performance can be found by migrating to the Reaper-based system however, by switching the platform, a significant increase in cost will result. The other two notional systems are what is referred to as “dominated solutions” in that they are inferior to the other possibilities in all dimensions of interest.

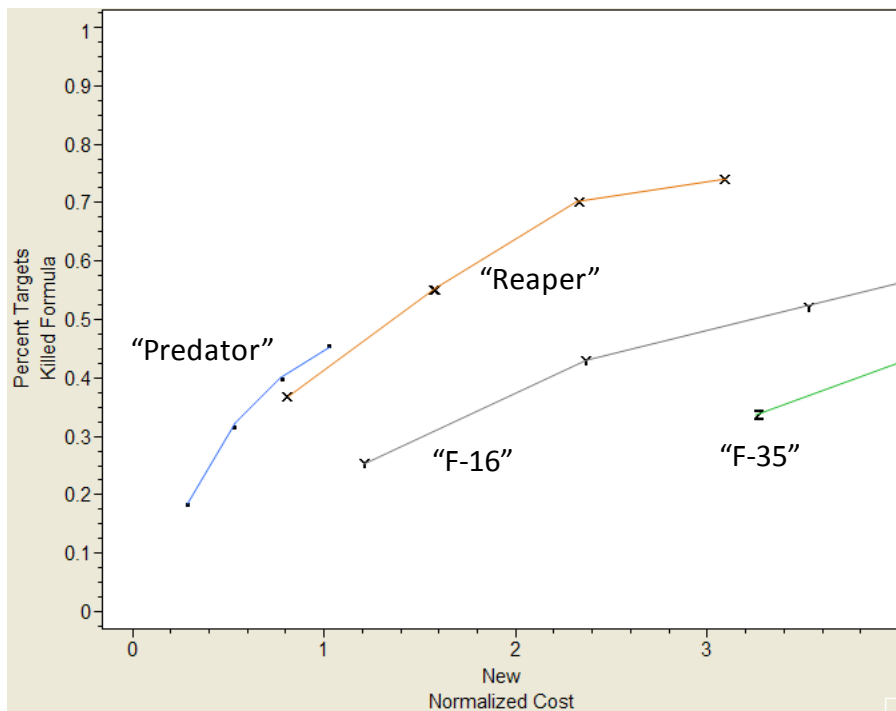


Figure 46: Individual Frontiers for Immediate Solutions

The filtering process continues by expanding out the time horizon to encompass all possible solutions available within 1 year. A visualization of the solution space can be seen below in Figure 47. Recall from the earlier description that within one year, all tactical and operational level doctrine is available provided a single aircraft architecture is used. Additionally, there are some technologies available on these aircraft. Figure 47 contains a new Pareto frontier which is visible as an orange line and can be compared relative to the original starting point denoted by the blue star.

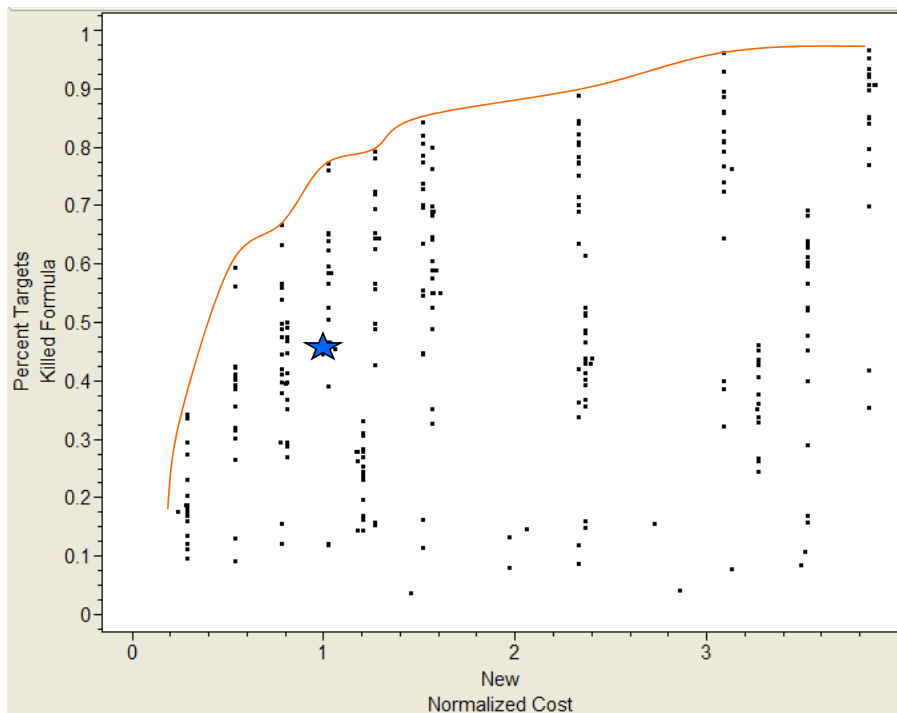


Figure 47: Pareto Optimal for 1-year solutions

Combining the results from the previous figures into a single graphic, it is possible to plot percent targets killed versus cost with the third dimension of time overlaid through a series of Pareto frontiers. This is shown in Figure 48. The long term solutions are represented here by the black line and these solutions can contain any mix of doctrine,

configuration, and technology. As this plot shows, progressively over time, the frontier of solutions shifts upward in effectiveness just as one would expect. However, the intriguing results of Figure 48 is that the curves appear to migrate to the left, meaning over time solutions with equal effectiveness and lower cost are present. This is a very counterintuitive result. Therefore, additional investigations into the content of this design space will try and glean answers to emerging questions.

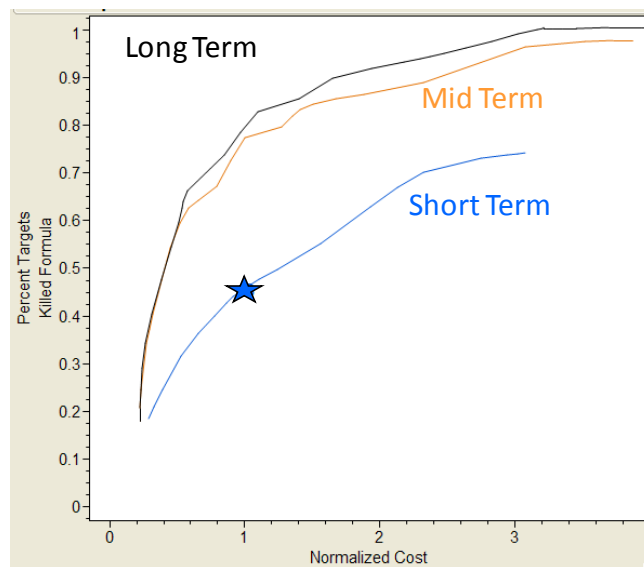


Figure 48: Progression of Pareto Frontiers over Time

Figure 49 below shows two clusters of points. The only located nearest to the blue star represents the notional starting point aircraft with various technology infusions while the other cluster of points represents the aircraft based on the notional Reaper. Recall that these were the two non-dominated aircraft types. This graphic depicts what is possible if none of the doctrine, other than aircraft selection, were changed; a constant number of aircraft, constant area assignment, and constant patrol pattern. This graph shows two things. First, the technological changes, even grouped together, are not equal to transforming one aircraft into another, based on the assumptions developed herein.

Secondly, while there is substantial improvement to be gained through changing the aircraft type, significant costs are incurred.

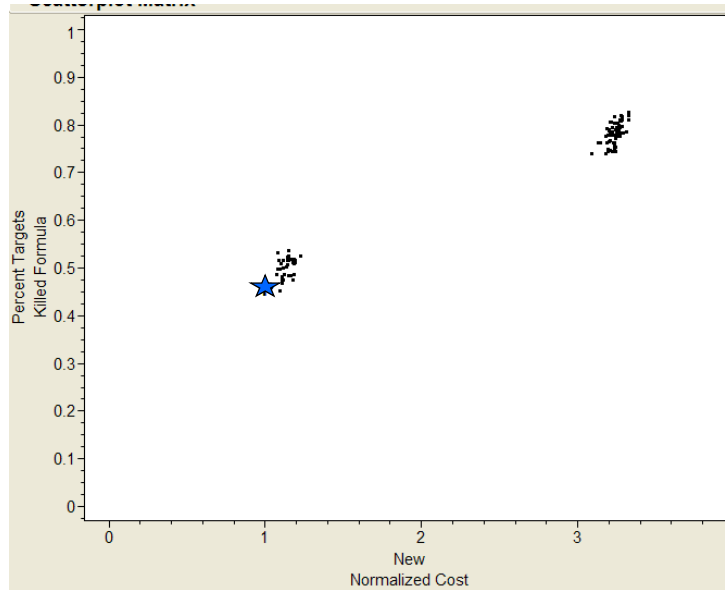


Figure 49: Fixed Doctrine, Technology Infusion for Two Types Starting Points

Next, Figure 50 shows the same clusters of points, only now the area division and patrol pattern have been allowed to vary. In this particular image, the clusters contain points with a “box” division of the area of operations and an orbit patrol pattern. It was found that this combination of doctrine provided the best results in terms of effectiveness. The main point to take away from this graph is that doctrinal changes offer improvement across different types of systems but it did not close the gap in terms of cost.

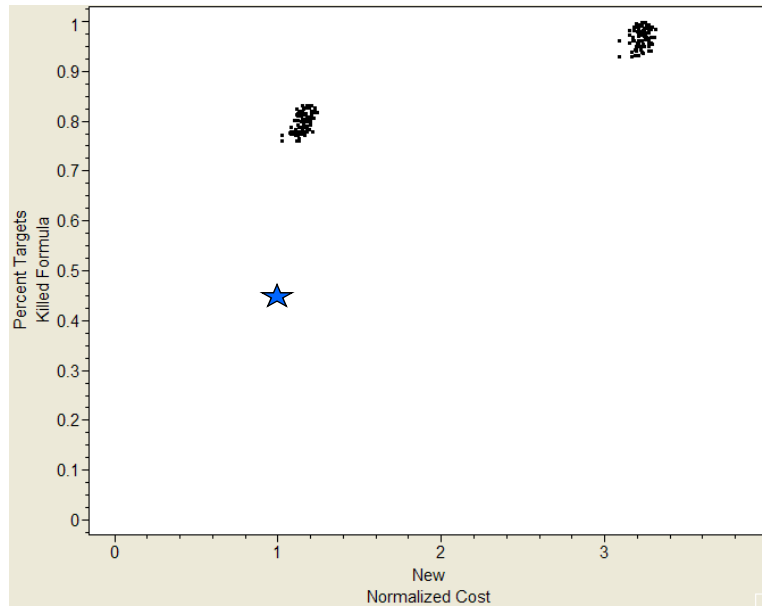


Figure 50: Improved Doctrine, Technology Infusion

Now the truly interesting discovery that can be made by examining the previous two figures pertains to the relative change in effectiveness between doctrine and materiel variables. To fully appreciate this comparison Figure 51 shows a cluster of points in blue which are technological combinations on the starting point doctrine while the orange cluster of points are technological combinations with the improved doctrine of box division with orbit patrol. What this graph shows is that the magnitude of change in effectiveness due to doctrine is approximately three times greater than that of the materiel changes. This is a fundamental finding for this problem. While it was hypothesized that doctrine would provide greater cost effectiveness, it was not anticipated that doctrinal changes could provide overall greater effectiveness.

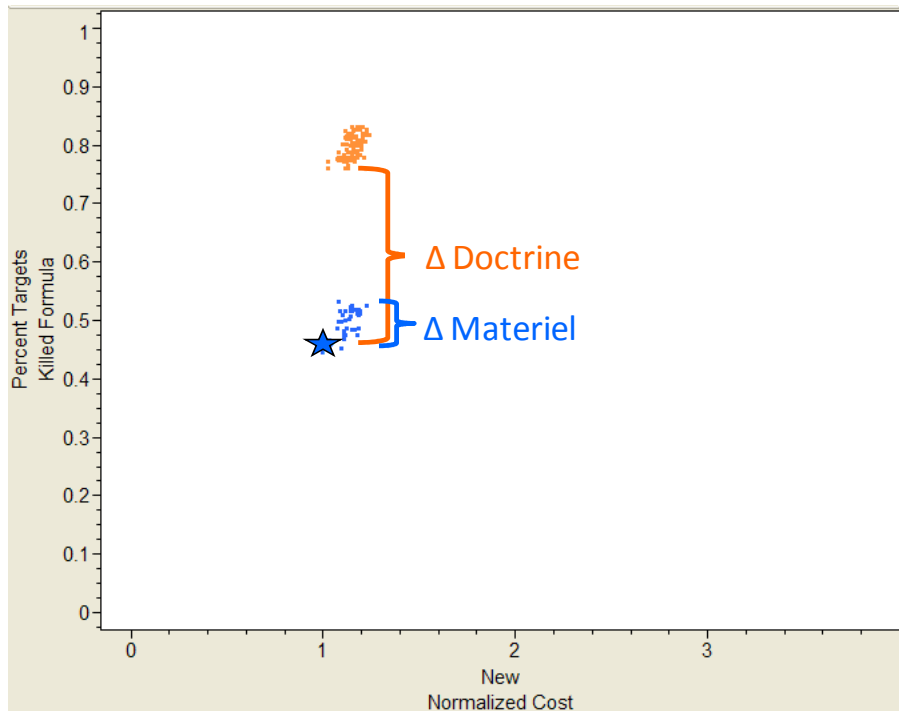


Figure 51: Change in Effectiveness due to Doctrine vs. Materiel

Having seen how important doctrine is, the next step to understand how effectiveness can be achieved through reduced cost is to vary the doctrinal parameter of number of aircraft. Figure 52 shows a plot of targets killed versus cost with two types of aircraft plotted; the notional Predator with the closed circles and the notional Reaper with the open diamonds. Additionally, each aircraft type is plotted in eight clusters representing a fleet size from 1 aircraft to 8 aircraft. Also, each approach used the box division and orbit patrol pattern and on top of that, combinations of technologies make each cluster.

As is seen in this graph, all clusters of points, with the exception of the lone Predator type aircraft, give superior performance to the initial starting point, denoted by the blue star. Also, there are many approaches, including the single Reaper type asset which have a lower cost than the starting point system. The main driver for reducing cost is reducing the number of assets. However, improved effectiveness can be gained through combinations of doctrine and materiel which make a smaller fleet even more effective.

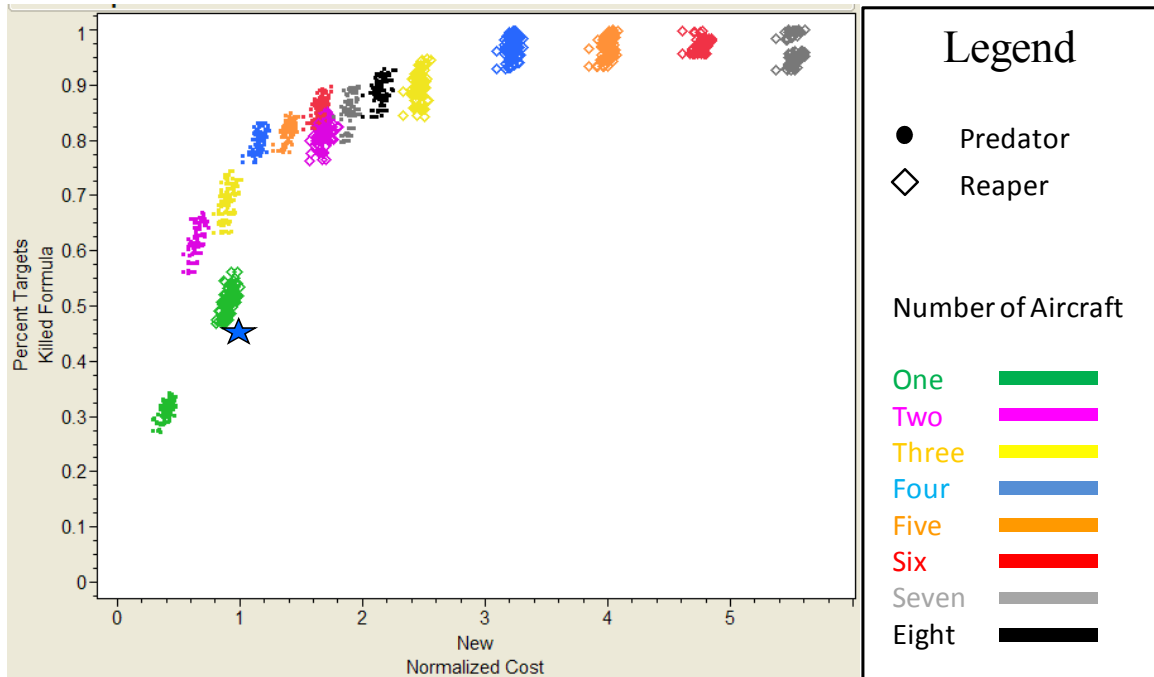


Figure 52: Varying Number of Aircraft

Now that the design space has been more thoroughly explored, the next step is to downselect to possible approaches. Figure 53 below shows the design space after it has been filtered to eliminate all approaches that do not meet the 98% threshold. Plotted in this multivariate plot are percent targets killed, normalized cost, hunter type, number of hunter aircraft, area division, and search pattern.

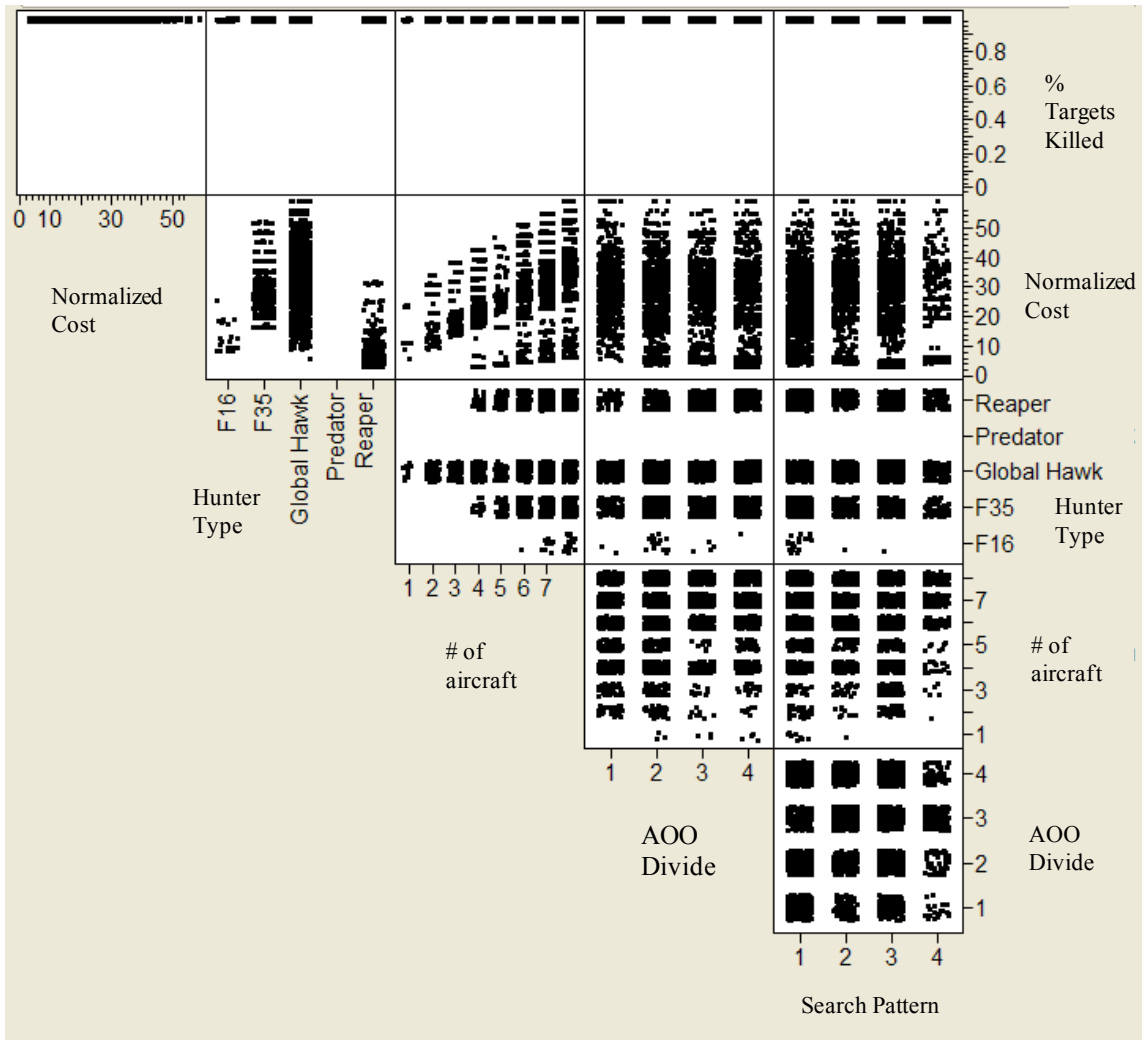


Figure 53: Design Space Filtered to 98% Requirement

The set of solutions in Figure 53 represents all solutions available at the far time horizon. Next, the design space will be filtered to solutions available within 2 years, as seen in Figure 54. Many of the potential approaches are now eliminated. What this means is that those approaches would have required very significant modifications to the doctrine and more so to the technologies in order to meet the performance requirement.

