3-20-2002

A Large Scale Integer Linear Program as a Decision Support Tool for Force Mix Selection

Craig A. Punches

A LARGE SCALE INTEGER LINEAR PROGRAM AS A

DECISION SUPPORT TOOL FOR FORCE MIX SELECTION

THESIS

Craig A. Punches, Captain, USAF

AFIT/GLM/ENS/02-15

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright Patterson Air Force Base, Ohio

AFIT/GLM/ENS/02-15

A LARGE SCALE INTEGER LINEAR PROGRAM AS A DECISION SUPPORT TOOL FOR FORCE MIX SELECTION

THESIS

Presented to the Faculty

Department of Operational Sciences

Graduate School of Engineering and Management

Air Force Institute of Technology

Air University

Air Education and Training Command

In Partial Fulfillment of the Requirements for the

Degree of Master of Science in Logistics Management

Craig A. Punches, B.S.

Captain, USAF

March 2002

APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED

AFIT/GLM/ENS/02-15

A LARGE SCALE INTEGER LINEAR PROGRAM AS A

DECISION SUPPORT TOOL FOR FORCE MIX SELECTION

Craig A. Punches, B.S.

Captain, USAF

Approved:

_/s/______________________________________	7 Mar 02
Stephen M. Swartz, Maj, USAF, Advisor Assistant Professor of Logistics Management Department of Operational Sciences	Date
_/s/______________________________________	7 Mar 02
William P. Nanry, COL, USA, Reader Assistant Professor of Operations Research Department of Operational Sciences	Date

Acknowledgements

While I am the author of this thesis, many other people have made significant contributions to its very existence.

First and foremost, I dedicate this to my wife, daughters, and son for knowing that my job is not an easy one, and doing everything they can to make it easier. A special thanks to my mom for teaching me determination in the face of adversity, to my dad for teaching me to dream and think "big picture," to grandma T who nurtured me and taught me to believe in myself, to grandpa T who taught me about life and the world around us, and to Mr. Smith, my 5th grade teacher and WWII veteran, who taught me war is not glamorous but requires patriots to protect our freedoms, racism is un-American, and that I can accomplish anything that I set my mind to. My in-laws deserve special thanks for treating me like a son and supporting our family in every way.

I'm also very grateful to my thesis committee for their help during this research:

- My advisor, for letting me discover what Socratic learning is.

- My reader, for providing valuable insight.

Craig Punches

Table of Contents

List of Figures

List of Tables

AFIT/GLM/ENS/02-15

Abstract

In the post cold war environment, the rapid deployment of combat capability is critical. Deployment lift capability is limited, however, so the real-time selection of the optimal combat asset mix that balances capability provided and sustainment required has become paramount. In this model, the value of a force mix is determined by the sum of the individual weapon system "suitabilities" against their assigned missions. The value is constrained by the numerical limits on the items required to create and support the force mix, and the lift required to move these items.

The research considered heuristic and complete enumeration methods against the problem structure to develop a decision support model that expedites the selection of the best overall force mix. War planners are provided a decision support tool that objectively compares alternative force mix packages and selects the optimal asset mix in a reasonable amount of time while explicitly considering logistics constraints. This demonstrates the feasibility of an approach that integrates intelligence, operations, and logistics issues into a single decision support and planning tool for force mix decisions.

A LARGE SCALE INTEGER LINEAR PROGRAM AS A DECISION SUPPORT TOOL FOR FORCE MIX SELECTION

I. Introduction

Background

In a perfect world, force mix optimization determinations (what and how many assets to take to meet mission requirements) and time phased force deployments (when) would not be necessary since the on-scene commanders would have everything instantaneously--the entire Air Force, Army, Marine Corps, and Navy fleet would be instantaneously in place.

However, the real world is not so generous, since this instantaneous availability of all desired resources would require unlimited and unconstrained transportation. This does not exist in contingency actions or war. Transportation planners would require a long, foreseeable planning window and/or a long build-up time to have effectively unlimited and unconstrained transportation. For example, the military would need to know that in 20 years they would be required to deploy “X” pieces of “Y” equipment to location “Z” or know that they face the perfect enemy, like Saddam Hussein, who would remain foolhardy and nonaggressive while we build up our forces.

These situations will not likely occur in future conflicts, causing transportation to be a very restrictive, binding constraint. The reduction of the United States (US) forward presence in overseas locations magnifies this and other logistics constraints. The

Aerospace Expeditionary Force (AEF) concept reflects the Air Force's Global Engagement vision of worldwide deployment from continental US and attempts to simultaneously pursue initiatives to reduce the resources required for deployment, the deployment *footprint*, and to increase management flexibility of resource allocation (Filcek, 2001).

If it were up to operations, they would want to take everything all at once. However, since this is impossible, a prioritized time sequence list must be developed reflecting what assets and logistics to take and when to transport it. This has traditionally taken the form of the Unit Type Code (UTC) based Time Phased Force Deployment Data (TPFDD) document. Although the Gulf War deployment has generally been considered a success, the mentality of trying to take everything with limited transportation and logistics support highlighted several negative results. According to the US Secretary of Defense at the time, "The readiness rates and operating tempos of primary platforms such as aircraft, tanks and fighting vehicles outpaced the ability of support structures and equipment. For instance, aerial tankers became a limiting factor in air operations" (Aspin, 1992: xiii.3). This observation was supported by recent research: "One of USTRANSCOM's most intractable and high-visibility problems during Desert Shield/Desert Storm was a backlog of sustainment cargo at aerial ports of embarkation" (Wakefield, 2001: 1). The amount of cargo arriving at Al Jubayl and Ad Dammam overwhelmed the ability of the system to manage the traffic. General Gus Pagonis, chief logistician of the Gulf War, noted, "We had to open some 28,000 of 41,000 arriving containers right there on the dock just to find out what was in them" (Pagonis, 1992: 206).

The traditional UTC based TPFDD document is a generic, large-scale plan that bases force mix on large, aggregate chunks of capability (forces) combined with the "close date" concept (Swartz, 2001). This aggregation over forces and time results in sub-optimization of the logistics system.

Forces aggregation is the grouping together of assets based on capability that fulfill general mission roles. Since this resource aggregation is for general purpose planning, mobilizing in pieces or for specific circumstances requires improvisation rather than systematic evaluation of the true effects on marginal logistics requirements and combat capability. For example, a theater commander in charge (CINC) may want three precision bombing sorties daily in desert conditions, but the traditional UTC is designed for bombers in nonspecific areas performing nonspecific bombing missions with preselected two or four-ship aircraft packages. How does the planner adjust the package to limit the logistics drain on the transportation system while ensuring the CINC gets what he needs? The planner could choose the 4-ship package and reduce the number of aircraft by one, but what cargo and personnel can be eliminated as redundant or unnecessary? The traditional UTC provides no means of conducting a marginal analysis of the trade-offs between the marginal cost of logistics and the marginal cost of combat capability.

Time window aggregation, "close date" concept, accentuates this large chunk delivery by having large scale combat capability delivered by some future date. This concept is illustrated by deploying all the bullets needed for a UTC on one shipment, followed by all the weapons on a separate shipment, then followed by all the combat troops on a final shipment. Each piece of material arrives by some future date, but the package as a whole is ineffective until that time. Another example would be sending aircraft before sending maintenance and fuel troops. According to former Joint Staff Director of Operations Lieutenant General Thomas Kelly, "At some point [during the Gulf War], a fuel truck became more important than the tank it supported because it is no good to have the tank if you did not have the fuel for it" (Aspin, 1992: 36). The UTC resource and time aggregation creates large batch delivery of combat capability into theater and results in combat capability only being available in a step-wise function (see Figure 1).