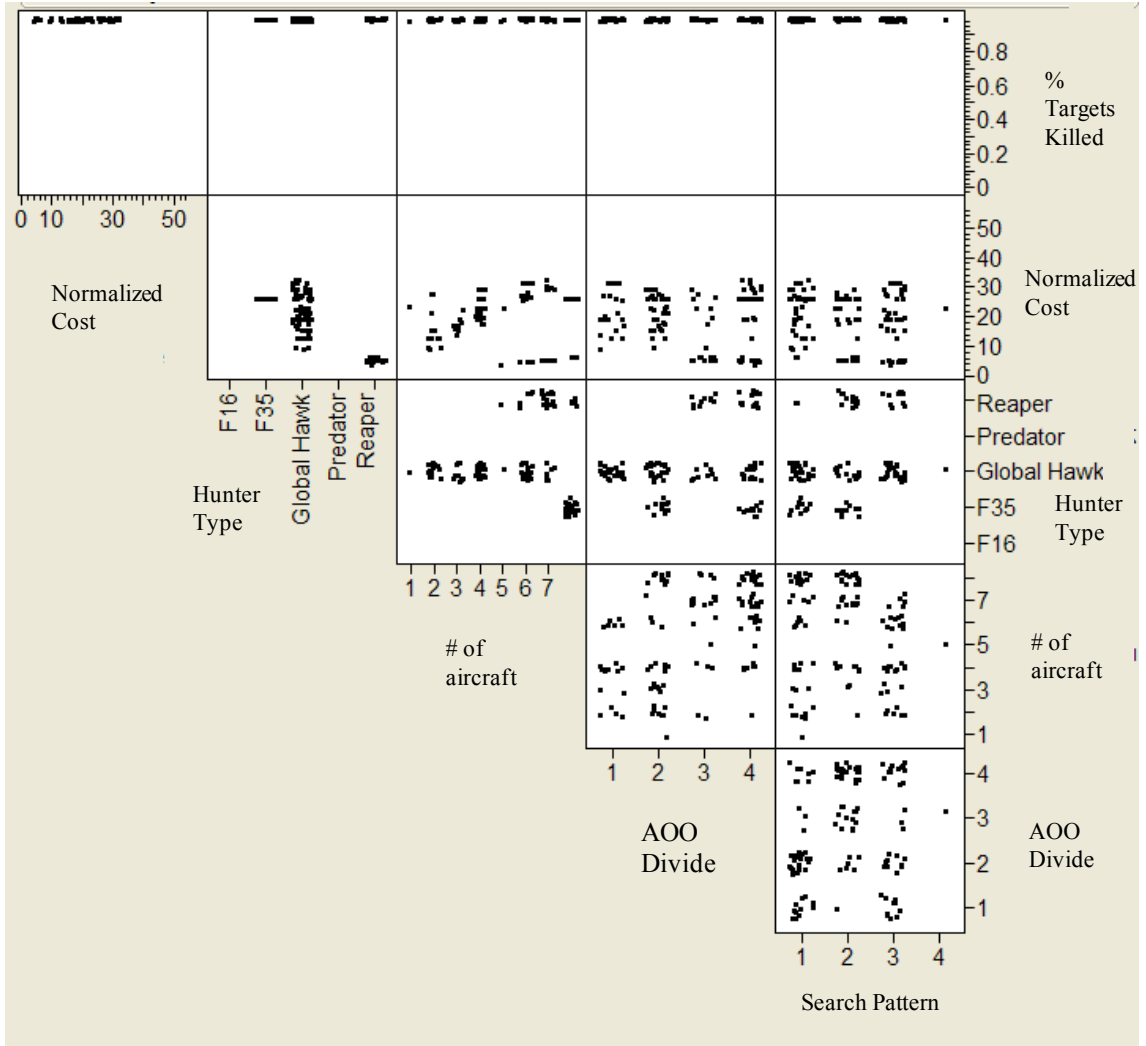


Figure 54: Solutions Available in 2 Years

Filtering one last time to the set of solutions available within 1 year leaves an even sparser design space shown below in Figure 55. There are several important conclusions to take away from this graph. First, whereas in Figure 54 there were several different architectures available including using a Reaper type hunter/killer, an F-35 type hunter/killer, and a Global Hawk type hunter with a different killer type aircraft, now the only solution that remains is the Reaper type. Furthermore, all of these solutions entail a fleet of six aircraft, a box division of the area of operations, and an orbit patrol pattern.

There are a few different options for technology combinations but regardless, those other doctrinal criteria must be met.

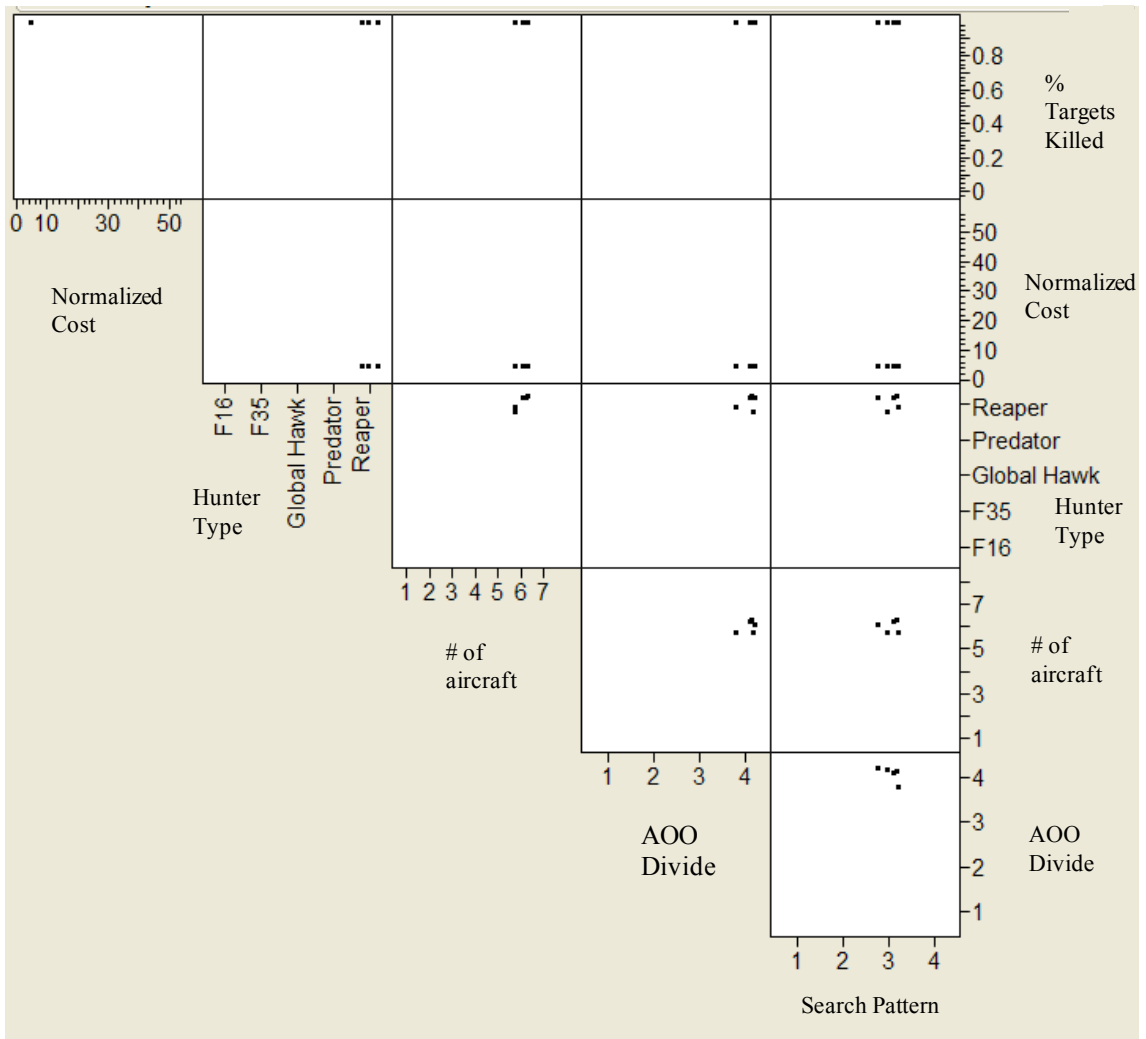


Figure 55: Solutions Available in 1 Year

To this point, the solution with the quickest realization time for meeting the criteria has been found, but it is not the cheapest approach possible. In fact, there exists an approach featuring 4 Reaper type aircraft with box division, orbit patrol, Technology 7 for fuel capacity, and Technologies 9 and 10 for sensors which will be available in two years that turns out to be the least expensive approach capable of meeting the goals regardless of any other future combination. The question now becomes, what are the

options for achieving that approach? What is the roadmap for how capabilities should be realized?

The first option for achieving that approach is to change the patrol pattern, area division and fleet sizing in accordance with the results of Figure 55. Then, wait for the technologies to mature and reduce the fleet size from 6 aircraft to 4. However, the problem here is that the aircraft cost is the largest driver and thus using two extra aircraft could be cost prohibitive. Therefore, additional paths to the solution were sought. Figure 56 below shows two such paths which will be explained next.

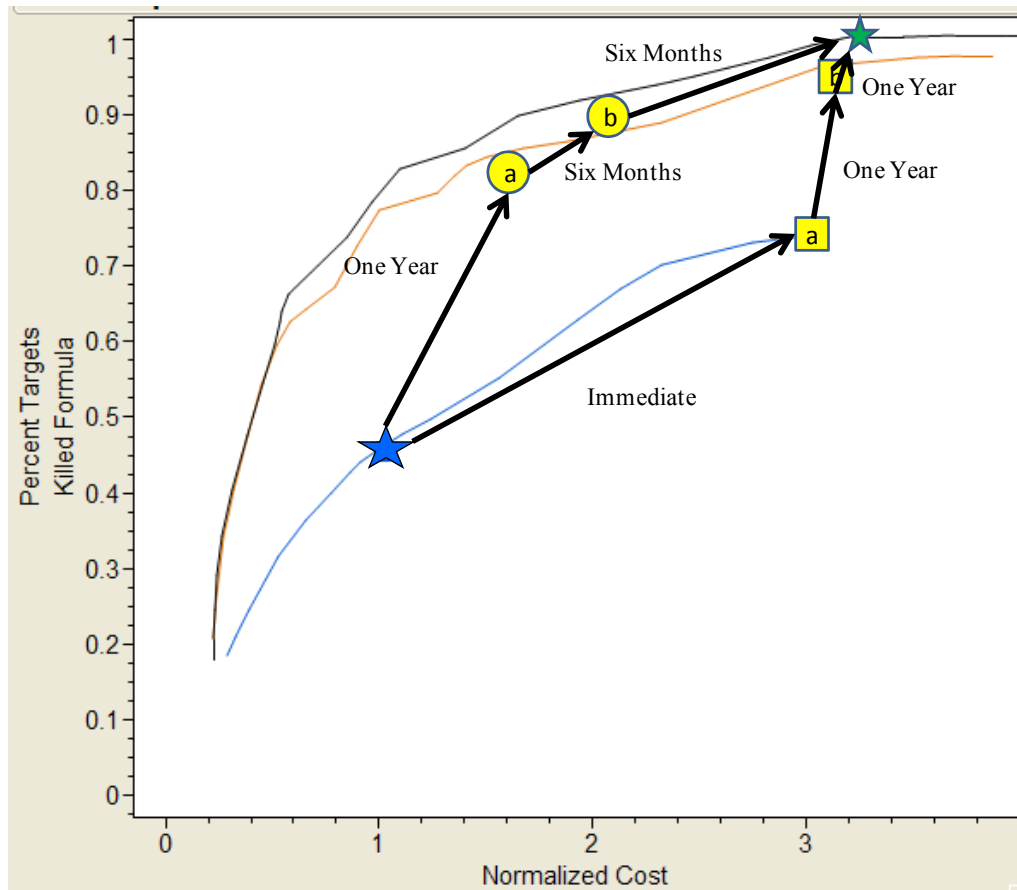


Figure 56: Potential Paths to the Solution

In Figure 56 there are the three original Pareto frontiers in blue, orange, and black along with the blue star denoting the assumed starting point. However, there is now a

great deal more information in this graph. First, the green star at the upper right corner of the graph shows the eventual solution; the approach available in two years that will meet the performance requirements at the lowest cost. There are also two sets of arrows, one connecting a series of circles and another connecting a series of squares. These shapes represent the intermediate points which progressively build capability in order to reach the final approach.

First, examining the path through the squares, it is possible to immediately move along the Pareto frontier by switching from Reaper type aircraft to Predator type aircraft. This point, box 'a' incurs much more cost but improves performance by over 25%. The next move from here requires developing new doctrine for area division and patrol pattern as well as a new technology for fuel capacity. This takes one year and brings the system to box 'b'. From here, the system is almost at the final state. To get the extra performance, an additional year is needed to fully mature two additional sensor technologies.

The alternative route is to make no immediate changes to the system while developing new doctrine and technologies. Circle 'a' involves an increase in the number of Predator type aircraft used from 4 to 8. Concurrently, the technology for fuel capacity can be developed and deployed on the new systems. From there it will take six months to develop the new doctrine for improved search pattern and area division. The final six months involve the maturation of technologies for sensors as well as a switch from the Predator type to the Reaper type.

Both of the above paths provide a route to the solution. However, they are different in terms of when the investments are made. This sort of mapping of solutions over time is

incredibly powerful and is a unique insight made possible by the incorporation of doctrine into the capability-based assessment.

5.7 Step VII: Reassess Gap

Now that all of the analysis cases have been run and the data reduced, the next step is to reexamine the capability gap that led to the FSA in the first place. Recall from Steps III and IV that the original system against the baseline mission was capable of 98% effectiveness but when trying to execute the advanced mission, the capability dropped to around 40% of targets killed. The goal was to see if there existed solutions which could return to the 98% threshold.

From the analysis of materiel/non-materiel approaches in Step VI it was shown that there were many solutions which met the 98% targets killed threshold yet none were available immediately or in the short term, regardless of cost. However, there were several strategies for meeting the capability threshold over time. Additionally, in the near term, there existed a variety of options for improving capability and/or reducing cost.

Since there are options with respect to the capability threshold, it is possible at this stage to say that the capability gap has been theoretically closed. Note that the gap is not officially closed because the solutions have not actually been implemented and verified but according to the results already presented, the means exist for closing the capability gap.

One question that remains is, what would have happened had the gap not been closed? There are several options available. First of all is to revisit the ideas for materiel/non-materiel approaches in Step V. Within this step there are three ways for the designer to proceed. First, the design ranges for the variables could be modified. Perhaps the space of certain variables was not wholly representative of what is physically possible. Secondly, the designer could add additional variables, either doctrine or

materiel. For example, in this scenario, other variables to study might be communication variables for aircraft materiel properties or multi-ship attack maneuvers for doctrine. Third, in this study, the new materiel programs were based on modifications to existing systems. However, it may be necessary to look into designing an entirely new aircraft from the ground up.

The other option, should the capability gap not yet be closed is to re-examine the capabilities and standards. For example, if none of the designs here were able to meet the 98% threshold and after iterating through the FSA, no other approaches could be found to close the gap, then the question for the designer is, are we willing to move forward with a system that is less than 98% capable? This sort of decision should only be made after *all* options in the FSA have been exhausted.

Alternatively to the acceptable level of a given capability is determining if the capability under investigation is the appropriate one to be measuring against. For example, if it was not possible to meet the 98% threshold, is there an additional capability that we are gaining that could be beneficial? For example, if certain systems are able to hold at least 75% of the total area of operations at risk continuously, that is a substantial capability which may be of value. It does not necessarily translate into 98% targets killed but perhaps there is sufficient need for that capability to continue along the CBA path.

Fortunately, there was no capability deficiency after the first iteration. Therefore, it is now possible to take those desirable cases and formulate recommendations which can then feed the initial capabilities document.

5.8 Step VIII: Recommendations

Now that all the cases have been analyzed and there a sufficient set of designs which are deemed acceptable, it is time to translate this into a form which can be used for a DCR or an ICD. Note that in this dissertation no attempt is made to formally craft a document. Instead, the “best” solutions will be presented as well as an indication of whether doctrine, materiel, or some combination are responsible for the success.

The first solid conclusion actually addresses an approach that should not be carried forward. In particular the notion of adjusting the architecture for dedicated hunter assets and dedicated prosecutor assets proved to be dominated by the set of solutions employing a single asset type. This is likely because a system using two different types will need more aircraft and thus is more costly. Additionally, due to the limited engagement capability of the hunter assets when the functions were divided, the response time to get a weapon on target took longer, reducing the chances for a successful engagement. In a similar vein, it is possible to say that developing the border patrol pattern is not worth the cost as it appeared so infrequently in the set of acceptable solutions.

The first approach which definitely should move forward is the Reaper type aircraft operating as a hunter/killer. Many of these combinations required technology infusion so they may not be available immediately. However, there were instances where an increased number of Reapers with strip or box subdivision and orbit patrol patterns could meet the criteria. One possible idea is to increase the number of Reapers available today and modify their doctrine while concurrently developing technologies to push the envelope even further. This would be an evolutionary, spiral acquisition strategy that could mitigate risk and an example of this was shown in Figure 56.

Another approach to consider is to expand the current fleet of Predator type aircraft. By adapting the doctrine these assets were able to get relatively close to the performance threshold at a low cost. It is possible that additional technologies in the future could achieve the necessary capability and thus this is a valid approach to output from the CBA.

It is easy to underestimate the importance of identifying candidate approaches. However, recall that these precious few approaches were born of a solution space that had literally *billions* of possible options. Furthermore, unlike the current state-of-the-art in doctrine analysis, all of the approaches were analyzed quantitatively with a constant set of assumptions against consistent criteria. This gives the set of solutions moving forward more credibility and provides a traceable history as to how acquisition decisions were made.

CHAPTER VI

CONCLUSION

6.1 *Revisit Research Questions*

Now that the implementation of the methodology has been run to completion, it is time to reexamine the research questions and hypothesis that catalyzed this research. First, for convenience the research questions are presented below.

Research Questions

1. What is the relationship between the doctrinal design variables and materiel design variables in a capability-based aircraft system-of-systems design study?
2. How much and in what way does the inclusion of a quantitative representation of doctrine influence the outcome of a capability-based assessment?
3. How can a method be developed that rigorously compares doctrinal approaches to materiel approaches within the context of a capability-based assessment?
 - a. How can doctrinal variables be quantitatively assessed in such a way that they are comparable to materiel variables?
 - b. How can the exploration of a doctrine-materiel design space be accelerated to enable quantitative analysis of a broader space of possible approaches?
 - c. How can the relative cost difference between different approaches be captured and compared?