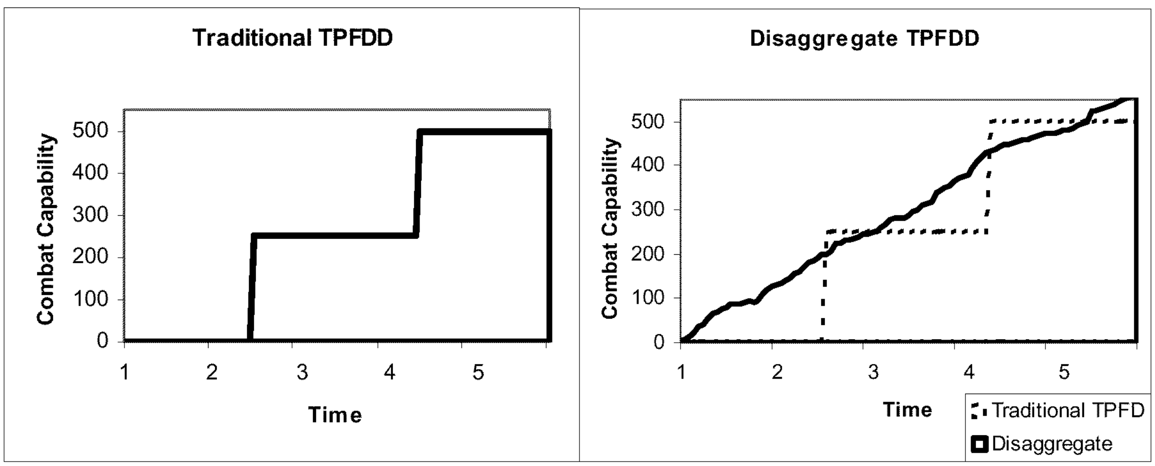


Figure 1. Traditional and Disaggregate TPFDD Combat Capability Comparison

This large batch delivery makes the TPFDD difficult to generate, non-responsive, and not adequate for dynamic/crisis planning (Carrico, 2000). The Air Force War and Mobilization Plans Division identified this shortfall by stating, "During the initial construction of 10 AEFs and 2 AEWs, functional area managers (FAMs) noted UTCs seemed too large for SSC [Small Scale Contingency] requirements…Kosovo highlighted need for more 'right-sized' UTCs" (Holleran, 2000). From Ms Holleran's briefing, the desirability of smaller "building blocks" of capability was identified during the 2000

World Wide Planner's Conference by the Air Force War and Mobilization Plans Division.

Instead of giving CINCs a preselected/prepackaged list of menu items (aggregate TPFDD) delivered by some future date, a dynamic planning tool would allow the CINCs the ability to tailor forces and timing. The realization that the traditional approach of aggregate TPFDDs used during the predictable cold war era would not effectively resolve the unpredictable challenges of the post-cold war environment (rogue nations with weapons of mass destruction, terrorism, ethnic tension, etc.) led to the creation of the Advanced Logistics Program (ALP).

ALP, a joint Defense Advanced Research Projects Agency (DARPA) and Defense Logistics Agency (DLA) research project, was developed to speed up the sourcing and tailoring of existing TPFDDs. ALP intended to create a multi-echelon, real time tool to rapidly provide operators and logisticians rational alternatives that balance mission attainment and logistic footprint while remedying the UTC based TPFDD shortfalls.

In appreciating the power of ALP, researchers at the Air Force Institute of Technology (AFIT) proposed that TPFDD disaggregation was now possible, practical, and highly desirable. Smaller, discrete pieces of capability can be delivered in incremental portions and time according to the CINCs' exact desires. It would be possible to create single resource UTCs for specific missions with one combatant and the necessary combat support. This eliminates excess items, creating more transportation space for items that are really needed and increasing delivery speed in turn.

Researchers, realizing that the TPFDD could be disaggregated (smaller UTCs and shorter time blocks), have been trying to take advantage of this smaller, discrete information system. AFIT is pursuing the development of a Mission-Resource Value Assessment Tool (M-R VAT) that will maximize utility or value delivered into a theater over time. This tool is designed to rapidly identify alternative force mixes by matching mission preferences to tasks, tasks to resources, and resources to logistics requirements. This should allow combat capability to be delivered sooner and more consistently, improving capability/defensibility and the projection of near perfect economy of force (see Figure 1). Going back to the example of Desert Storm, US forces were not capable of preventing Iraqi infiltration until mid-September, one month after deployment (Pagonis, 1992). This might have been accelerated using the disaggregate UTC approach.

The M-R VAT uses the concept of "Mission Ready Resources" (MRRs) to accomplish this matching. A Mission Ready Resource is the basic unit of these small, discrete blocks of capability. An MRR is composed of a resource type and its logistics requirements, i.e. aircraft, pilot, fuel, munitions, support equipment and personnel, etc., that has a certain "suitability" for a specific task. It is envisioned that the underlying M-R VAT logic can be applied to a wide range of similar problems. This would help the Army determine the best mix of tanks, artillery, aircraft, and soldiers; the Navy determine best mix of ships; the Marines determine the best mix of vehicles and personnel; and the Air Force determine the best mix of aircraft.

Problem Statement

Since we can now plan and execute for much smaller units of capability to be brought into theater in much smaller time increments, what is the best sequence of deliveries? Specifically, how would we optimize the delivery of capability subject to constrained resources?

Research Question

Is there a method that can rapidly find solutions to incremental force package mixes that balances mission supportability and capability?

Investigative Questions

To successfully create a quantitative decision support tool to enhance the force mix selection efforts of campaign planners, this paper addresses the following questions:

1. What is the underlying nature and structure of the problem being studied?
    a. What are we trying to maximize?
    b. What are our constraints?
2. What are the solution methodologies that best fit this problem structure?
    a. What are the key characteristics (to solution type) of the problem?
    b. What are the matching solution types?
3. Can the effectiveness and efficiency of the selected methodology be tested?
4. What are the results of this test (runtime, quality of solution, do you get an answer, does it converge on known optima)?
5. What test would be performed and what inferences could/should be drawn from the test results?

Answering these questions will ensure the decision support model is valid for determining the best force mix that balances capability and supportability.

Research Methodology

This research involved four phases designed to answer the five investigative questions. The first phase included a comprehensive literature review to determine what data was required, what analysis would be performed, and how the results will be interpreted. The data required included MRR type, MRR task suitability values, MRR lift consumption values, resource availability quantity, deployment lift availability quantity, resource daily mission turn rates, and CINC mission/task integer preference values.

The second phase included data collection. The MRR suitability and deployment lift consumption were based on notional values from the ALP Pilot problem since actual mission specific resource capabilities were classified. To ensure this research receives the widest degree of evaluation for validity, this research and its conclusion were restricted to the unclassified realm. Actual logistical support for a given resource is difficult to determine, is often undefined until just prior to movement, and has not previously been evaluated via the MRR concept (Judge, 1998:32). This research was forced to make simplifying assumptions about the relationship between MRR missions and lift/logistics consumption.

The third phase involved the creation of the decision support model and exploration of how to make the module fit within deployment material flow planning.

The fourth phase evaluated the methodologies against solution quality and runtimes. Solution quality and runtime are discussed in Chapter III and Chapter IV. This paper reevaluated David Wakefield's work using other approaches to develop an algorithm that converges on an acceptable solution in a reasonable amount of time.

Assumptions

Since the actual mission specific asset capabilities are classified, this research will be constructed around a notional AEF of a size and diversity similar to that depicted in the ALP Pilot Program (Swartz, 1999). Actual resource suitability and lift costs will not be used. Once the results demonstrate the validity of the model, the actual classified values can be substituted into the model for full implementation. To support this research, this paper assumes that lift consumption is accurate and available for planning. Past research has suggested that for standard UTC sized F-16 deployments, the relationship of resource quantity to consumption is linear (Goddard, 2001).

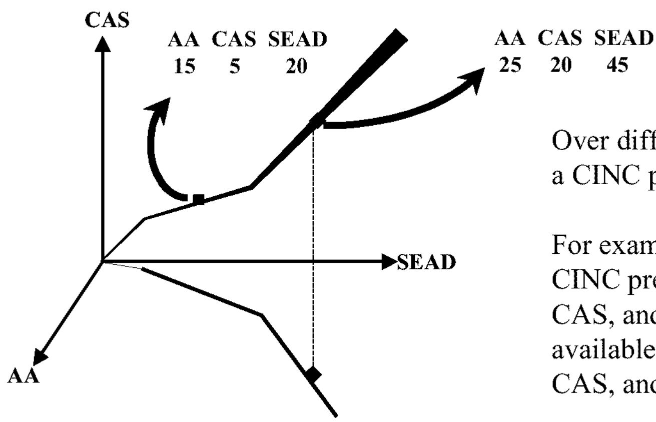
This decision model's validity is based on the continuance of the USAF's Global Engagement and Global Reach vision. That is, the USAF responds to theater crises by primarily deploying combat troops from CONUS stations. A dramatic change in this operating concept could be represented by a return to the cold war military era of forward basing, in which nearly all forces are based in the theaters of crisis. Such a change could lessen or eliminate the positive impact of this decision support tool. However, the assumption of continuing Global Engagement and Global Reach is sensible, given the end of the cold war, the end of large defense budgets, and the inefficiency of forward basing.