- d. How can the expansive design space be best parsed and analyzed so that better information can be output from the capability-based assessment?

Furthermore, the following is a list of the hypotheses that were tested by the experiments in this dissertation:

Hypotheses

1. Doctrine variables will provide greater cost-effectiveness than materiel design variables. Furthermore, the interaction effect between doctrine and materiel will substantially improve the effectiveness of both.
2. The method presented will improve the quantification of doctrine, permitting a side-by-side comparison with materiel solutions. Furthermore, this method will be faster, more transparent, and more consistent than the current state-of-the-art in doctrine analysis.
3. Through the use of
 - a. Agent-based modeling in a constructive simulation
 - b. Morphological matrices, design of experiments, surrogate modeling
 - c. Incorporation of acquisition and operating and support costs
 - d. Filtered Monte Carlo and Multi-attribute Decision Making

this method will rigorously compare doctrinal approaches to materiel approaches for a capability-based assessment. This method will be advance the state of the art in doctrinal analysis by increasing the number of cases assessed and improving transparency of analysis.

First, consider hypothesis number 2. This hypothesis was interested in how the conduct and output of a CBA would change due to the incorporation of doctrinal assessment. First of all, given that mathematical relationships between doctrine variables and capability metrics were developed, it is fair to say that the method developed resulted in a more quantitative study of doctrine. Also, as opposed to wargames or training exercises where tens of cases are evaluated over a period of months, over *100,000* different designs were eventually tested and compared. Furthermore, the method employed surrogate models which provide a unique diagnostic capability in addition to accelerating analyses. Therefore, this doctrine infused CBA was more transparent than past efforts. Finally, since the analysis was done in a constructive simulation, the results are repeatable as opposed to a more qualitative technique that could rely on who is in the room at a time the decision is made.

Based on this information, research question #2 can be answered as follows. The inclusion of doctrine eventually led to a very different set of concepts being selected than would have otherwise occurred without such an inclusion. Furthermore, the manner in which the assessment was executed significantly improved the CBA with respect to speed, quantification, and traceability.

Hypothesis #1 was interested in testing the relationship between doctrine and materiel approaches. As it turned out, doctrine not only provided greater “bang for your buck”, in many cases it provided more “bang” out-right. This was actually counterintuitive to the hypothesis. However, it was shown through the use of the prediction profiler that there is a very strong interaction between doctrine and technology and the precise nature of that interaction can be captured.

Therefore, it is possible to answer research question #1 in the following manner. For a capability-based system of systems study, doctrine and materiel solutions combine to provide complex interactions and unique solutions in a broad possibility space. Furthermore, depending on which doctrine and materiel options are being considered, changes in doctrine can be more cost-effective and even simply more effective than materiel changes.

Finally, the ultimate objective of this research was to develop a methodology to quantify the impact of doctrine and facilitate the comparison of such approaches to traditional materiel approaches in a capability-based assessment. Given that Research Questions 1 and 2 were both answered with new insights that were not available previously, it is possible to conclude that indeed this method is capable of quantifying doctrine. Additionally, the ability to rapidly and fairly compare a huge number of possible approaches means that this methodology is appropriate for most capability-based assessments and advances the state of the art in how CBA's are executed.

6.2 Summary of Methodology

This dissertation began with a motivational argument that a methodology for doctrinal analysis at the CBA stage of acquisition was needed. Next, a set of research questions were posed which guided a comprehensive literature review of doctrine, defense acquisition, and a Next Generation Unmanned Aerial System problem. Four major technical challenges were identified as barriers to successful execution of a doctrine-materiel CBA. For each of these challenges, a solution was hypothesized in the form of techniques and methods from across disciplines including: aerospace engineering, systems engineering, biology, and economics. These techniques were integrated into a capability-based assessment method based on current Department of Defense guidance for the requirements of CBAs. Then, a proof-of-concept implementation of the methodology was executed on the Next Generation UAS problem.

To conclude and summarize the methodology, this section will focus on the steps and enablers of the method. In particular, it will be how the contributions of this dissertation truly enable the successful execution of a doctrine-materiel CBA.

First, in response to the four technical challenges identified earlier as well as additional challenges encountered during the implementation, this methodology used several key techniques and enablers to meet the stated research objective. However, it should be noted that the selection of these techniques and their synthesis into a single methodology is in and of itself a major hypothesis. To illustrate this point, consider Table 14 below. This table is another morphological matrix, however this time, instead of containing all the possible system attributes for a UAS, it contains all the possible enablers which could constitute a doctrine-materiel CBA process. The left-most column

contains the elements of the methodology that required new techniques: Problem Definition, Analysis Type, Identification of Approaches, Simulation Environment, Analysis Techniques, Surrogate model Type, and Visualization method. The techniques and enablers which were used in this dissertation are highlighted in green.

Table 14: Morphological Matrix of Methods

Matrix of Methods						
Category	Options					
Problem Definition	Subject Matter Experts (SMEs)	UJTL	Doctrine Manuals			
Analysis Category	Combat Experience	Military Exercise	Modeling and Simulation	Qualitative	Wargames	
Identification of Approaches	Combat Experience	Functional Decomposition	Literature Survey	Morphological Matrix	SME	
Simulation Environments	FLAMES	MATLAB	NetLogo	OPUS	SEAS	STORM
Analysis Techniques	Agent-Based Modeling	System Dynamics	Petri Nets	Wargame	Live Exercise	Survey of Combat
Surrogate Type	Response Surface Equation	Neural Network	Gaussian Process	Kriging	Radial Basis Functions	
Visualization of Results	Prediction Profiler	Multivariate Plot	Tabular Format			

Just as with the system attribute matrix which had many possible combinations, the matrix of methods contains over 40,000 different methodologies. The first half of this dissertation explained why modeling and simulation, agent-based modeling, NetLogo, neural networks, prediction profilers, and multivariate plots were ideally suited to this problem. The execution of the proof-of-concept problem and the results of that study, including insight into the doctrine-materiel design space as well as improvements in quantification and analysis speed, reinforce the hypothesis that the proposed methodology is appropriate for doctrine-materiel CBAs.

Combining the selected enablers with the basic CBA process, the new methodology is presented graphically in Figure 57.

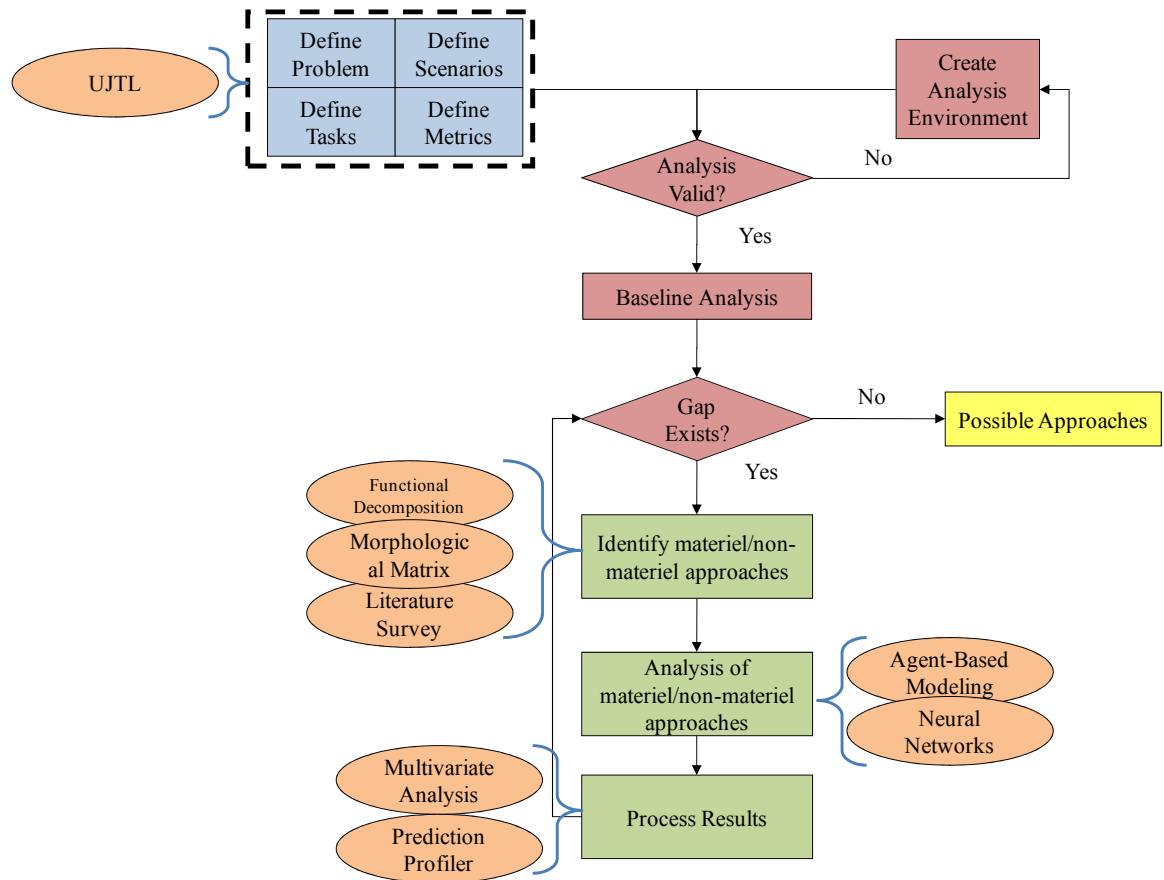


Figure 57: Doctrine/Materiel Methodology with Enablers

This process chart contains the same structural elements as Figure 6, however the yellow ovals represent the key enablers and they are shown supporting the steps of the CBA methodology. The complete process will now be described in step-by-step form.

Step I: Problem Definition – The problem shall be defined in terms of tasks, scenarios, and metrics. A key enabler for this to happen is incorporation of current guidance from the Universal Joint Task List and any service specific task lists. This step shall remain as solution independent as possible, with the possible exception of any baseline system. This criterion is essential for capability-based acquisition as capabilities are independent of the means used to achieve them and the results must not be biased from the onset.

Step II: Create Analysis Environment – In the case where no current analysis environment exists, one must be created. The criteria for the analysis environment are driven by the problem definition. Any environment must be responsive to the functional area analysis and provide the necessary information to measure the appropriate metrics and tasks. For capability-based assessments, there is a broad space of possibilities therefore constructive modeling and simulation is an invaluable tool. Capturing doctrinal behavior is achieved through agent-based modeling.

Step III: Baseline Analysis – The baseline system or systems shall be analyzed with respect to the appropriate scenario(s). Again, this step does NOT entail any exploratory analysis. This step can involve verification and validation of analysis environments.

Step IV: Gap Analysis – Once the baseline analysis is complete, the results shall be compared to all relevant capabilities. All effort must be made at this step to gain the deepest possible insight into: first determining the presence or absence of a capability gap; second quantifying the extent of any gaps; and third providing information on the cause of any capability gaps.

Step V: Identify Materiel/Non-Materiel Approaches – If a capability-gap does exist, then this step investigates candidate approaches to close the gap. The first step towards successfully identifying approaches is to recognize how truly expansive the space of potential approaches can be. A morphological matrix is an ideal tool for accomplishing this. In order to populate the matrix of approaches, there are two key enablers to employ. First, a literature survey of doctrine manuals, concepts of employment, concepts of operations, white papers on relevant missions, or any other available sources can collect a wide array of ideas. This search can identify officially sanctioned procedures or platforms

to consider, or it can identify categories of solutions (such as type of aircraft, or patrol pattern) which can then be populated by the analyst. Secondly, a functional decomposition of the mission can provide insight into approaches. By decomposing the mission into segments and eventually into basic entities and actions, alternative concepts can be identified.

Step VI: Analysis of Materiel/Non-Materiel Approaches – Once all of the possible approaches have been identified, the approaches shall be quantified for goodness with respect to the metrics identified in Step I. In order to accomplish this, the analysis environment developed in the FNA for the baseline/gap analysis shall be used. An appropriate level of fidelity must be used to capture the necessary physics and behavior. Agent-based modeling provides a unique means for quantifying the changes in behaviors of entities within a simulation. In order to effectively evaluate the expansive space of approaches identified in Step V, a design of experiments shall be used to intelligently sample the design space. With the information generated by the design of experiments, surrogate representations of the original analysis environment can be created which enable increased analysis speed as well as transparency into the analysis environment. The behavior a doctrine-materiel design space is highly non-linear and discrete, rendering traditional surrogate techniques ineffective. Therefore, neural network surrogate models shall be used. Then, through the use of prediction profiles, rapid parametric trades can be performed in a bottom-up manner; examining possible approaches much faster than in the original analysis environment. Extending this concept, the surrogates can be executed many times in a known statistical way through a process known as Monte Carlo simulation. This enables a top-down exploration of the design space where the analyst

can identify not only the goodness of one particular combination, but all possible combinations that meet the specified threshold of goodness. Visualization and manipulation of the design space is enabled through the use of multivariate plots.

Step VII: Reassess Gap – After performing the analysis and reducing the data, the results of the FSA shall be compared to the standards prescribed in the FAA. In the event the gap is still not closed, the process iterates with new ideas or new analysis environments. This step is also enabled by the visualization and data processing capability of the prediction profiler and multivariate plots.

Step VIII: Recommendations – Once potential approaches to closing the gap have been identified, the CBA methodology outputs recommendations to the rest of the defense acquisition process. These recommendations can take the form of an ICD if a new materiel program is required, or a DCR if a doctrinal change is needed.

6.3 *Limitations of Methodology*

The methodology presented in this dissertation was demonstrated for a specific proof-of-concept problem that typifies the class of problems encompassed by capability-based assessments. However, in order for the powerful results demonstrated herein to be applied to other future problems it is important to establish what problems the method is suited for. What are the limitations? When should this method be used? These are important questions which will be addressed in this section.

Foremost among the applicability issues is determining the set of problems for which this methodology is valid. The methodology is not limited to only aircraft design problems or Air Force assets; any branch of the military undertaking a capability-based assessment could follow use this approach. In fact, organizations outside of the military would also benefit greatly from this methodology. Any organization with equipment and guidance for employment of that equipment technically has materiel and doctrine whether they call it that or not.

Currently, the methodology is not scripted to handle aspects of DOTMLPF other than doctrine and materiel. However, the adaptations needed to accomplish a quantitative analysis of organization or training could be minor. Many of the problem definition steps and data reduction steps could be preserved while a determination as to the applicability of agent-based modeling and neural networks is investigated.

Another potential limitation is that no investigation was made regarding basic level doctrine. It was shown that tactical and operational doctrine can be captured through this methodology but questions still remain for basic doctrine. Part of what makes the

investigation into basic doctrine more uncertain stems from how fundamental and wide-sweeping basic doctrine is.

The next question regarding limitations of the methodology applies to the scalability. The term scalability refers to how well the methodology performs as the number of variables and/or number of entities. Of these two scalability problems, first examine how the number of entities affects the execution.

There are several ways that the number of entities affects scalability. First, as the number of entities in the simulation increases, each entity must be instantiated in the simulation. As it turns out, the method performs very favorably in this regard. The creation of large numbers of agents is relatively simple and the methodology can create 1000 of a single agent-type just as easily as it can create 1. In all likelihood, multiple agent types will be needed for any realistic simulation. Creating multiple types of agents does require upfront effort in creating the code necessary for generating the agents, their behaviors, and other properties. However, once a class of agents is defined, creating more agents of that class incurs minimal penalty.

The other way in which number of entities affects scalability is simulation run-time. Each agent must interact with the other agents in the simulation as well as the environment and the simulator must execute the actions of each and every agent. In order to test the affect the number of agents has on simulation time, a series of cases were prepared in the NetLogo simulation environment in which the number of hunter aircraft was varied between 1 and 100. The reason that the number of hunters was varied is that the hunter aircraft are always active in the simulation; they do not appear and disappear as the targets do. Each simulation had the same random seed and same number of targets.

Every simulation was run for the same amount of simulation time, namely the 2000 units of time in NetLogo representative of a single engagement. The time which was measured was how much real time elapsed during the execution of the simulation. The results of this study are plotted below in Figure 58.

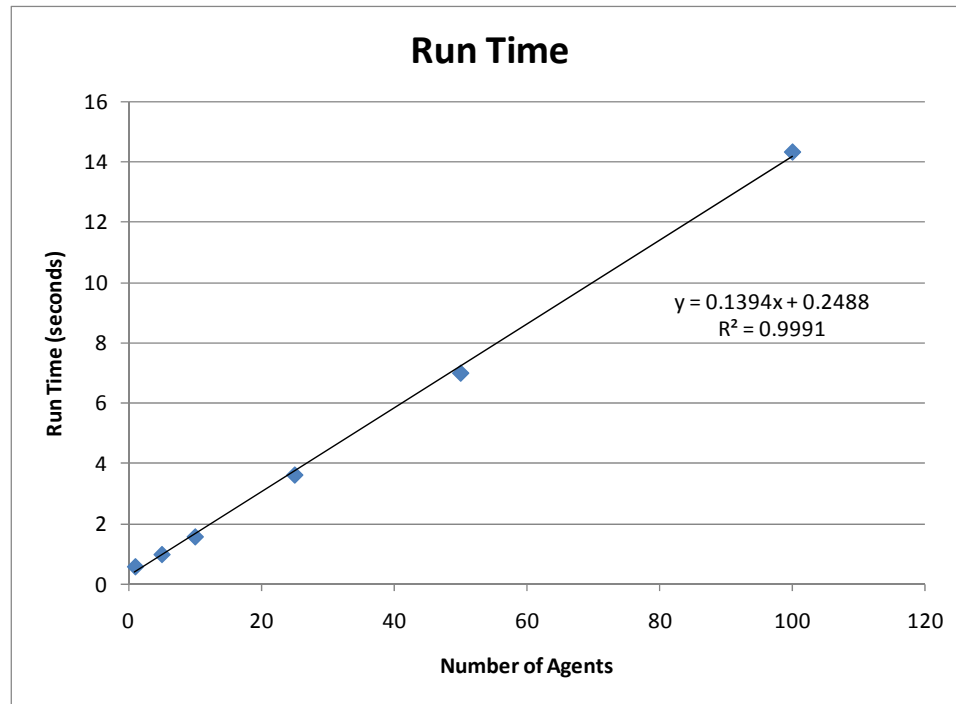


Figure 58: Simulation Run Time

The points in Figure 58 represent the six cases in the study with number of hunter agents equal to 1, 5, 10, 25, 50, and 100. A trend-line was created based on these runs and it shows a very linear relationship between number of hunter agents and computer run time. Each additional agent adds just over one-tenth of a second to the total simulation time. Using this relationship to predict larger values, one finds that the run time with 1000 agents would be approximately 140 seconds. Testing this case using the same conditions as those used for Figure 58 above obtained a run time of 212 seconds.

It is important to note that all of the cases executed here took place on a Dell Inspiron 6400 laptop computer with an Intel Core 2 Duo processor 2GHz and 2 GB of RAM. Therefore, the run times listed are meant to show trends rather than provide an absolute computational time. If a faster computer were used or the system was optimized for running simulations the run times would be less. Additionally, the simulations cases could be distributed across multiple computers reducing the “real-time” run time though not the total computer time.