It is also assumed that the MRR task suitability values and numbers available are stable and constant. This appears reasonable since a change in resource capabilities and quantity generally takes years to fund and implement. In addition, changes to resource suitability could be easily programmed into the model.

The asset type turn rates will be assumed to be constant and stable over time and all mission types. If an F-15 aircraft can do three missions a day, on average, the turn rate will be three regardless of the type of mission. This value is used for the calculation of the asset availability.

This research and M-R VAT were only concerned with the deployment planning and build-up phase while deployed resources were expected to be redeployed to their origins or elsewhere at the end of the conflict. Therefore, it was assumed that the quantity of missions available would not decrease. Resources would build-up over time, and the CINC would desire more missions (i.e. sorties) over time. As the level of total resources increased, the CINC would prefer a "mix" of MRRs for different missions. This changing mix preference would describe a curve or vector that can be referred to as the "mission preference curve."

Since the ultimate goal of a logistics planner is to satisfy the CINCs requirements, it is assumed to be highly desirable to move along the mission preference curve. The decision maker's task preference between diverse missions, such as Air to Air (AA), Suppression of Enemy Air Defenses (SEAD), or Close Air Support (CAS), can be explicitly indicated by the number of missions, i.e. sorties, desired for each task at a given resource level (missions available). This is illustrated in Figure 2. For instance, given a relatively low amount of missions available (sortie generation rate) at the beginning of a campaign, a CINC may prefer a ratio of 55 percent AA, 40 percent SEAD, and only 5 percent CAS to achieve air superiority. Over time, the number of available missions increases and the next campaign phase may emphasize ground attack. This is reflected in the CINC's task preferences: 10 percent AA, 30 percent SEAD, and 60 percent CAS.



Over differing levels of total resources available, a CINC prefers a different "mix" of missions.

For example, when 40 sorties are available, the CINC preference curve would include 15 AA, 5 CAS, and 20 SEAD missions. If 90 sorties were available, the CINC would prefer 25 AA, 20 CAS, and 45 SEAD.

Figure 2. Mission Preference Vector (Curve) (Johnson and Swartz, 2000:27)

If a commander decided that fewer missions were required or that a different ratio or values of task types applied, then a new campaign phase would begin and the planning process described here would be re-accomplished from scratch.

Since Air Force deployment cargo is generally denser (heavy) than it is bulky causing deployment lift to be constrained by deployment weight rather than volume, it is assumed that minimizing volume is unnecessary. If volume does becomes an issue, it can be easily managed as an additional constraint, just like weight, by setting the sum of the MRRs multiplied by their volume equal to or less than the available deployment volume.

It is assumed that the availability of deployment lift can be determined, thus deployment weight would be a hard constraint making weight minimization unnecessary.

Scope/Limitations

This research makes use of the composite (asset's designed capability to accomplish a specific mission or task and situational considerations such as host nation runway capability, political considerations, or fuel support) suitability of MRRs to examine the tradeoff between MRR suitability against MRR lift cost. Since situational constraints placed on the US by host nations or at the staging base may prevent the US from selecting certain combat assets or the combat assets that best meet the situational constraints may have little to no value for required tasks, selecting only part of the composite suitability would provide an incomplete solution and possibly an infeasible solution. To accurately represent reality, the composite suitability is required to balance both design suitability and situational considerations. This process is envisioned to create the composite suitability value (Best Value Asset Set Per Phase) by incorporating the Value Focused Thinking (campaign specific goodness) Decision Support Tool developed by Chistopher Buzo and Paul Filcek with the asset design capabilities; however, this process and evaluation is beyond the scope of this research. This merging of design capabilities and situational considerations to create the composite suitability is illustrated in Figure 3. Each specific campaign issue is compiled to create the Campaign Specific Value, which is combined with the Generic Mission Goodness (Design Suitability) to create the Best Value Asset Set Per Phase (Composite Suitability). The composite value is discussed in greater detail in Chapter II.