At the rate estimated above the total number of entities in the simulation to make the execution time of a single case reach one hour per run would be over 25,000 entities. For a single case to take a full day, the simulation could have over 600,000 entities. For some historical context, if one were trying to simulate the aircraft Battle of Midway from World War II, there were approximately 600 total aircraft. The much longer Battle of Britain featured approximately 6,000 total aircraft. For a naval example, the Battle of Jutland featured 250 ships of various types. Finally, the Battle of Antietam of the Civil War was fought among 130,000 troops. Now, none of these figures are presented to suggest how long it would take to simulate any of these engagements. The simulation times shown above were for a particular scenario and each of these scenarios are quite different. However, these historical battles are some of the largest of all time and provide a frame of reference when it is stated that running one case per hour could contain 25,000 entities. For a capability-based assessment, it is unlikely that a full scale battle such as one of the aforementioned examples would be simulated. Therefore, while the number of entities in the simulation increases run time, for all practical purposes, the number of agents is unlikely to limit the applicability of this methodology.

While the number of entities influences the execution time for a single simulation that is only half of the equation. The other aspect to consider is: how many simulations need to be run? The total number of required simulations depends on how many variables are being studied and the level of detail required to estimate the impact of a particular variable on a given response. For example, assume that there are n variables under investigation and each variable has L settings or levels being considered. The total number of combinations is expressed as L^n . In order to gain an appreciation of how sensitive the number of combinations is to number of variables and number of levels, examine Table 15 below.

Table 15: Full Factorial Number of Cases

Number of Variables	Number of Cases		
	2 Levels	3 Levels	4 Levels
2	4	9	16
3	8	27	64
4	16	81	256
5	32	243	1,024
6	64	729	4,096
7	128	2,187	16,384
8	256	6,561	65,536
9	512	19,683	262,144
10	1,024	59,049	1,048,576
11	2,048	177,147	4,194,304
12	4,096	531,441	16,777,216
13	8,192	1,594,323	67,108,864
14	16,384	4,782,969	268,435,456
15	32,768	14,348,907	1,073,741,824
16	65,536	43,046,721	4,294,967,296
17	131,072	129,140,163	17,179,869,184
18	262,144	387,420,489	68,719,476,736
19	524,288	1,162,261,467	274,877,906,944
20	1,048,576	3,486,784,401	1,099,511,627,776

It is clear from Table 15 that the number of possible combinations becomes unmanageably large very quickly. Assume for a moment that the simulation in question contained 100 agents and therefore had an execution time of around 15 seconds. Also assume that for practical purposes all of the simulation cases needed to be completed in one day. This would limit the number of variables which could be investigated. For 2 levels, only 12 variables could be considered, for 3 levels, only 7 variables, and for 4 levels, only 6 variables.

Obviously most practical engineering problems will have many variables of interest. It is because of this fact that the design of experiments concept was employed in this methodology to reduce the number of cases required. Another option for reducing the design space is to coordinate the modeling efforts with lessons learned from actual combat experience or wargaming. Those cases can serve as starting points which are then explored upon through this quantitative methodology.

In summary, this methodology, like any other, cannot and should not be blindly applied to a problem. The methodology is applicable to a wide range of problems including capability-based analysis with numbers of agents much larger than the primary investigation in this dissertation. This discussion of limitations illustrates why the definition of scenarios and metrics is so important. Through this step the analyst learns what needs to be modeled and that guides how many entities, variables, and levels are ultimately chosen.

6.4 Future Work

This research was never meant to be the final chapter in the evolution, study, or interest in doctrine. While there were many interesting and important conclusions, there are also many questions and ideas for future research.

First of all, this dissertation started out by lumping doctrine into the larger subset of non-materiel solutions in DOTMLPF. Clearly there is work to be done on all of these non-materiel solutions. This methodology contains elements that are applicable to the analysis of training or facilities, or organization. Perhaps the variables are selected a certain way, the analysis technique may no longer be agent-based modeling, but the steps to guarantee quantification, traceability, and fair comparison are still valid. It is anticipated that application of this method to other non-materiel problems could yield equally insightful conclusions regarding those approaches.

Another possible future work is to examine a radically different scenario to expand the database of doctrinal studies. This was only one test case and the more information that can be gleaned about the behavior of different doctrines and the interactions with technology the better.

Finally, a very pressing item of future work is to apply the methodology for the same problem, but use higher fidelity/classified information. The outputs are only as good as the inputs and in this case the inputs had to be estimated from publically available sources. Furthermore, some of the more detailed tactics, techniques, and procedures are not available. Comparing true values of materiel approaches and classified doctrinal concepts would provide answers for the high valued Next Generation UAS problem that served as the test bed for the methodology

6.5 Research Conclusions

The research presented in this paper is not meant to develop new doctrine for the Air Force or recommend a new materiel approach. Rather, the goal is to advance the state-of-the-art in how a capability-based analysis is done through addressing the non-materiel aspect, specifically doctrine. There were many conclusions to be drawn from this research beyond just answering the research questions.

First, a major conclusion is that doctrine can play an enormous role in system-of-systems problems. Doctrinal variables led to vast improvements in capability at very reasonable cost values. This does not actually guarantee that doctrine is *always* superior to materiel. There are several possible exceptions. One is that the variables and ranges did not highlight enough of the possible space for materiel approaches. Put another way, only evolutionary technologies were considered, not revolutionary technologies. In fact, depending on which doctrine and materiel variables are chosen, the design space could look different. However, what is undeniable is that doctrine is a key player. History bears this fact out and now that it is possible to rigorously quantify the extent to which doctrine is a player, significant advancements can be made in acquisition of new systems.

One item to note is that it has been stated doctrine provided significant improvement over the baseline configuration. However, this raises the question of how accurate was the baseline? To reiterate, no classified or for official use information was used in this dissertation. Therefore, the values of the baseline approach could be off. However, regardless of what the baseline configuration is, the fact remains that the spread of doctrinal solutions was vastly more expansive than that of materiel solutions. Whether or not doctrine improved or degraded performance relative to the presumed baseline, there

were many cases that resulted in a quantum leap or fall in capability. This means that moving forward, it is essential to explore doctrinal combinations or provide sufficient evidence for the selected doctrine. Otherwise, there could be whole portions of the design space that were not covered.

The most fundamental lesson learned from this research is that doctrine has a role in conceptual design. This notion may have been qualitatively supposed but this research finally showed how to incorporate doctrine as well as the type of results achievable. In fact, for this proof-of-concept study, technological changes were essentially marginal. They weren't unimportant, just overshadowed by doctrine. This result may be counterintuitive but it shouldn't be ignored. In fact, this result should prompt a new paradigm in thought regarding the creation of capabilities. There is a fundamental question that emerges: can revolutionary capability be achieved without a change in doctrine? That is to say, will changes in technology, no matter how great, only result in evolutionary advancements in capability if doctrine does not adapt? Evidence from this study suggests that revolutionary advancements cannot be accomplished through technology alone.

The most fundamental contribution of this research is an advancement in the state of the art in how capability-based assessments are conducted. Agent-based modeling was capable of handling different level of doctrine quantitatively. Systems engineering was able to define and manage a complex design space. This methodology paves the way for integrating the efforts of soldier, commander, analyst, and engineer to address the acquisition problem from all fronts.

Applying this methodology can provide better information for acquisition decisions. The ultimate goal is to field the appropriate mix of ways and means to implement national security policy. Maintaining quantitative doctrinal analysis as a central tenet of design will lead to reduced realization times, less expensive capabilities, and ultimately an improved acquisition process that benefits the United States military as well as the country it protects.

APPENDIX A

DOTMLPF Discussion

Up to this point in the paper, the term DOTMLPF has been presented as the complete spectrum of materiel and non-materiel solutions the military considers. However, before the research problem can be scoped, it is necessary to fully define each option. Additionally, the following sections will provide examples of work being done in the defense community today within each area. These examples will augment the definition and begin to establish context for making a decision about which option to explore further. Finally, to make that choice, a review of the pros and cons for each option is presented.

Doctrine

The term doctrine is not a trivial word to define. People from different walks of life define it differently, and even people within the same field might have different working definitions of doctrine. Since this research is motivated by the Defense Department's need to analyze non-materiel approaches, it is appropriate to begin with the official DoD definition. According to the DoD Dictionary of Military and Associated Terms, doctrine is defined as the "fundamental principles by which the military forces or elements thereof guide their actions in support of national objectives. It is authoritative but requires judgment in application." Doctrine is related to both the actions that military units

perform and the overall goals of the nation. The part of the definition that emphasizes the use of judgment is in the definition because The Defense Department furnishes an online resource, Defense Acquisition University. Within this resource is a feature entitled Ask A Professor where DoD experts respond to public inquiries. One such inquiry concerned the terms of DOTMLPF and the simplistic definition for doctrine is “the way we fight.” Such a simple phrase carries a lot of meaning but a good synonym for doctrine might be guidance. Doctrine is guidance, not just instructions, that directs the way the military fights.

Organization

Organization is a much less amorphous term than doctrine. As the name suggests, it is the way the military organizes to fight. Organization is the arrangement of forces, not physically on the battlefield per say, but rather how the military is arranged in terms of command structure or unit composition. Organization applies to all levels of the Defense Department. At the highest level, for example, is the organization of the Department of Defense, shown in Figure 59. This illustrates how the DoD is organized into the distinct services, Defense Agencies, the Joint Chiefs of Staff and so on.

Department of Defense

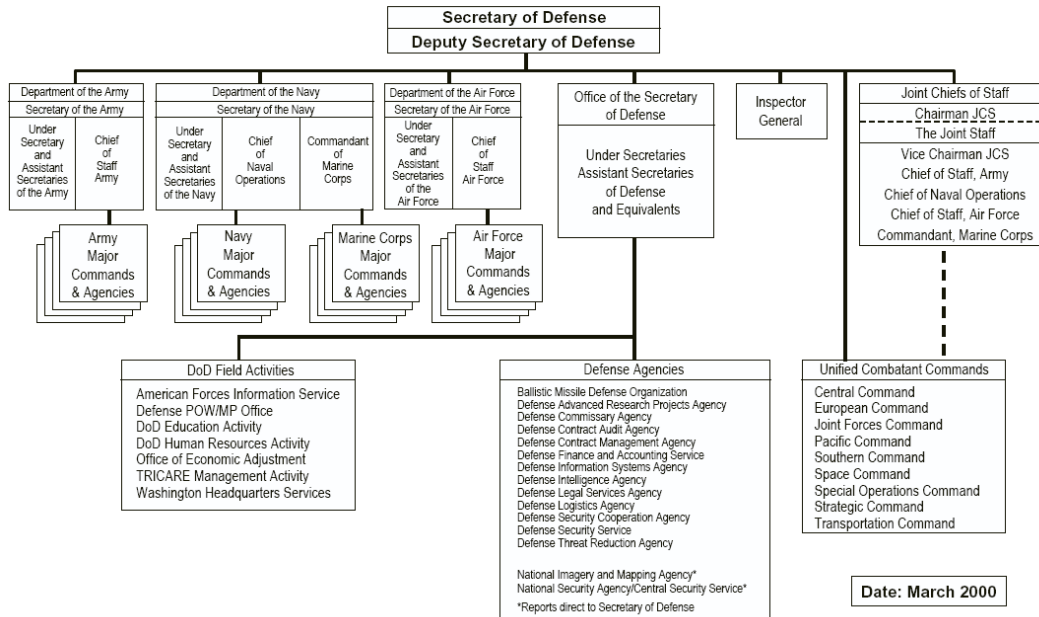


Figure 59: Department of Defense Organizational Chart

Organization also defines the elements within the military branches. For example, consider the Air Force. Overall the Air Force is broken down into “nine major commands, 35 field operating agencies, four direct reporting units and their subordinate elements”. Examining the Air Force organization from the bottom-up begins with the most basic element: the flight. Two or more flights then form a squadron. Squadrons are the lowest organizational level with a dedicated headquarters. Two or more squadrons make a group. Two or more groups form a wing. A wing is a sizeable collection of force. Wings can be comprised of several thousand military and civilian personnel and are commanded by Colonel or Brigadier General ranked officers. Wings can then be grouped into Numbered Air Forces and/or Major Commands. The difference is that Numbered Air Forces are

generally used only in time of conflict while Major Commands are groupings based on common mission or attribute.

How does a military branch determine its organizational structure? Consider again as an example the Air Force. Air Force Instruction 38-101 provides the official policy and information regarding organization of the U.S. Air Force. It states that there are five principles that are used to determine organization. First is the notion that the structure of the service should be such that in the event of wartime, no additional modification to the organization is needed. Basically, the forces should be configured in such a way that when they need to deploy, time is not spent making sure the right pieces go together. The Air Force must be ready to go at a moment's notice. Secondly, the Air Force is organized into groups that share a common objective, such that each element within the group complements the other elements. Think of it like the parts on a body. Each part has a specific function and the appropriate parts must work together to accomplish a goal. Third, organization is done in such a way that decisions are made as rapidly as possible without going through a lot of "middle-men". Fourth is something referred to as "Skip-Echelon Structure." Basically this means that not every level of the Air Force has the amount of staff. For example, squadrons and wings have a full and complete staff that can perform all of the possible tasks needed. However, recall that multiple squadrons form a group and multiple groups form a wing so in terms of organization, it is possible to say that groups fall between squadrons and wings. The group level in between squadrons and wings will not have the complete staffing. There will be command elements for the group but the idea is to cut down on the amount of paperwork shuffling

done to accomplish a particular task. Finally, the last principle by which the Air Force organizes is a standardization of levels. If some parts of the Air Force were not configured according to the flight-squadron-group-wing standard, then that would inhibit interoperability and collaborative operations.

Now it is important to understand the current issues and work being done with organization within the military. The primary area of work being done relates to the first principle identified in the paragraph above: being ready to fight at a moment's notice. It is not that the military cannot deploy quickly, but rather the issue is how quickly the military can deploy configured appropriately for the mission at hand. The modern military was designed principally to deal with a major nation-state threat such as the former Soviet Union. However, today's military finds itself combating a dramatically differently enemy. Insurgents, terrorists, and similar threats are characterized by smaller more agile forces and dramatically shorter lead times where the U.S. military can gain intelligence on their intentions. Accordingly, all services of the U.S. military are performing a reorganization to create force structures that can more readily respond to this new type of threat.

Training

According to the Defense Acquisition University, training, as the name implies, is how the military prepares to fight. At one end of the training spectrum is the basic training every soldier receives upon entering the military. In the Army, this is called Basic Combat Training (BCT) and it lasts 9 weeks. Here the recruits learn about what it takes to

be a soldier including physical fitness and mental discipline. At the complete opposite end of the spectrum are large multi-service and even multi-national joint training exercises. In 2006 the U.S. military completed Valiant Shield; a training exercise that involved the Navy, Marines, Air Force, and Army with over 22,000 participants. In between these two ends of the training spectrum are training programs that provide the warfighter their specialties, practice working as a unit, and so on.

Most of the training referred to in the previous paragraph involves all of the participants to gather in the same location and physically undertake the exercises. However, this can be a costly endeavor since real fuel is burned, real weapons expended, and the time and logistics involved in assembling so many people can be quite large. In order to rectify this, some military training is being done in the virtual world. Virtual training could be pilots practicing maneuvers in a flight simulator on their air base, or it could be many personnel doing a joint exercise similar to Valiant Shield mentioned above only this time the participants are distributed to simulation centers across the country or world. The goal of training is to prepare military personnel to perform real missions and the military advocates using whatever methods possible to achieve that end.

The question to ask now is what research is being undertaken to advance the state of the art of military training? In order to understand this issue however, first consider the reasons why changes might be required of the military training regime. First, it is possible that the soldiers are not getting the appropriate training. This could occur any time there is a technology change. As a simple example, hundreds of years ago when the firearm

replaced the bow and arrow, armies across the globe needed to teach their infantry how to handle this new weapon. Likewise today, as the military constantly seeks to develop new technologies soldiers must be trained in the appropriate weapons and methods of war. Also, if there is a shift in the threat, as happened with the dissolution of the Soviet Union, or the proliferation of weapons of mass destruction, then the mission changes and the military must prepare its personnel to deal with the new mission. Another reason to modify the training program is that the soldiers are not receiving enough training. Starting November 2nd, 2007, the U.S. Army is extending its basic training program to 10 weeks on a temporary basis to determine if the extra week of training improves advanced individual training and eventually combat effectiveness.

Materiel

This category is all about the physical tools and equipment that the military uses to fight. It covers a broad range and includes anything from submarines, tanks, and bombers; to torpedoes, shells, and bombs; to propellers, treads, and engines; to nuts, bolts, and screws. Materiel can be large systems or it can be parts and spares. Additionally, materiel includes new technologies that can be incorporated into existing systems.

Historically, materiel is the area of the military that has received the most research and development. In fact, one of the guiding documents listed in the motivation of this paper states that the military feels it can do materiel analysis pretty well. Since the focus here is to examine new methods and techniques for enabling the analysis of non-materiel approaches, no further discussion of the materiel category is needed. While the non-

materiel analysis will eventually be compared to materiel changes, no research will be conducted into new methods for materiel analysis.

Leadership and Education

The meaning of this category might not be as obvious as some of the others so it is important to get a clear definition. Leadership here does not refer to whether or not the military has enough leaders or the hierarchy of command. Rather, leadership and education is a professional development category. The military must make certain that its leaders are given the appropriate tools necessary to effectively command in any situation. This category is concerned with the types of things that the leaders are taught. One might suggest that this is no different than the training category discussed above, however, there are in fact important differences. Training is concerned with the tactical execution of the missions. The goal of training is to make sure that a soldier has the appropriate physical and mental skills to do their job. Leadership and education is a different type of training; a type that assumes the soldier has the basics needed to do his or her job, and then builds on those basics by developing an individual capable of directing others in their tasks.

There are similarities between training and leadership in that both must adapt with changing missions and technology. Therefore, it is to be expected that the leadership education is changing with the changing military reality in a similar manner to how training is evolving. One example of research in this area is a recent paper developed by the RAND Corporation under sponsorship of the U.S. Army. This study assessed how the Army currently develops leaders as well as what some of the key differences in today's

threat environment are compared to years passed. Recommendations included an emphasis on rapid and adaptive decision making, more dedicated time in careers for development, and using new technology to improve leadership training.

Personnel

According to the Defense Acquisition University, the term personnel refers to having a sufficient number of qualified individuals to perform the set of functions required by the military. Notice that there is more to that definition than simply the total number of soldiers in the service. The inclusion of the terms “qualified” and “functions” indicates that personnel means having the right person do the appropriate job. Consider as an example some enemy aircraft on an attack run against a U.S. airbase that houses a wing of F-15s. If there are no pilots, or less pilots than there are aircraft, the defense of the airbase cannot be performed. Furthermore, assume for the moment there are a sufficient number of pilots at the base. If all the pilots are qualified to fly B-52s instead of F-15s then the capability is similarly degraded.

What are some types of research or work being done regarding personnel in the military today? Well first of all is the obvious debate regarding how many people are needed in the military and if an all-volunteer service is sufficient.

Facilities

As the name implies, facilities are the installations that support military operations. They can range from forward airbases, to command and control centers, to basic training bases,

and even to production plants owned by the military. Why might facilities be an important consideration for the military? Well if a new aircraft was developed that provided some essential capability but required a runway length far greater than that offered by most U.S. airfields, then the utility of that aircraft is minimized and there are many theaters where it could not operate. As another example, consider the types of operations performed in Afghanistan. The terrain there is extremely mountainous and if there were no U.S. bases in mountainous terrain with large enough land to host joint training exercises, then the combat situation would be the first time soldiers were exposed to that type of environment and performance would likely be suboptimal.

APPENDIX B

Taxonomy of Variables

Part of the difficulty with working with doctrine relates to how doctrine can be captured into quantifiable variables instead of remaining an abstract concept. In order to accomplish this, it is essential to develop a taxonomy of variables that describes what the possible categories of variables are. The following discussion identifies types of variables and provides examples to further illustrate the differences.

Doctrinal variable – A doctrinal variable is any parameter that describes the operations, actions, decisions, and employment of an asset or a collection of assets. However, there are several distinct levels of doctrine which must each be addressed.

Basic doctrine variable – Parameter which characterizes fundamental warfighting principles

- Target priority
- Autonomy

Operational doctrine variable – Parameter which describes conduct of an operation

- Assignment of units
- Number of assets

Tactical doctrine variable – Parameter which describes behavior or employment of fundamental elements

- Speed
- Patrol pattern

- Orientation to target to begin attack sequence

Beyond doctrinal variables, there are several other categories of variables which are closely related and therefore must be differentiated. The first of these are materiel variables.

Materiel variables – Parameter which describes an asset’s physical properties

- Take-off gross weight
- Max speed
- Payload capacity

Additionally, organizational concepts are another aspect of DOTMLPF which closely relates to doctrine.

Organizational variables – Parameter which describes the types of military organizations, their size, composition, and command hierarchy

- Number of assets in a wing
- Commander type

Simply defining those categories is not sufficient. As one might assume from reading the above definitions, there are very tight relationships between these variable types. Sometimes, the relationship between two types of variables is so close that two variables are confounded. This means you cannot determine which type of variable (for example doctrine or materiel) is being examined. Without further delineation, a confounded variable cannot be specified and therefore cannot be accurately analyzed.

The first way variables can be confounded is along doctrine-materiel lines. One example of this is speed. When speed refers to a characteristic of an asset, such as

maximum speed, it is a materiel variable. When speed refers to the speed at which an aircraft will fly a given mission, then that refers to a doctrinal variable.

Confounding can also occur between doctrine and organization. One example of this is number of assets. As an organizational parameter this could detail how many assets constitute a wing or a squadron or so forth. As a doctrinal parameter, the number of assets provides details about the operations and execution of a mission. So a set of organizational variables says a squadron is no more than 6 aircraft reporting to a single forward air controller. An operational level doctrine would then specify that 4 aircraft will be used.

It is also possible to get confounding within levels of doctrine. For example, target priority can exist at all levels of doctrine. At the basic level, a doctrinal concept such as “establish air supremacy” can be captured through assigning priority values to different targets and prioritizing anti-air targets highest. At the operational level target priority can be used to describe which anti-air assets in a particular theater or mission are the highest priority. And at the tactical level, an asset can have different priorities for targets that it is supposed to prosecute. Only operational and tactical level doctrine variables were examined. To truly change basic level doctrine, the change must come from changes in national strategy or through widespread and unanimous success as an operational concept.

When using a confounded variable, it is important to specify the context to give insight into which type of variable is ultimately being investigated. Examining one aspect of a confounded variable will entail an assumption across the other degrees of confounding.

There are several conclusions that can be made from this taxonomy.

1. Doctrine variables are unique in that they describe actions and employment of assets.
2. Properly defining which category a particular variable falls into is important because that can influence the analysis.
3. Also, properly defining the variable type will enable a more accurate extraction of conclusions from the results as you can identify the type of change required in accordance with defense acquisition guidance.
4. When using a confounded variable, it is important to specify the context to give insight into which type of variable is ultimately being investigated. Examining one aspect of a confounded variable will entail an assumption across the other degrees of confounding.
5. Including other types of variables beyond materiel makes the design space vastly more expansive. An even greater challenge is that, while the set of materiel parameters is somewhat fixed (e.g. an aircraft has a takeoff gross weight, an aspect ratio, a ceiling, etc), the set of doctrinal variables is much more fluid and may not be constant.

APPENDIX C

Stepping Through A Simulation

The simulation starts by opening the NetLogo environment. A snapshot of the environment is shown below. The large black area on the right hand side is the “world”. This is a visualization aid that will show the agents executing all aspects of the simulation. On the left-hand side are several boxes; three grey ones at the top and a series of green boxes beneath. The grey boxes are the buttons used to initiate function calls in the simulation. As the name implies, the “setup” box sets all of the entities and environmental parameters in the simulation. The box labeled “go once” executes a single time step of the simulation when it is pressed. Alternatively, the box labeled “go” with two circular arrows executes time steps of the simulation repeatedly until the stopping criteria for the simulation are satisfied. The set of green boxes on the left-hand side are the input variables for the simulation. Prior to pressing the “setup” button, each of these variables should be specified.

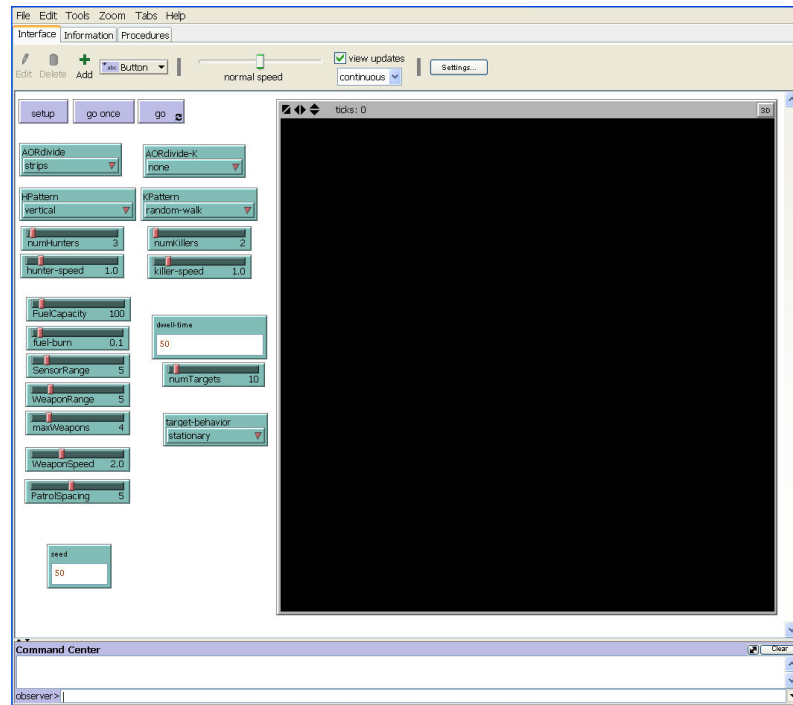


Figure 60: Basic NetLogo Display

The figure below shows the environment after the setup function has been executed. Within the “world” a green square has formed which represents the area of operations. Also within the world, in the lower right-hand corner, a small green square with blue figures has appeared. This area is the base from which the aircraft stage their mission. For this particular scenario, there are 4 hunter aircraft, one killer aircraft, and a controller located at the base. Note that throughout the entire simulation, the controller will remain at this location while the aircraft must return here to refuel and rearm.

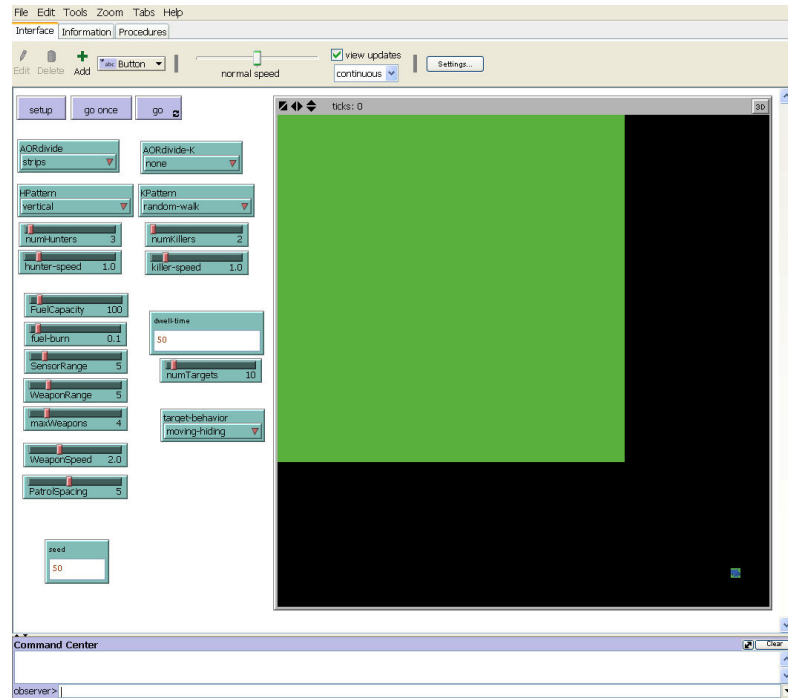


Figure 61: Initialized Environment

The next image below show the aircraft patrolling the area of operations. The four hunter aircraft are indicated in blue while the killer aircraft is represented in yellow. Behind each aircraft is a line showing the path they have take up to this point. The hunter aircraft are executing a parallel search pattern with an entry point division while the killer aircraft is executing a random patrol.

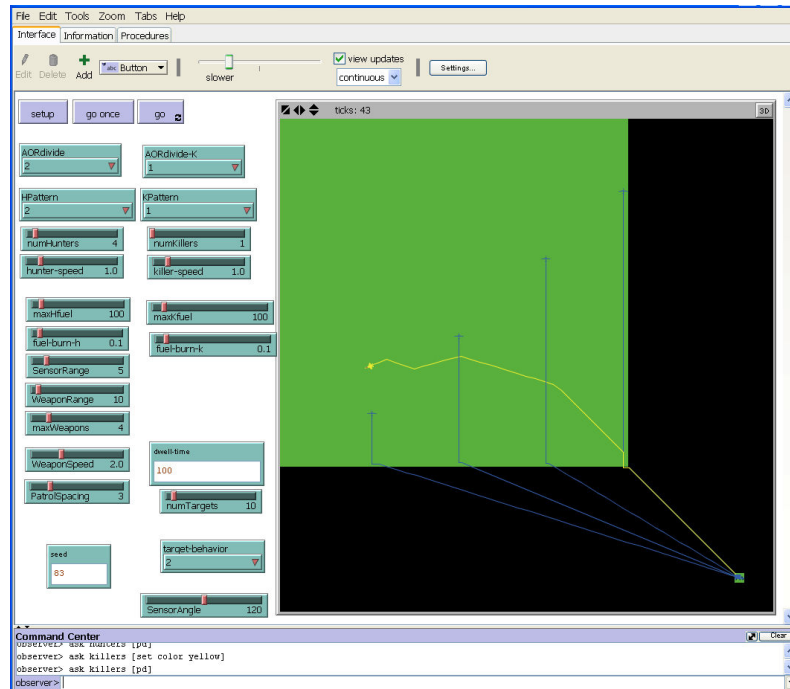


Figure 62: Aircraft Patrolling Area of Operations

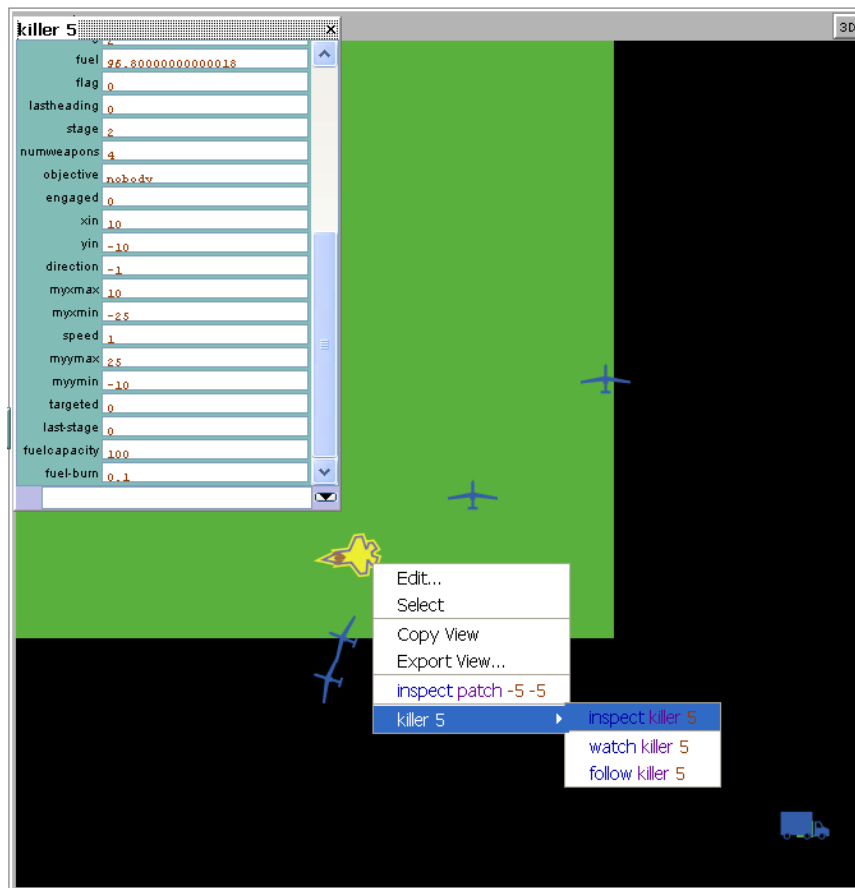


Figure 63: Querying Agents

Figure 63 above illustrates more about what is happening behind the scenes during an execution of the simulation. Again the hunter aircraft are blue while the killer aircraft is yellow. Now however, the size of the agents has been greatly exaggerated to aid in visualization. In this image, the killer aircraft is highlighted and a series of menus appear. This option allows the user to explore or manipulate parts of the world. In this example, the option to “Inspect Killer 5” has been chosen and a dialogue box with the title ‘killer 5’ appears in the upper-left portion of Figure 63. The name ‘killer 5’ is the identifier in the NetLogo environment; each agent has one and the identification number is unique to each agent. The identification numbers begin at zero and each successive agent, regardless of type, is assigned an identifier.

The ‘killer 5’ dialogue box contains an entry for all of the variables specific to that agent as well as the values of those variables. For this class of agents, there are 32 variables which the agent uses to monitor itself and the environment in order to make decisions and perform actions. Examples include: fuel remaining, weapons remaining, asset speed, bounds of area of operations, and target objective among others. In this case, the target objective is ‘nobody’ meaning that no target has been assigned to this killer aircraft. There are also variables that are used to plot the agent’s movement. Each hunter and killer aircraft is executing a certain type of patrol pattern and throughout the patrol the agent will use its current position, the bounds of its patrol area, its fuel state, and instructions from the controller to guide future actions.

The next item to examine is how the simulation changes with the addition of targets. In Figure 64 below there are two frames representing two sequential time steps during which a target appears. On the left-hand side of Figure 64 the hunter and killer agents are

executing their respective maneuvers and there are no targets to be seen. On the right-hand side, a red vehicle has appeared in the lower-left portion of the area of operations. This appearance is the visual representation of the target “emerging” from its fictitious hiding place. However, this does not mean that the hunter agents can now see the target. Rather, the user who has an omniscient view of the world can see the target and this means it is *possible* for the hunters to find the target.

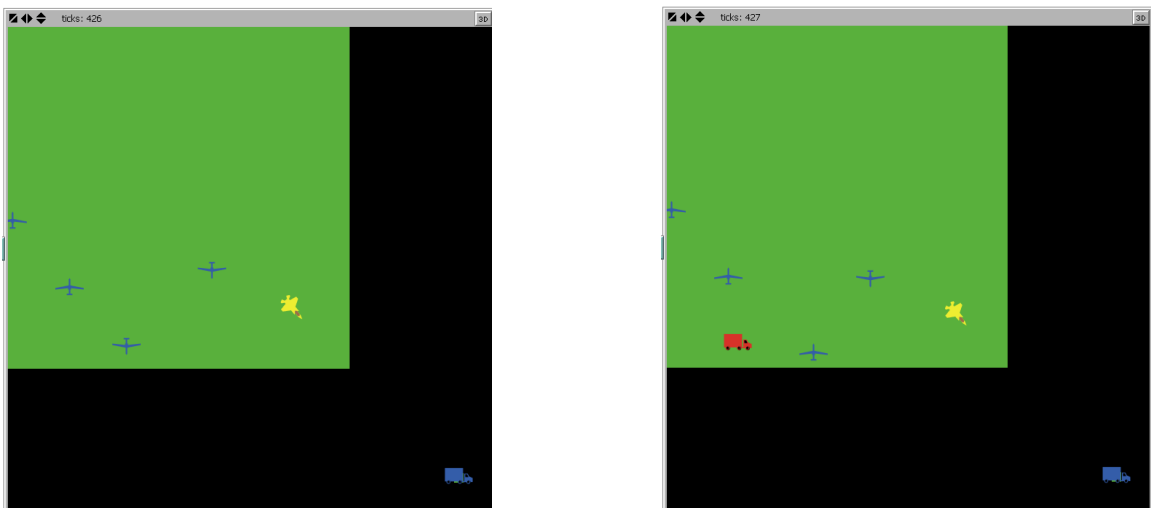


Figure 64: Appearance of a Target

Even though it is possible to now find the target, there is no guarantee that will happen. Target detection is dependent on many factors including how much ground the hunters can cover and the relative motion of the targets and the hunters. In Figure 65 below, several timesteps have been simulated since the target first appeared in Figure 64. The paths of each agent are drawn out behind them with lines colored in accordance with the color of the agent type. Figure 65 contains a great deal of information which provides insight into the execution of the simulation and warrants further explanation.

First, examine the blue hunter agents and the paths traced out by their patrols. All four hunters are performing the parallel search which means that they will move forward within the area of operations along the top-to-bottom dimension. When the aircraft are within a distance of the edge of the area of operations where the next time step would carry them past the boundaries of the area of operations, the agents execute a turn. The turn moves the agent over by an amount defined as the patrol spacing from the inputs and reverses the heading 180 degrees. This pattern proceeds until the agent is sufficiently close to the edge of the area of operations that another turn would take them beyond the boundary. At this point, the agent turns so that it is aligned with the boundary of the area. Also at this point, an internal variable tells the agent that the next turn it makes should be in the opposite direction. Examining the path of a single agent, for example the right-most hunter in Figure 65, shows that the parallel search is gradually moving the aircraft to the right in the area of operations. Once the aircraft reaches the edge of the area, it will reverse its turns and execute a parallel search in the opposite direction.

Also of note in Figure 65 regarding the hunter agents, there are a series of blue patches located in front of each hunter. These patches represent the sensor area covered by the hunter aircraft. For this simulation the sensor range is 5 units and the sensor field of view is 120 degrees. Therefore, the blue patch extends 5 units out from the aircraft and to 60 degrees off the heading of the aircraft on both sides. The sensor coverage area appears as blocks due to the visualization resolution of NetLogo however the actual computations are done to a much higher degree of precision and a more appropriate visualization of the sensor coverage is overlaid with the black wedge on the right-most hunter. At each timestep the hunter aircraft tries to determine if there is a target located

within its field of view. In this figure, there are no hunter aircraft which have yet detected the target.

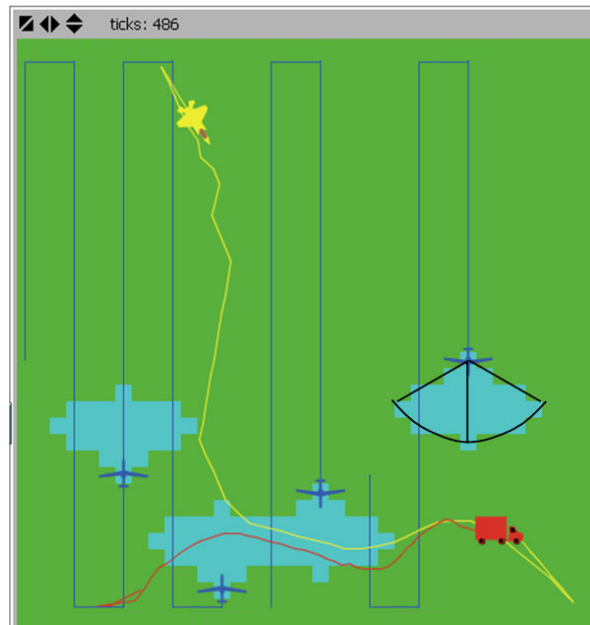


Figure 65: Agents Executing the Mission

Next, consider the yellow killer aircraft in Figure 65. The path traced out by this aircraft is random and covers the entirety of the area of operations. At each time step, the killer aircraft checks with the controller to see if there is a target for it to pursue. When there is no target, the aircraft continues on its patrol. The random path is random in two ways. First, the agent randomly determines when it will change heading and secondly, when there is a heading change, the agent randomly chooses how much heading change to implement. Now the random choice is bounded to an absolute value of 45 degrees of heading change at any one turn. This constraint was implemented to prevent the agent from implementing radical maneuvers that would be unrealistic. Just as the killer agents execute their random walk, so too do the targets. The logic is the same although the different classes move at different speeds. The targets are “dumb” in that they do not

sense the presence of aircraft and therefore do not execute maneuvers to avoid detection. This is an assumption of target behavior but given that the current generation of hunter/killer assets are used for missions such as prosecuting truck convoys or terrorists, the targets can be unsophisticated in their countermeasure capability and thus the assumption has a sound practical basis.