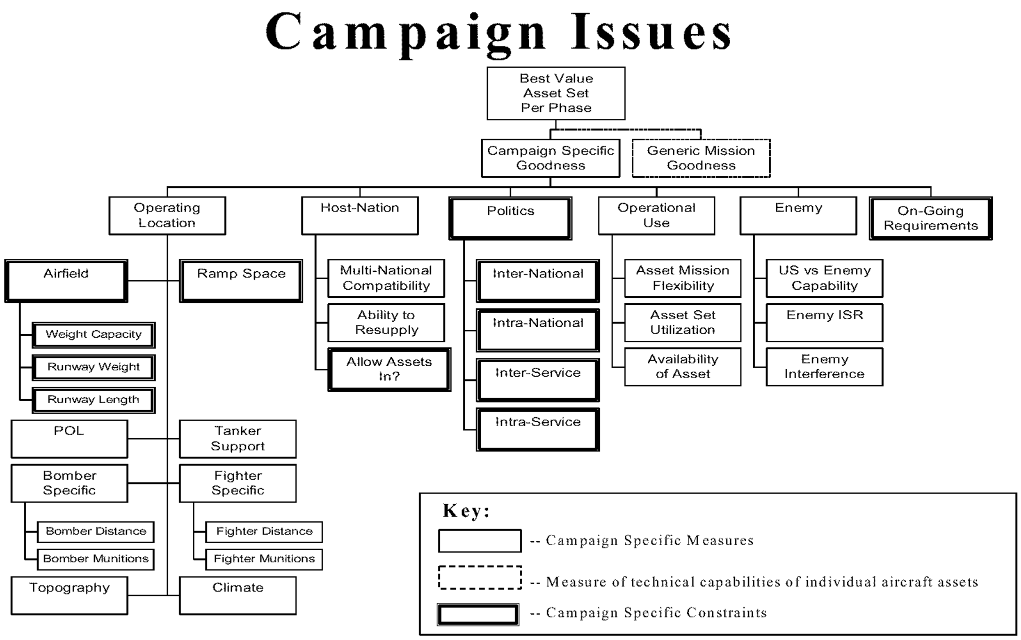


Figure 3. Campaign Issues Value Hierarchy (Buzo, 2000:80)

Discussion of operational plans and actual task suitability may be classified. Therefore, this research, research conclusions, and the information used are unclassified. All data will be reasonable but notional. The purpose of the research is to develop and validate a structure and an approach, not discern a specific answer to a specific force mix and flow problem.

Summary

This chapter provides the justification and motivation for developing a decision support tool that will balance the desires of operations and logistics planners to aid selection of the best force mixes for contingencies. With a continuing resource constrained environment, campaign and deployment planning is critical. This research proposes to develop a methodology for presenting campaign planners a best force package mix balancing combat capability and supportability.

Chapter II provides a background on the characteristics of the problem and reviews classic and modern mathematical programming techniques. Chapter III describes the methodology used to construct the large-scale integer linear programming problem and evaluate the solution approach. Chapter IV details the results using the selected integer linear programming optimization methodology. Chapter V provides conclusions on research contributions and makes recommendations for further research.

II. Literature Review

Introduction

The overall purpose of this chapter is to answer the second investigative question: what are the solution methodologies that best fit this problem? To answer this investigative question, it will be broken down into its two subcomponents: problem structure and solution approaches. First, the key characteristics of the problem were determined. After the problem structure was determined, matching solution types were reviewed.

This chapter first presents the problem structure by reviewing changes in post cold war concepts of operations (CONOPS); the need for joint service, combined service, and merging operations and logistics planning and decision support tools; and previously used methodologies. The second section of this chapter provides an overview of the various solution approaches. The characteristics, requirements, and benefits of these approaches will be reviewed. The chapter concludes by discussing why large-scale integer linear programming was the selected approach.

Problem Structure

Cold War CONOPS Change.