Figure 66 below shows the state of the world just prior to target acquisition. The target remains just outside of the sensor range of one of the hunter aircraft and the killer aircraft still has no objective, as seen in the ‘killer 5’ dialogue box on the right-hand side of the image.

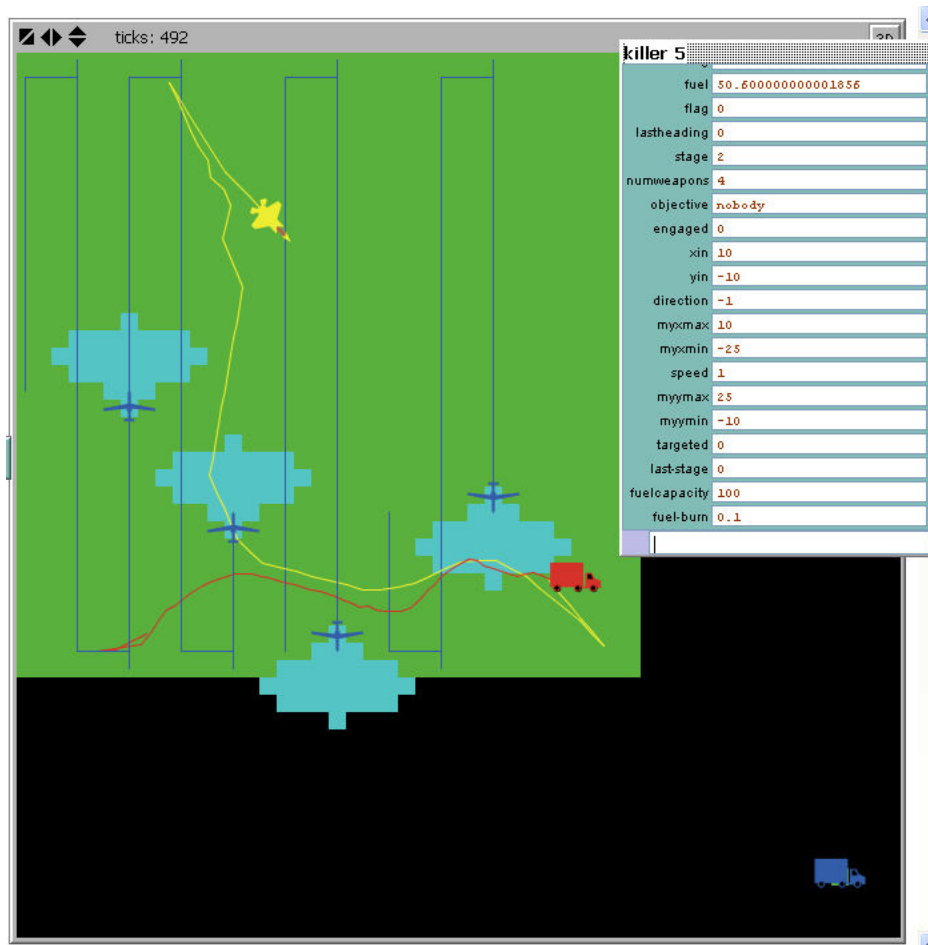


Figure 66: Nearing Target Acquisition

Finally, in Figure 67, as one of the hunter aircraft makes its turn, the target is detected. This image is only an instantaneous picture but there is a great deal happening here. First, the hunter aircraft registers that a target has been found and then stores the identifier for that target in its internal objective variable. Next, the hunter relays that information to the controller. For this example there are no other targets in the controllers queue however if there were, the controller would prioritize the list of targets to build a tasking order for dissemination to the killer aircraft. Since there is just the single located target, after the controller receives the information from the hunter aircraft it passes this information along to the killer aircraft. If there were multiple killer aircraft in the simulation, the controller would assign the target to the closest unengaged asset.

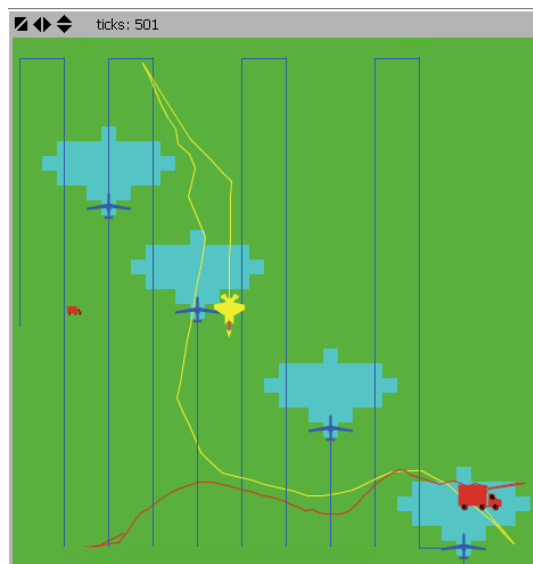


Figure 67: Target Acquired

After the target acquisition, the next phase of the mission is prosecuting the target. Figure 68 shows a series of three snapshots of various stages of the engagement. First, on the far left, the killer aircraft has oriented itself to the target. It is assumed that the killer

aircraft knows the location of the target once it has been found. Also at this point, the killer's internal indicator has changed to notify the controller that it is currently engaged in the pursuit and prosecution of a target and the aircraft should not be assigned a new target, regardless of the priority of the new target.

At each point in the pursuit function, the killer aircraft calculates the distance between itself and the target and then compares this distance to the maximum range of its weapon. If the distance is less than or equal to the weapon range, the aircraft fires its weapon. The middle image in Figure 68 shows that that aircraft has closed to a point sufficiently close to the target that the weapon can be launched. The weapon takes information about the target from its parent aircraft and hones in on the target. On the far-right of Figure 68, the weapon has reached and destroyed its target. At this point, the killer aircraft breaks off the attack and commences a random patrol. Additionally, the target clears its objective variable and engagement indicator meaning that it is now available for more missions. Note that for the hunter aircraft, once they locate a target and pass the information to the air controller, they no longer carry the target's information in internal variables. However, the hunters will communicate with the controller so that if any hunter "finds" a target that has already been discovered, it does not count as a separate detection.

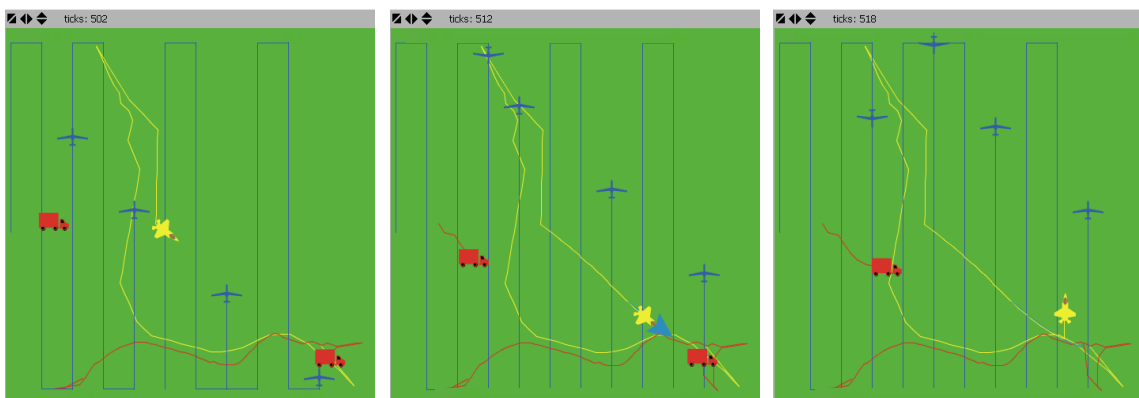


Figure 68: Pursuit and Engagement

Throughout this engagement there are several data collecting algorithms taking place. First, each time a hunter finds a target, a tally is incremented to count the total number of targets found in the simulation. Also, when the killer aircraft launches its weapon, another tally is incremented to count the total number of weapons launched in the simulation. Not all weapons are guaranteed to kill their target as targets can disappear before the weapon impacts. When a target is destroyed, another counter notes that and records the total number of targets killed. Finally, every time an aircraft returns to base, the amount of fuel expended by that aircraft is recorded and added to a running total so the total value of fuel used in the simulation can be counted. Recall that the fuel capacity and fuel burn rate of each class is defined as input variables.

With respect to fuel and weapon tracking, there are also individual monitors on the agents which keep track of how much fuel remains and how many weapons remain, if applicable. When the weapons are expended or the fuel status is sufficiently low, the aircraft return to base. Figure 69 below shows an aircraft returning to base once its weapons have been expended. When an aircraft is egressing from the area of operations it is incapable of finding or prosecuting targets. Similarly, aircraft that are ingressing back into the area are not capable of finding or prosecuting.

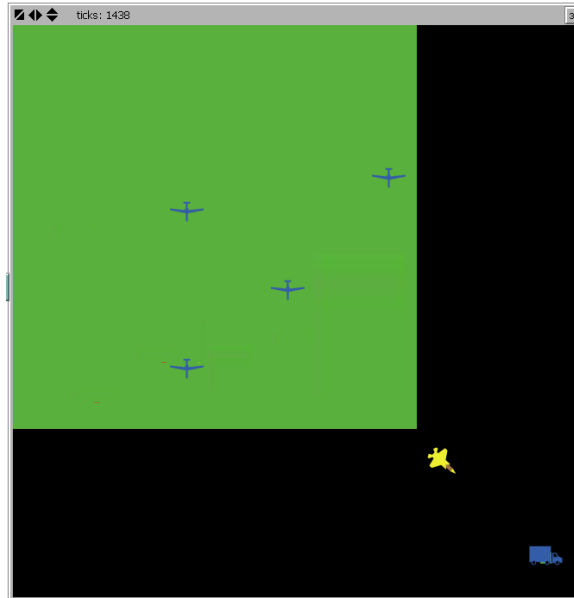


Figure 69: Returning to Base

The steps outlined above are repeated until the time horizon of 2000 units of time has been met. All targets are programmed to appear randomly such that they are exposed for the complete dwell time before the simulation stops. Once the simulation stops, all of the counter information regarding targets found, destroyed, fuel expended, and weapons fired is output.

The above example demonstrated the execution of a simulation given a fleet of four hunter aircraft with parallel search patterns and a single killer aircraft and a random search pattern. The process would be similar in the case with only a single aircraft type performing all mission functions.

APPENDIX D

Analysis Code

This section contains all of the NetLogo analysis code broke up by file name.

Doctrine.nlogo

```
;;Doctrine.nlogo
;;Created by Steven Tangen

;;Include files:
;;divide.nls - operational level doctrine, how the AOR is divided for multiple aircraft
;;targetprocedures.nls - the routines that the targets execute to act in the simulation
__includes["divide.nls" "targetprocedures.nls" "allpatterns.nls"]

breed [hunters hunter]
breed [killers killer]
breed [caocs caoc]
breed [tcts tct]
breed [amraams amraam]
breed [stingers stinger]

globals [
xmin
xmax
ymin
ymax
x
y
xbase
ybase
numCaocs
TotalFuel
TotalWeapons
```

tcts-created ;;This is how many have been created so far at any point in the sim
tcts-finished ;;This is how many have been "finished" either by being killed or by
finishing their dwell time
tcts-killed ;;This is how many have been killed by the HKs
born-times ;;This is the list of the times that tcts will be created
blue-killed
failed-case
]

hunters-own [
leg
fuel
flag
lastheading
stage
numWeapons
objective
engaged
xin
yin
direction
myxmax
myxmin
speed
myymax
myymin
targeted
last-stage
]

killers-own [
leg
fuel
flag
lastheading
stage
numWeapons
ktarget
objective
engaged
xin
yin
direction
myxmax
myxmin
]

```
caocs-own [  
  ctarget  
]
```

```
tcts-own [  
  life  
  targeted  
  priority  
  die-time  
  target  
  engaged  
]
```

```
amraams-own [  
  power  
  target  
  parent  
]
```

```
stingers-own [  
  target  
  parent  
  counter  
]
```

```
;;Setup function and accompanying subfunctions-----  
to setup  
  clear-all  
  random-seed seed  
  create-base  
  create-AOR  
  make-caocs  
  make-hunters  
  make-killers  
  set tcts-created 0  
  if target-behavior = 1 or target-behavior = 3 or target-behavior = 4 [make-tcts set tcts-  
  created numTargets] ;;this is the stationary behavior  
  if target-behavior = 2 [ ;;this is the moving/hiding behavior  
    set born-times n-values numTargets [random (2000 - dwell-time)] ;;2000 - dwell-  
  time  
    set born-times sort born-times]  
  set numCaocs 1  
  set tcts-finished 0  
  set tcts-killed 0  
  set blue-killed 0
```

end

to create-AOR

```
set xmin min-pxcor
set xmax 10
set ymin -10
set ymax max-pxcor
set x xmin
set y ymin
while [x < xmax + 1]
  [while [y < ymax + 1]
    [ask patch x y [set pcolor green]
      set y y + 1]
    set y ymin
    set x x + 1]
```

end

to create-base

```
set xbase max-pxcor - 3
set ybase min-pycor + 3
ask patch xbase ybase [set pcolor green]
set TotalWeapons 0
```

end

to make-caocs

```
set-default-shape caocs "truck"
create-caocs numCaocs [
set color blue
setxy xbase ybase
set ctarget []]
```

end

to make-hunters

```
set-default-shape hunters "rq-4a"
create-hunters numHunters [
set color blue
setxy xbase ybase
divide ;;Call the algorithm to divide the total AOR into sub-areas for a/c to patrol
facexy xin yin
set fuel FuelCapacity
set numWeapons maxWeapons
set flag 0
set lastheading 0
set leg 1
set stage 1
set objective nobody
```



```
set direction -1
set speed hunter-speed]
end
```

```
to make-killers
set-default-shape killers "f-35a"
create-killers numKillers [
set color blue
setxy xbase ybase
set heading 90
set lastheading heading
set direction -1
set fuel FuelCapacity
set numWeapons maxWeapons
set flag 0
set stage 1
set ktarget nobody]
end
```

```
to make-tcts
set-default-shape tcts "truck"
create-tcts numTargets [
set color red
let tempxcor random (xmax - xmin)
let tempycor random (ymax - ymin)
setxy (tempxcor + xmin) (tempycor + ymin)
set targeted 0
set target nobody]
end
```

```
;;Set up function and accompanying subfunctions-----
```

```
;;Go procedures-----
```

```
to go
if ticks > 2000 or tcts-finished = numTargets or (blue-killed = numHunters and not
any? amraams) [ask hunters [set TotalFuel TotalFuel + (FuelCapacity - fuel)] stop]
ask hunters [fly]
if target-behavior = 2 [
ask tcts [evade]
manage-tcts]
if target-behavior = 3 [
ask tcts [red-search shoot-back]
;;manage-tcts
]
if target-behavior = 4 [
```

```

    ask tcts [evade red-search shoot-back]]
  weapon-fly
  SAM-fly
  tick
end

to fly
  ask self [
    if stage = 1
      [ingress]
    if stage = 2
      [Hpatrol
        if any? tcts in-cone SensorRange SensorAngle [ask self[set objective min-one-of tcts
[distance myself]] set stage 3]
        if objective = nobody and (xcor > myxmax or xcor < myxmin or ycor > myymax or
ycor < myymin) [set stage 1]
        if fuel - (distancexy xbase ybase) * fuel-burn / speed <= 1 or numWeapons = 0 [set
flag 0 set stage 4]
        if targeted = 1 [set last-stage stage set heading heading - 180 set stage 6]
        ]
      if stage = 3 [
        if objective = nobody [
          ifelse(xcor > myxmax or xcor < myxmin or ycor > myymax or ycor < myymin) [set
stage 1] [set stage 2]]
          ;;Hpatrol
          fire
          if targeted = 1 [set last-stage stage set heading heading - 180 set stage 6]]
        if stage = 4
          [egress
            if targeted = 1 [set last-stage stage set heading heading - 180 set stage 6]]
        if stage = 5
          [refuel]
        if stage = 6
          [avoid
            if targeted = 0 [set stage last-stage]]
          ]
      ]
    ]
  end

to ingress
  ask self [
    if distancexy xin yin < speed
      [set xcor xin
        set ycor yin
        set heading lastheading
        fd speed
        set fuel fuel - fuel-burn

```

```

    ;;set leg 1
    set stage 2
    stop
  ]
  facexy xin yin
  fd speed
  set fuel fuel - fuel-burn
]
end

```

```

to egress
  ask self [
    if flag = 0
      [set xin xcor
        set yin ycor
        if heading mod 90 = 0 [set lastheading heading]
        facexy xbase ybase
        set flag 1]
    if flag = 1
      [fd speed
        set fuel fuel - fuel-burn
        if distancexy xbase ybase < 1
          [set xcor xbase
            set ycor ybase
            set stage 5]
      ]
  ]
end

```

```

to refuel
  ask self [
    ;;wait 4
    set TotalFuel TotalFuel + (FuelCapacity - fuel)
    set fuel FuelCapacity
    ;;set TotalWeapons TotalWeapons + (maxWeapons - numWeapons)
    set numWeapons maxWeapons
    set flag 0
    facexy xin yin
    set stage 1
  ]
end

```

```

to pursue
  ask self [
    face objective
    fd (1.25 * speed)
  ]
end

```

```

    set fuel fuel - (1.5 * fuel-burn)]
end

```

```

to fire
  ask hunters with [objective != nobody] [
    ifelse distance objective > WeaponRange [pursue]
      [if numWeapons > 0 and engaged = 0 [
        hatch-amraams 1 [
          set target [objective] of myself
          set color sky
          set parent myself
        ]
        set numWeapons numWeapons - 1
        set TotalWeapons TotalWeapons + 1
        set engaged 1]
      ]
    if engaged = 1 [pursue]
  ]
end

```

```

to weapon-fly
  ask amraams [
    if target = nobody [if parent != nobody [ask parent [set engaged 0 set flag 0]]
      die]
    ifelse distance target <= WeaponSpeed
      [move-to target
       ask target [die]
       set tcts-killed tcts-killed + 1
       set tcts-finished tcts-finished + 1
       if parent != nobody [
         ask parent [
           set engaged 0
           set stage 2
           ifelse flag = 1 [set flag 0 set heading lastheading] [set heading lastheading]]]
       die]
      [face target
       fd WeaponSpeed]
  ]
end

```

```

to avoid
  ask self [
    ifelse abs(xcor - xmax) < 1 or abs(xcor - xmin) < 1 or abs(ycor - ymax) < 1 or abs(ycor
- ymin) < 1
      [set heading heading + (180 - random 10)
       fd 2]

```

```
[ifelse random 3 < 2
  [fd (1.2 * speed)
   set fuel fuel - (1.5 * fuel-burn)]
  [set heading heading + (random 45 * (-1)^((random 2) + 1))
   fd (1.2 * speed)
   set fuel fuel - (1.5 * fuel-burn)]]
]
End
```

Allpatterns.nls

;;This is the file that contains all of the patrol patterns.
;;The first procedure decides reads from the main file which patrol pattern will be used
and then calls the correct procedure below.

```
to Hpatrol
  if HPattern = 1 [random-walk]
  if HPattern = 2 [patrol-vertical]
  if HPattern = 3 [orbit]
  if HPattern = 4 [box-patrol]
end
```

```
to Kpatrol
  if KPattern = 1 [random-walk]
  if KPattern = 2 [patrol-vertical]
  if KPattern = 3 [orbit]
  if KPattern = 4 [box-patrol]
end
```