Faced with the threat of a known adversary, the Cold War (although a time of great danger) had a predictable, certain, and stable environment where rivals typically used conventional, symmetric means of attack and planning. Since there was a continuous threat from a known opponent, the Department of Defense could plan for and acquire resources necessary for operational requirements. After the Gulf War, "a prominent theory arose that there would no longer be a need for large land forces, that power projection and national strategy could primarily be carried out through precision strikes using technologically advanced air and naval forces" (AV2010, 1996). This theory was termed the "standoff" approach. The approach was expected to eliminate the need for large land forces, since the degree of US involvement and commitment would be dramatically reduced. "Reality proved that theory to be invalid" (AV 2010, 1996). This is illustrated in Figure 4 and Figure 5. The Army conducted 10 deployments from 1950 to 1989 and 25 from 1990 to 1996, an increase in missions by a factor of 16 (AV 2010, 1996).

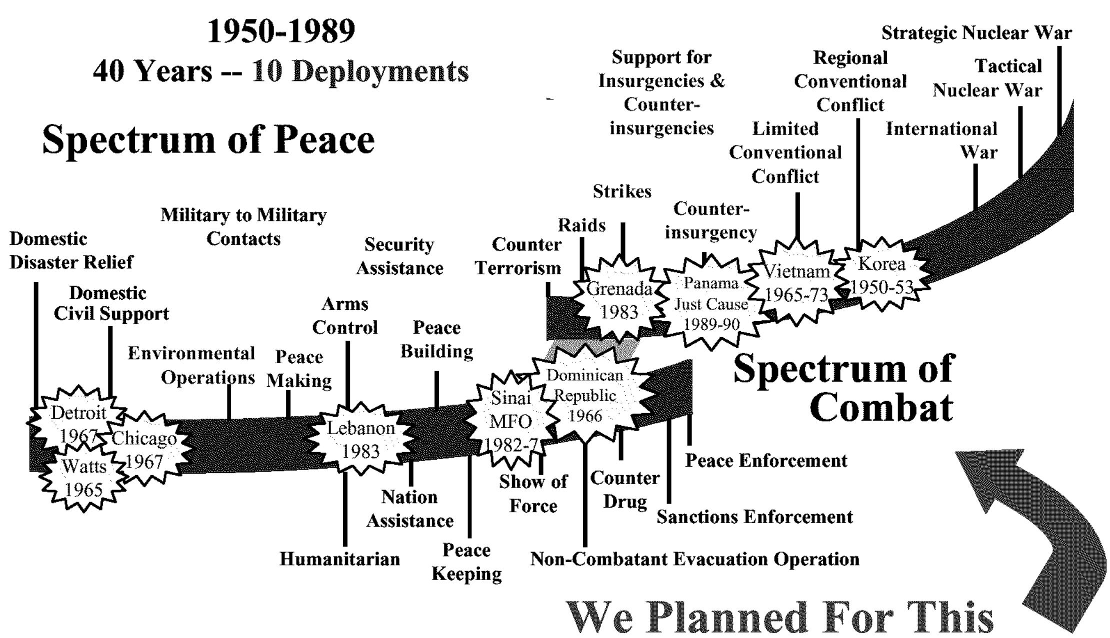


Figure 4. 1950-1989 Army Deployments (AV 2010, 1996)

Today, as suggested by Figure 5, threats to national security are discontinuous, rapidly changing, dynamic, unconventional, and unpredictable with adversaries who will