;;Patrol vertical involves the aircraft flying in a "lawnmover" fashion within their
specified area. The legs run vertically, the turns are made
;;at the extreme top and bottom of the area.

```
to patrol-vertical-old
  ask self [
    ;;Leg 1 is meant to the "start", going from one edge to the other. Fixes the problem of
    having the "direction" reset every step along the edge.
    if leg = 1 [
      if (myymax - ycor <= speed and heading = 0) or (ycor - myymin <= speed and
heading = 180)[
        set xcor xcor + direction * PatrolSpacing
        set heading heading + 180
        fd speed
        set fuel fuel - fuel-burn
        set leg 2]]
    ;;if leg = 2 [
      ;;if myymax - ycor <= speed or ycor - myymin <= speed [
        ;;set xcor xcor + direction * PatrolSpacing
        ;;set heading heading + 180
        ;;fd speed
        ;;set fuel fuel - fuel-burn
        ;;set leg 3]]
    if leg > 1 [
      if myymax - ycor <= speed or ycor - myymin <= speed [
        ifelse abs(xcor - myxmax) < PatrolSpacing or abs(xcor - myxmin) < PatrolSpacing
```

```

    [ifelse direction = 1 [let tempmove (myxmax - xcor) set xcor xcor + direction *
tempmove]
        [let tempmove (xcor - myxmin) set xcor xcor + direction *
tempmove]

    set heading heading + 180
    set fuel fuel - fuel-burn]
    [set xcor xcor + direction * PatrolSpacing
    set heading heading + 180
    fd speed
    set fuel fuel - fuel-burn]]
    if (xcor = myxmax or xcor = myxmin) and leg = 2 [
        set direction direction * -1
        set leg 1]]
    fd speed
    set fuel fuel - fuel-burn
]
end

```

```

to patrol-vertical
ask self [
    fd speed
    set fuel fuel - fuel-burn
    if myymax - ycor <= speed or ycor - myymin <= speed [
        ifelse xcor + direction * PatrolSpacing < myxmin or xcor + direction * PatrolSpacing
> myxmax
            [ifelse direction = 1 [let tempmove (myxmax - xcor) set xcor xcor + direction *
tempmove]
                [let tempmove (xcor - myxmin) set xcor xcor + direction * tempmove]
            set direction direction * -1
            ifelse abs(myymax - ycor) < abs(ycor - myymin) [set ycor myymax] [set ycor
myymin]
                set heading heading + 180
                fd speed
                set fuel fuel - fuel-burn]
            [set xcor xcor + direction * PatrolSpacing
            ifelse abs(myymax - ycor) < abs(ycor - myymin) [set ycor myymax] [set ycor
myymin]
                set heading heading + 180
                fd speed
                set fuel fuel - fuel-burn]]
    ]
end

```

;;Random Walk is to have the aircraft just move aimlessly around the AOR

```

to random-walk
  ask self [
    if leg = 1 [
      set heading 315
      fd 3
      set fuel fuel - fuel-burn
      set leg 2]
    if leg = 2 [
      ifelse abs(xcor - myxmax) < 2 or abs(xcor - myxmin) < 2 or abs(ycor - myymax) < 2
      or abs(ycor - myymin) < 2
      [set heading heading + (180 - random 10)
      fd 2
      set fuel fuel - fuel-burn]
      [ifelse random 3 < 2
      [fd speed set fuel fuel - fuel-burn]
      [set heading heading + (random 45 * (-1)^((random 2) + 1))
      fd speed
      set fuel fuel - fuel-burn]
      ]
    ]
  ]
end

```

;;Orbit means the aircraft will orbit a specified point. Leg 1 computes the point to orbit about based on the assigned area for that aircraft.

;;Leg 2 take the aircraft from the entry point in the region to the orbit point.

;;Leg 3 moves directly up half the length of a normal side. Leg 4 moves to the left half the length of a normal side.

;;Leg 5 is the normal orbit. The side length is specified in the code. The orbit is CCW.

```

to orbit
  ask self [
    let side PatrolSpacing
    if leg >= 5 [
      fd speed
      set fuel fuel - fuel-burn
      set leg leg + 1 / (side * speed)
      if leg >= 6 [set leg 5 set heading heading - 90]]
    if leg >= 4 and leg < 5 [
      fd speed
      set fuel fuel - fuel-burn
      set leg leg + 1 / (side * speed / 2)
      if leg >= 5 [set heading 180 set leg 5]]
    if leg >= 3 and leg < 4 [
      fd speed
      set fuel fuel - fuel-burn
      set leg leg + 1 / (side * speed / 2)
    ]
  ]
end

```



```

    if leg >= 4 [set heading 270 set leg 4]]
  if leg = 1 [
    set xin (myxmax - myxmin) / 2 + myxmin
    set yin (myymax - myymin) / 2 + myymin
    facexy xin yin
    set leg 2]
  if leg = 2 [
    ifelse distancexy xin yin <= speed
      [set xcor xin
        set ycor yin
        set heading 0
        set leg 3]
      [fd speed
        set fuel fuel - fuel-burn]]
  ]
end

```

;;The box patrol goes around the edges of the sub-AOR's.

```

to box-patrol
  ask self [
    if leg = 1 [
      if heading = 0 [set leg 2]
      if heading = 90 [set leg 5]
      if heading = 180 [set leg 4]
      if heading = 270 [set leg 3]]
    if leg = 5 [
      ifelse myxmax - xcor <= speed
        [set xcor myxmax
          set heading 0
          set leg 2]
        [fd speed
          set fuel fuel - fuel-burn]]
    if leg = 4 [
      ifelse ycor - myymin <= speed
        [set ycor myymin
          set heading 90
          set leg 5]
        [fd speed
          set fuel fuel - fuel-burn]]
    if leg = 3 [
      ifelse xcor - myxmin <= speed
        [set xcor myxmin
          set heading 180
          set leg 4]
        [fd speed
          set fuel fuel - fuel-burn]]
  ]
end

```

```
if leg = 2 [  
  ifelse myymax - ycor <= speed  
    [set ycor myymax  
     set heading 270  
     set leg 3]  
    [fd speed  
     set fuel fuel - fuel-burn]]  
set lastheading heading]  
end
```

Divide.nls

```
;;Different ways to divide up an AOR.  
;;The first procedure decides reads from the main file which division/assignment  
algorithm will be used and then  
;;calls the correct procedure below.
```

```
to divide  
  if AORdivide = 1 [no-divide]  
  if AORdivide = 2 [basic-divide]  
  if AORdivide = 3 [strip-divide]  
  if AORdivide = 4 [box-divide]  
end
```

```
;;Basic divide takes the number of aircraft and the total width of the AOR and calculates  
evenly spaced entry points into the AOR. It does NOT  
;;assign a bounded region to each aircraft. That is, they will begin patrolling at their  
specified points and then progressively cover all  
;;of the AOR.
```

```
to basic-divide  
  ;;calculates ingress points for the aircraft to start. Does not specify where they will stop;  
they patrol whole box  
  set xin ceiling(((count hunters - (who - numCaocs)) / (count hunters)) * (xmax - xmin) +  
xmin)  
  set yin ymin  
  set myxmax xmax  
  set myxmin xmin  
  set myymax ymax  
  set myymin ymin  
end
```

```
;;Strip divide takes the number of aircraft and the total width of the AOR and generates  
sub-areas for each aircraft to patrol. The entry points  
;;for the aircraft are the same as in the basic divide but now a particular aircraft will not  
patrol into the area being patrolled by another  
;;aircraft. That is to say, they will stop at the point where the next aircraft entered the  
whole AOR. The sub-areas are equal width areas and  
;;are extend the complete length of the AOR in the vertical direction.
```

```
to strip-divide  
  ;;want to divide the total AOR into n strips.  
  set xin ceiling(((count hunters - (who - numCaocs)) / (count hunters)) * (xmax - xmin) +  
xmin)  
  set yin ymin  
  set myxmax xin  
  set myxmin ceiling(((count hunters - ((who - numCaocs) + 1)) / (count hunters)) *  
(xmax - xmin) + xmin)
```

```

set myymax ymax
set myymin ymin
end

```

;;No divide is a baseline condition where we put all of them in at the same spot.

```

to no-divide

```

```

  set xin xmax
  set yin ymin
  set myxmax xmax
  set myxmin xmin
  set myymax ymax
  set myymin ymin
end

```

```

to box-divide

```

```

  if numHunters = 1 [
    set xin xmax
    set yin ymin
    set myxmax xmax
    set myxmin xmin
    set myymax ymax
    set myymin ymin]
  if numHunters = 2 [
    set xin ceiling(((numHunters - (who - numCaocs)) / (numHunters)) * (xmax - xmin) +
xmin)
    set yin ymin
    set myxmax xin
    set myxmin ceiling( ( (numHunters - (who - numCaocs) + 1) ) / (count hunters)) *
(xmax - xmin) + xmin)
    set myymax ymax
    set myymin ymin]
  if numHunters > 2 [
    ifelse numHunters mod 2 = 0
      ;;If we have an even number of hunters
      [set xin ceiling(((numHunters / 2 - (floor(who / 2) - numCaocs)) / (numHunters / 2)) *
(xmax - xmin) + xmin)
        set myxmax xin
        set myxmin ceiling(((numHunters / 2 - (floor(who / 2) - numCaocs + 1)) /
(numHunters / 2)) * (xmax - xmin) + xmin)
        ifelse who mod 2 = 1 ;;If the current hunter has an odd who number
          [set yin (ymax - ymin) / 2 + ymin
            set myymin yin
            set myymax ymax]
          [set yin ymin
            set myymin ymin
            set myymax (ymax - ymin) / 2 + ymin]]
    ]

```

```

;;If we have an odd number of hunters
[ifelse who mod 2 = 0 ;;If the current hunter has an odd who number
  [set yin (ymax - ymin) / 2 + ymin
    set xin ceiling(((ceiling(numHunters / 2) - (floor(who / 2) - numCaocs)) /
ceiling(numHunters / 2)) * (xmax - xmin) + xmin)
    set myymin yin
    set myymax ymax
    set myxmax xin
    set myxmin ceiling(((ceiling(numHunters / 2) - (floor(who / 2) - numCaocs + 1)) /
ceiling(numHunters / 2)) * (xmax - xmin) + xmin)]
  [set yin ymin
    set xin ceiling(((floor(numHunters / 2) - (floor(who / 2) - numCaocs)) /
floor(numHunters / 2)) * (xmax - xmin) + xmin)
    set myymin yin
    set myymax (ymax - ymin) / 2 + ymin
    set myxmax xin
    set myxmin ceiling(((floor(numHunters / 2) - (floor(who / 2) - numCaocs + 1)) /
floor(numHunters / 2)) * (xmax - xmin) + xmin)]
]
End

```

Targetprocedures.nls

```
;;Target procedures
to evade
  ask self [
    ifelse abs(xcor - xmax) < 1 or abs(xcor - xmin) < 1 or abs(ycor - ymax) < 1 or abs(ycor
- ymin) < 1
      [set heading heading + (180 - random 10)
        fd 2]
      [ifelse random 3 < 2
        [fd 0.5]
        [set heading heading + (random 45 * (-1)^((random 2) + 1))
          fd 0.5]
      ]
  ]
end
```

;;I want to create targets at random times and in random locations. I want a fixed dwell time.

;;They should be created, evade while alive, and then die after their dwell time.

```
to manage-tcts
  if tcts-created < numTargets [
    if ticks = item 0 born-times [
      while [not empty? born-times and ticks = item 0 born-times] [
        make-a-tct
        set born-times but-first born-times]]]
  ask tcts [
    if ticks = die-time [
      set tcts-finished tcts-finished + 1
      die]]
end
```

```
to make-a-tct
  set-default-shape tcts "truck"
  create-tcts 1 [
    set color red
    let tempxcor random (xmax - xmin)
    let tempycor random (ymax - ymin)
    setxy (tempxcor + xmin) (tempycor + ymin)
    set targeted 0
    set die-time ticks + dwell-time]
  set tcts-created tcts-created + 1
end
```

```
to red-search
  if any? hunters in-radius 7 [ask self[set target min-one-of hunters [distance myself]]]
```

```

end

to shoot-back
  ask tets with [target != nobody] [
    if distance target < 5 and engaged = 0 and [ycor] of target > ymin [
      hatch-stingers 1 [
        set target [target] of myself
        set color orange
        set parent myself
      ]
      set engaged 1
      ask target [set targeted 1]
    ]
  ]
end

```

```

to SAM-fly
  ask stingers [
    if target = nobody [die]
    ifelse parent = nobody
      [fd 1
        set counter counter + 1
        if counter >= 20 [
          ask target [set targeted 0]
          die]]
      [ifelse distance target <= 1
        [move-to target
          ask target [set TotalFuel TotalFuel + (FuelCapacity - fuel) die]
          set blue-killed blue-killed + 1
          if parent != nobody [ask parent [set engaged 0]]
          die]
        [face target
          fd 1
          set counter counter + 1
          if counter >= 20 [
            ask target [set targeted 0]
            die]
        ]
      ]
  ]
end

```

REFERENCES

- [1] Air Force Space Command, Space and Missile Systems Center, Directorate of Developmental Planning (SMC/XR). “What is SEAS?” <http://www.teamseas.com>. Last Accessed February 25th, 2009.
- [2] Alexandrov, N. M.; Lewis, R. M.; Gumbert, C. R.; Green, L. L. & Newman, P. “Optimization with Variable-Fidelity Models Applied to Wing Design”, *38th Aerospace Sciences Meeting & Exhibit*. 2000.
- [3] Alexopoulos, C. & Goldsman, D. “Verification and Validation”, ISyE6644 Class Lecture, Georgia Institute of Technology, Spring 2008.
- [4] Allen, P. & Wilson, B.M. Abrams, P. H. & Comfort, J., ed. *Modeling Qualitative Issues in Military Simulations with the RAND-ABEL Language*. 1988.
- [5] Amaral, L. & Ottino, J. “Complex networks: Augmenting the framework for the study of complex systems”, *The European Physical Journal B* Vol. 38, 147-162. 2004.
- [6] Andreou, A.S., Zombanakis, G.A., Georgopoulos, E.F., Likothanassis, S. (2000), “In Search of a Warning Strategy Against Exchange-rate Attacks: Forecasting Tactics Using Artificial Neural Networks”, *Discrete Dynamics in Nature and Society* Vol. 5, 121-137.
- [7] Anzalone, C. “40th NDIA Air Targets, UAVs Range Operations Symposium, Panama City, FL”, Technical report, Air Armament Center. 2002.
- [8] Bankston, B. & Key, T. “White Paper on Capabilities Based Planning.” 2006.
- [9] Benedicenti, L.; Succi, G.; Vernazza, T. & Valerio, A. “Object Oriented Process Modeling with Fuzzy Logic”, 267-271. 1998.
- [10] Berg, Paul D. “Doctrine and Technology”. *Air & Space Power Journal*. Spring 2008. <http://www.airpower.maxwell.af.mil/airchronicles/apj/apj08/spr08/focus.html>. Last accessed February 24th, 2009.
- [11] Biltgen, P. T. “A Methodology for Capability-Based Technology Evaluation for Systems-of-Systems”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2007

- [12] Bone, E. & Bolkcom, C. “Unmanned Aerial Vehicles: Background and Issues for Congress”, Technical report, Congressional Research Service. 2003.
- [13] Borer, N. K. “Decision Making Strategies for Probabilistic Aerospace Systems Design”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2006
- [14] Brock, G.; Ianacone, K. J. & Stringer, T. A. “Survivability Validation Protocol Issues for a System-of-Systems”, Technical report, Defense Nuclear Agency. 1993
- [15] Brownlee, L. & Schoomaker, P. J. “Serving a Nation at War: A Campaign Quality Army with Joint and Expeditionary Capabilities”, *Parameters* Summer, 4-23. 2004.
- [16] Bullock, R. K.; McIntyre, G. A. & Hill, R. R.J.A. Joines, K. K. & Fishwick, P., ed. *Using Agent-Based Modeling to Capture Airpower Strategic Effects*. 2000.
- [17] Cares, J. R. “Rule Sets for Sense and Respond Logistics: The Logic of Demand Networks”, Alidade Incorporated. 2004.
- [18] Cares, J. R.E. Yucesan, J. S. & Charnes, J., ed., *The Use of Agent-Based Models in Military Concept Development*. 2002.
- [19] Center, B. & Verma, B. “Fuzzy Logic for Biological and Agricultural Systems”, *Artificial Intelligence Review* 12, 213-225. 1998.
- [20] Chadwick, S. H. “Defense Acquisition: Overview, Issues, and Options for Congress”, Technical report, Congressional Research Service. 2007.
- [21] Chairman of the Joint Chiefs of Staff. “CJCSM 3170.01C Operation of the Joint Capabilities Integration and Development System”. 2007.
- [22] Chairman of the Joint Chiefs of Staff. “Joint Publication 3-01: Countering Air and Missile Threats”, Joint Chiefs of Staff. 2007.
- [23] Chairman of the Joint Chiefs of Staff. “CJCSM 3500.03A Joint Training Manual for the Armed Forces of the United States”. 2002.
- [24] Chae, H. G. “A Possibilistic Approach to Rotorcraft Design Through A Multi-Objective Evolutionary Algorithm”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2006.
- [25] Cioppa, T. M. & Lucas, T. W.R.G. Ingalls, J. S. & Peters, B., ed., *Military Applications of Agent-Based Simulations*. 2004.

- [26] Cook, G. P. "JCIDS: The New Language of Defense Planning, Programming and Acquisition", Technical report, Gregory P. Cook LLC. 2006
- [27] Cooper, R.; Bray, D. A. & Parzen, M. I. "Who Wins in a Dynamic World: Theory of Constraints Vs. Activity-Based Costing?", *in* North American Association for Computational Social and Organizational Science Conference. 2007
- [28] Davis, P. K. "Analytic Architecture for Capabilities-Based Planning, Mission-System Analysis, and Transformation", RAND Corporation.
- [29] Davis, P. K. "Effects-Based Operations: A Grand Challenge for the Analytical Community", RAND Corporation. 2001.
- [30] Davis, P. K.; Bankes, S. C. & Egner, M. "Enhancing Strategic Planning with Massive Scenario Generation", Technical report, RAND Corporation. 2007.
- [31] Davis, P. K. & Bigelow, J. H. "Motivated Metamodels: Synthesis of Cause-Effect Reasoning and Statistical Metamodeling", RAND Corporation. 2003.
- [32] Davis, P. K. & Henninger, A. "Analysis, Analysis Practices, and Implications for Modeling and Simulation", RAND Corporation. 2007.
- [33] Defense Science Board Task Force. "Advanced Modeling and Simulation For Analyzing Combat Concepts in the 21st Century." Washington: Office of the Under Secretary of Defense for Acquisition and Technology, 1999.
- [34] Department of Defense. "DoD Architecture Framework Version 1.5, Volume I: Definitions and Guidelines." April, 2007.
- [35] Department of the Navy. "Strategy." Washington: Headquarters United States Marine Corps, 1997.
- [36] Department of the Navy. "Tactics." Washington: Headquarters United States Marine Corps, 1997.
- [37] Druckenmiller, D.; Acar, W. & Troutt, M. "Agent-Based Modeling and Simulation of Strategic Scenarios with Repast 2.0". 2004.
- [38] Ehrhart, Robert C. "Some Thoughts on Air Force Doctrine." Air University Review March-April 1980.
<http://www.airpower.maxwell.af.mil/airchronicles/aureview/1980/mar-apr/ehrhart.html>. Last accessed on February 24th, 2009.
- [39] Ender, T. R. "A Top-Down, Hierarchical, System-of-Systems Approach to the Design of an Air Defense Weapon", PhD thesis, School of Aerospace

- Engineering, Georgia Institute of Technology. 2006.
- [40] Feuchter, C. A. “Air Force Analyst’s Handbook: On Understanding the Nature of Analysis”, Office of Aerospace Studies. 2000.
- [41] Fishwick, P. A. Osman Balei, R. E. N., ed. *Fuzzy Set Methods for Qualitative and Natural Language Oriented Simulation*. 1990.
- [42] Force Structure, Resources, and Assessments Directorate (J-8). “Capabilities-Based Assessment User’s Guide, Version 2”. 2006.
- [43] Frits, A. P. “Formulation of an Integrated Robust Design and Tactics Optimization Process for Undersea Weapon Systems”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2004.
- [44] Gerber, D. K. “Adaptive Command and Control of Theater Airpower”, Master’s thesis, School of Advanced Airpower Studies, Maxwell Air Force Base, Alabama. 1999.
- [45] Germanos, J. T. H. *Understanding TWV Capability Gaps in the Context of the JCIDS Process*. 2006.
- [46] Gile, R. H. “Global War Game Second Series 1984-1988”, Technical report, Naval War College. 1998.
- [47] Griffith, Paddy. “Battle Tactics of the Western Front: The British Army’s Art of Attack, 1916-18”. Great Britain: St Edmundsbury Press, 1994.
- [48] Hallion, Richard P. “Doctrine, Technology, and Air Warfare: A Late Twentieth – Century Perspective.” *Airpower Journal* Fall 1987. <http://www.airpower.maxwell.af.mil/airchronicles/apj/apj87/fal87/hallion.html>. Last Accessed on February 24th, 2009.
- [49] Hayes, D. R. E. & Iwanski, M. S. Ross-Witowski, C., ed. “*Analyzing Effects-Based Operations Workshop Report*.” 2002.
- [50] Henninger, A. E.; Gonzalez, A. J.; Georgipoulos, M. & DeMara, R. F. “Modeling Semi-Automated Forces with Neural Networks: Performance Improvement Through a Modular Approach”.
- [51] Henry, M. H. & Hogan, W. C. “Cost and Effectiveness Integration”, in Annual Department of Defense Cost Analysis Symposium. 1993
- [52] Hieb, M. & Kearley, J. “A Methodology for Doctrine in Modeling and Simulation: Battle Management Language (BML) and the Mission to Means Framework (MMF)” “2004 Fall Simulation Interoperability Workshop”. 2004.

- [53] Hill, J. M. D. “Anticipatory Planning with Agents Using Genetic Algorithms and Simulation”, PhD thesis, Texas AM University. 2001.
- [54] Hill, R. R.; Miller, J. & McIntyre, G. A.B.A. Peters, D. M. & Rohrer, M., ed. “*Applications of Discrete Event Simulation Modeling Military Problems.*” 2001.
- [55] Hilton, Henry J. “Successful Application to Weapon Acquisitions Requires Changes in DOD’s Environment”. U.S. General Accounting Office. 1998
- [56] House, Jonathan M. “Toward Combined Arms Warfare: A Survey of 20th – Century Tactics, Doctrine, and Organization.” Leavenworth, KS: U.S. Army Command and General Staff College, 1984.
- [57] http://www.deagel.com/Strike-and-Fighter-Aircraft/FA-22A-Raptor_a000531001.aspx last accessed February 16th, 2009
- [58] www.globalsecurity.org last accessed February 16th, 2009
- [59] <https://akss.dau.mil/ifc/> last accessed February 16th, 2009
- [60] <http://home.ubalt.edu/ntsbarsh/simulation/sim.htm> last accessed February 16th, 2009
- [61] http://cmm.cit.nih.gov/intro_simulation/node20.html last accessed February 16th, 2009
- [62] <http://www.cs.unc.edu/~lin/COMP290-72-S2K/> last accessed February 16th, 2009
- [63] http://books.nap.edu/openbook.php?record_id=9074&page=33 last accessed February 16th, 2009
- [64] http://www.msrr.army.mil/index.cfm?RID=MNS_A_1000185 last accessed February 16th, 2009
- [65] <http://rodin.wustl.edu/cosc/doc3.html> last accessed February 16th, 2009
- [66] <http://www.airpower.maxwell.af.mil/airchronicles/apj/apj04/sum04/vorsum04.html> last accessed February 16th, 2009
- [67] <http://ftp.fas.org/irp/program/collect/predator.htm> last accessed February 16th, 2009
- [68] <http://archives.cnn.com/2002/WORLD/meast/11/04/yemen.blast/index.html>

last accessed February 16th, 2009

- [69] http://www.navy.mil/search/display.asp?story_id=40532. USS New Hampshire Joins Fleet. Last Accessed February 20th, 2009
- [70] <http://www.m-w.com>. Merriam-Webster Online Dictionary. Last Accessed February 20th, 2009
- [71] <http://www.apc.au.af.mil/main.htm>. Air & Space Power Course. Last Accessed February 20th, 2009.
- [72] <http://www.dtic.mil/doctrine/jel/doddict/>. DoD Dictionary of Military and Associated Terms. Last Accessed February 20th, 2009
- [73] Johnson, R. O. & O'Neill, M. R. "Unmanned Aerial Vehicles in Perspective: Effects, Capabilities, and Technologies", Technical report, United States Air Force Scientific Advisory Board. 2003.
- [74] Joint Chiefs of Staff. "Joint Requirements Oversight Council (JROC) Programmatic Processes For Joint Experimentation And Joint Resource Change Recommendations". Chairman of the Joint Chiefs of Staff Instruction 3180.01. 2003.
- [75] Joint Chiefs of Staff. "Joint Publication 3-50.21: Joint Tactics, Techniques, and Procedures for Combat Search and Rescue". 1998
- [76] Joint Chiefs of Staff. "Requirements Generation System", Chairman of the Joint Chiefs of Staff Instruction CJCSI 3170.01B. 2001.
- [77] Joint Chiefs of Staff "Universal Joint Task List", Chairman of the Joint Chiefs of Staff Manual CJCSM 3500.04C. 2003.
- [78] Jones, J. R., "Development of Air Force Basic Doctrine 1947-1992", Air University Press. 1997.
- [79] Lovelace, Douglas C. Jr. "U.S. Army War College Key Strategic Issues List", Army Strategic Studies Institute. 2007.
- [80] Kelton, W., Sadowski, R., Sturrock, D. "Simulation with Arena." 4th Edition. McGraw-Hill. New York. 2007
- [81] Kewley, R. & Larimer, L. "An Agent-Based Modeling Approach to Quantifying the Value of Battlefield Information", *Phalanx* June, 10-13,25-26. 2003.
- [82] Landolt, J. P. & Evans, J. R. "R&D Initiatives in Modeling and Simulation for

- Capability Modernization of the Canadian Air Force”, *Canadian Military Journal* Spring, 37-42. 2001.
- [83] LaPlante, J. B.; Garner, D. P. & Hutzler, P. I. “Logistics in Wargaming - An Initial Report”, *Joint Force Quarterly* Winter, 97-101. 1996
- [84] Law, Averill. “Simulation Modeling & Analysis.” 4th Edition. McGraw-Hill. New York. 2007.
- [85] Lehmann, W. “Upgradeable Agent-Based Model to Explore Non-Linearity and Intangibles in Peacekeeping Operations”, Master’s thesis, Naval Postgraduate School. 2006.
- [86] Leonard, H. A.; Polich, J. M.; Peterson, J. D.; Sortor, R. E. & Moore, S. C. “Something Old, Something New: Army Leader Development in a Dynamics Environment”, RAND Corporation. 2006.
- [87] Lewe, J. “An Integrated Decision-Making Framework for Transportation Architectures: Application to Aviation Systems Design”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2005.
- [88] Macal, C.M., North, M.J. “Tutorial on Agent-Based Modeling and Simulation.” Proceedings of the 2005 Winter Simulation Conference. 2005.
- [89] Martin, P.P.A. Farrington, D. S. & Evans, G., ed. *The Modeling of Tactics and Procedures Using A Component Based System*. 1999.
- [90] Mattingly, J., D., Heiser, W., H., Daley, G., H., *Aircraft Engine Design*, AIAA, Washington, 1987.
- [91] Matsumura, John, et al. “Analytic Support to the Defense Science Board: Tactics and Technology for 21st Century Military Superiority.” Santa Monica, CA: RAND Corp, 1997.
- [92] Mazzei, Francesco. “Air Doctrine, echoes from abroad.” *Air University Review*. July-August 1981.
<http://www.airpower.maxwell.af.mil/airchronicles/aureview/1981/jul-aug/mazzei.htm>. Last Accessed February 20th, 2009.
- [93] McAllister, M. L.; Ovchinnikov, S.; Dockery, J. & Adlassnig, K.D. Gantz, S. S., ed. , *Tutorial on Fuzzy Logic in Simulation*. 1985.
- [94] Michael Bowling, M. V. “An Analysis of Stochastic Game Theory for Multiagent Reinforcement Learning”. 2000.
- [95] Min, F.; Ma, P. & Yang, M.S.G. Henderson, M. H. J. S. J. T. & Barton, R., ed.

A Knowledge-Based Method for the Validation of Military Simulations. 2007.

- [96] Moffat, J.; Smith, J. & Witty, S. “Emergent Behaviour: Theory and Experimentation Using the MANA Model”, *Journal of Applied Mathematics and Decision Sciences* **May**, 1-13. 2006.
- [97] Mulgund, S.; Harper, K.; Krishnakumar, K. & Zacharias, G. *Air Combat Tactics Optimization Using Stochastic Genetic Algorithms*. 1998.
- [98] National Defense Industrial Association “Game Report NDIA/USMC War Game 2005 Distributed Operations”. 2005.
- [99] Office of Aerospace Studies. “Functional Solution Analysis Handbook: A Guide for Implementing Functional Solution Analyses”. 2005.
- [100] Office of the Secretary of Defense. “Unmanned Aircraft Systems Roadmap 2005-2030”, Technical report, United States Department of Defense. 2005.
- [101] Office of the Under Secretary of Defense for Acquisition, Technology & Logistics “Strategic Goals Implementation Plan”, Technical report, Under Secretary of Defense for Acquisition, Technology, and Logistics. 2007
- [102] Office of the Under Secretary of Defense for Acquisition, Technology & Logistics “Department of Defense Acquisition Modeling and Simulation Master Plan”. 2006
- [103] Office of the Under Secretary of Defense for Acquisition, Technology & Logistics. “System of Systems Systems Engineering Guide: Considerations for Systems Engineering in a System of Systems Environment”, Technical report. 2006.
- [104] Office of the Under Secretary of Defense for Acquisition, Technology & Logistics “Defense Science Board Task Force on Modeling and Simulation for Analyzing Combat Concepts in the 21st Century”. 1999.
- [105] OR Concepts Applied. “What Is OPUS?” <http://www.orconceptsapplied.com>. Last Accessed February 25th, 2009.
- [106] Papadimitriou, G, Sadoun, B, Papazoglou, C. “Fundamentals of System Simulation.” In *Applied System Simulation: Methodologies and Applications* by Obaidat, M, Papadimitriou, G (Eds.)
- [107] Patricio, M.; Carbo, J.; O. Perez, J. G. & Molina, J. “Multi-Agent Framework in Visual Sensor Networks”, *EURASIP Journal on Advances in Signal Processing*. 2007.

- [108] Pawloski, J. S. “Modeling Tactical Level Combat Using A Multi-Agent System Design Paradigm (GI Agent)”, Master’s thesis, Naval Postgraduate School. 2001.
- [109] Pfaender, J. H. “Competitive Assessment of Aerospace Systems using System Dynamics”, PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2006.
- [110] Przemieniecki, J.S. “Mathematical Methods in Defense Analyses, 3rd ed.” Reston, VA: American Institute of Aeronautics and Astronautics Inc, 2000.
- [111] Raymer, D., *Aircraft Design: A Conceptual Approach*, 3rd Ed. American Institute of Aeronautics and Astronautics, Reston, 1999.
- [112] Rickerman, L. D. “Effects-Based Operations: A New Way of Thinking and Fighting”, School of Advanced Military Studies, United States Army Command and General Staff College. 2003.
- [113] Robinson, S.L.F. Perrone, J. L. B. L. D. N. & Fujimoto, R., ed. “*Conceptual Modeling for Simulation: Issues and Research Requirements.*” 2006.
- [114] Rumsfeld, D. H. “Quadrennial Defense Review Report”, United States Department of Defense. 2001.
- [115] Rumsfeld, Donald. “Quadrennial Defense Review Report”. Department of Defense. 2006
- [116] Schieman, K. G. & Passaro, J. R. “Cost Estimating Cases: Educational Tools for Cost Analysts”, Master’s thesis, Air Force Institute of Technology. 1993
- [117] Schinasi, K. V. “GAO-01-288 Better Matching of Needs and Resources Will Lead to Better Weapon System Outcomes”, United States Government Accountability Office. 2001.
- [118] Shaw, Robert L. “Fighter Combat: Tactics and Maneuvering.” Naval Institute Press, Annapolis, Md. 1985.
- [119] Sheehan, J. H.; Deitz, P. H.; Bray, B. E.; Harris, B. A. & Wong, A. B. “The Military Missions and Means Framework”. 2004.
- [120] Simpkins, S. D.; Paulo, E. P. & Whitaker, L. R. Peters, B.; Smith, J.; Medeiros, D. & Rohrer, M., ed. “*Case Study in Modeling and Simulation Validation Methodology.*” 2001.
- [121] Smith, R. “Unmanned Aircraft Systems (UAS) Joint Analysis Team (JAT)”, Technical report, Office of Deputy Under Secretary Defense Science

Technology/Weapon Systems. 2007.

- [122] Soban, D. S. "A Methodology for the Probabilistic Assessment of System Effectiveness as Applied to Aircraft Survivability and Susceptibility", PhD thesis, School of Aerospace Engineering, Georgia Institute of Technology. 2001.
- [123] St. Laurent, Janet. "Force Structure: Capabilities and Cost of Army Modular Force Remain Uncertain", GAO-06-548T. U.S. Government Accountability Office. 2006.
- [124] Steeb, Randall, et al. "Examining the Army's Future Warrior: Force-on-Force Simulation of Candidate Technologies." Santa Monica, CA: RAND Corp, 2004.
- [125] Steele, W. M. & Jr., R. P. W. "Training and Developing Army Leaders", *Military Review*, 2-9. 2001.
- [126] Tanenbaum, P. J.; Bray, B. E. & Sheehan, J. H. "Mission and Means Framework Application: Presentation to 73rd MORSS U.S. Military Academy". 2005.
- [127] Ternion Corporation. "What Is FLAMES?" <http://www.ternion.com>. Last Accessed February 25th, 2009.
- [128] The MathWorks, Inc. "MATLAB Overview." <http://www.mathworks.com/products/matlab/>. Last Accessed February 25th 2009.
- [129] Tirpak, J. A. "UAVs With Bite", *AIR FORCE Magazine* January, 46-50. 2007.
- [130] Titus, N. "Air Force CONOPS & Capabilities Based Planning", Air Force Studies Analyses Agency. 2004.
- [131] Ullah, A. S. "A Fuzzy Decision Model for Conceptual Design", *Systems Engineering* 8, 296-308. 2005.
- [132] United States Air Force. "Air Force Doctrine Document 2-1.6: Personnel Recovery Operations". 2005.
- [133] United States Air Force. "Air Force Task List" Air Force Doctrine Document 1-1. Secretary of the Air Force. 1998.
- [134] United States Air Force. "Air Force Instruction 38-101: Air Force Organization". 2006.

- [135] United States Air Force. "Air Force Policy Directive 10-6: Capabilities-Based Planning and Requirements Development". 2006.
- [136] United States Air Force. "Air Force Instruction 10-604: Capabilities-Based Planning". 2006.
- [137] United States Air Force. "Air Force Instruction 10-601: Capabilities-Based Requirements Development", Technical report, United States Air Force. 2006
- [138] United States Air Force. "Air Force Doctrine Document 2-1.7: Airspace Control In The Combat Zone", Secretary of the Air Force. 2005.
- [139] United States Air Force. "The U.S. Air Force Remotely Piloted Aircraft and Unmanned Aerial Vehicle Strategic Vision", 2005.
- [140] United States Air Force. "Air Force Policy Directive 63-1: Capability-Based Acquisition System". 2003.
- [141] United States Air Force. "The U.S. Air Force Transformation Flight Plan", Technical report, HQ USAF/XPXC Future Concepts and Transformation Division. 2003
- [142] United States Air Force. "Air Force Doctrine Document 1: Air Force Basic Doctrine", Secretary of the Air Force. 2003.
- [143] United States Air Force. "Air Force Instruction 10-2305: Wargaming", Secretary of the Air Force. 2003.
- [144] United States Air Force. "Air Force Doctrine Document 2-1.1: Counterair Operations". 2002.
- [145] United States Air Force. "Air Force Doctrine Document 2-1: Air Warfare". 2000.
- [146] United States Air Force. "Air Force Instruction 10-1302", Secretary of the Air Force. 1998
- [147] United States Air Force. "Operational Test and Evaluation", Technical report, Air Force Instruction 99-102. 1998.
- [148] United States Air Force. MQ-1 Predator Factsheet. <http://www.af.mil/factsheets/factsheet.asp?fsID=122>. July 14, 2008.
- [149] United States Air Force. MQ-9 Reaper Factsheet. <http://www.af.mil/factsheets/factsheet.asp?id=6405>. July 14, 2008

- [150] United States Army Department of Command, L. & Management “*How the Army Runs: A Senior Leader Reference Handbook*,” U.S. Army War College. 2008.
- [151] United States Department of Defense. “Defense Acquisition Guidebook” <https://akss.dau.mil/dag/welcome.asp>. July 14, 2008
- [152] United States Department of Energy. “Department of Energy Directive Guide: Cost Estimating, Analysis, and Standardization”, DOE G 430.1-1. 1997.
- [153] U.S. Army Cost and Economic Analysis Center. “Department of the Army Cost Analysis Manual”, Department of the Army. 2002.
- [154] U.S. Army War College. “Army Transformation Wargame 2001”. 2001.
- [155] U.S. Government Accountability Office. “Cost Assessment Guide: Best Practices for Estimating and Managing Program Costs”, GAO-07-1134SP. 2007.
- [156] U.S. War Department. “War Department Field Manual FM 100-20: Command and Employment of Air Power”. 1943.
- [157] Watts, Barry D. “Doctrine, Technology, and War.” Air & Space Doctrinal Symposium Maxwell AFB, Montgomery, Al 1996.
<http://www.airpower.maxwell.af.mil/airchronicles/cc/watts.html>. Last Accessed February 24th, 2009.
- [158] Wijesekera, D.; Michael, J. & Nerode, A. “*An Agent-Based Framework for Assessing Missile Defense Doctrine and Policy*.” 2005.
- [159] Wilensky, U. NetLogo. <http://ccl.northwestern.edu/netlogo/>. Center for Connected Learning and Computer-Based Modeling, Northwestern University. Evanston, IL. 1999.
- [160] Willers, C. J. “Modeling Military Information Operations with Multi-Agent Complex Adaptive System Techniques”.
- [161] Woodaman, R. F. “Agent-Based Simulation of Military Operations Other Than War Small Unit Combat”, Master’s thesis, Naval Postgraduate School. 2000.
- [162] Wu, C.F.J., Hamada, M. “Experiments: Planning, Analysis, and Parameter Design Optimization.” John Wiley and Sons, INC. New York. 2000.
- [163] Yarger, H. R. “*Strategic Theory for the 21st Century: The Little Book on Big Strategy*,” Army Strategic Studies Institute. 2006.

- [164] Yoner, N. "Major Weapon Systems Acquisition and Life Cycle Cost Estimation: A Case Study", Master's thesis, Naval Postgraduate School. 2001.
- [165] Zeigler, Bernard. "A Framework for Modeling and Simulation."