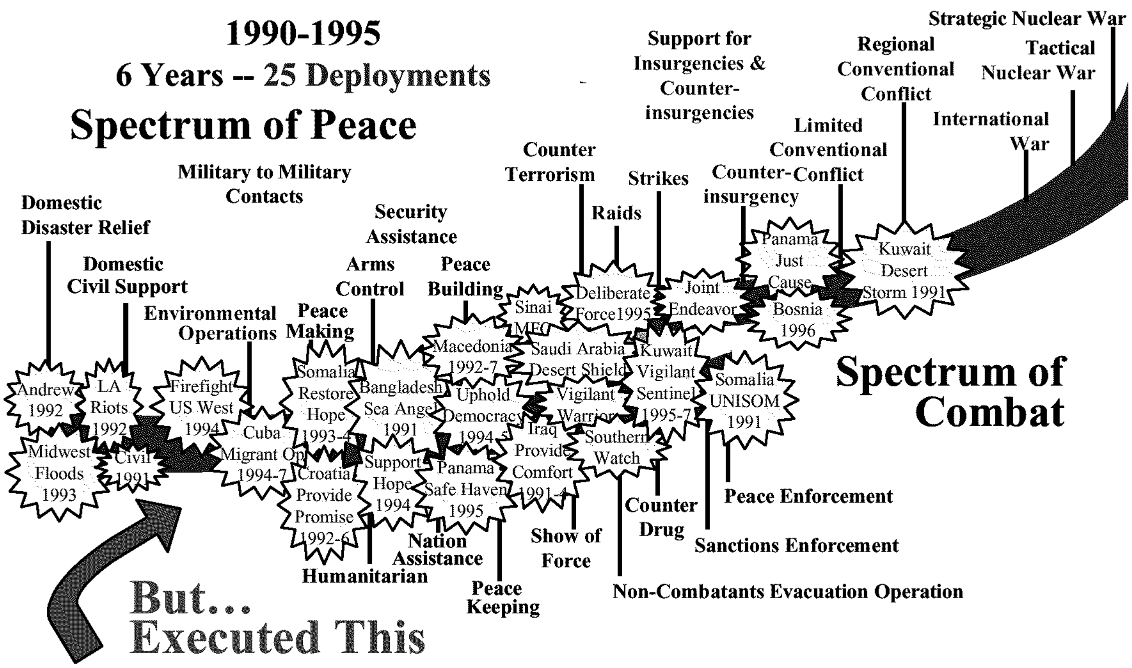


Figure 5. 1990-1996 Army Deployments (AV 2010, 1996)

likely seek asymmetric means of attacking US interests due to America's conventional military superiority. To effectively and efficiently respond to this changing/dynamic environment; the DoD would need state-of-the-art or "right-sized" organizations, infrastructure, legal and regulatory structure, and business practices. However, the system that ensured the US won the Cold War forced DoD's current support systems, structures, and practices to be outdated and antiquated (Cohen, 1998). According to the acting US Secretary of Defense in 1998: "DoD has labored under support systems and business practices that are at least a generation out of step…the defense establishment remains frozen in Cold War structures and practices" (Cohen, 1998). Since the DoD has a fixed budget, an excessive infrastructure siphons monetary resources that could be used for readiness and modernization efforts. The acquisition and procurement legislation and oversight rules limit DoD's ability to rapidly transition new technology, limit DoD's ability to negotiate and find "best value," and increase the price of doing business. For example, since there are only two competitors in the military aerospace industry, is it logical to attempt to use competitive market strategies while trying to negotiate an acquisition? Additionally, DoD is required to hold back a portion of business for disadvantaged and small businesses instead of finding the most efficient use of the money. Lastly, since any unspent DoD money at the end of a fiscal year is taken back and future budgets will likely reflect a reduction equal to the amount of the unused portion (since it is perceived that it is unneeded), there is an incentive to spend the entire budgeted amount and no incentive/reward to efficiently use the money. The Secretary of Defense acknowledged this mismatch between the current environment and DoD's ability to respond to it and addressed a plan of action:

> The fundamental challenge confronting the Department of Defense is simple, but daunting. U.S. forces must meet the demands of a dangerous world by shaping and responding throughout the next 15 years…To meet this challenge, the Department must prepare now to meet the security challenges of an unpredictable future. As the nation moves into the next century, it is imperative that it maintain its military superiority in the face of evolving, as well as discontinuous, threats and challenges…To maintain this superiority, the United States must achieve a new level of proficiency in its ability to conduct joint and combined operations. This proficiency can only be achieved through a unified effort by all elements of the Department toward the common goal of full spectrum dominance envisioned in Joint Vision 2010, the Chairman of the Joint Chiefs of Staff's blueprint for the future military operations. (Cohen, 1998)

This need for joint and combined planning and proficiency is discussed in the next subsection. With the new era of rapidly changing, unpredictable threats, the defense industry and the DoD needed to transform more than their ability to operate in a full spectrum of crises, they needed to implement a culture change to meet the new challenges of a new century. Reacting to the new challenges and the need for a culture change, the DoD established two corporate-level goals:

> Goal 1. Shape the international environment and respond to the full spectrum of crises by providing appropriate sized, positioned, and mobile forces. Goal 2. Prepare now for an uncertain future by pursuing a focused modernization effort that maintains U.S. qualitative superiority in key warfighting capabilities. Transform the force by exploiting the Revolution in Military Affairs, and reengineer the Department to achieve a 21st century infrastructure. (DoD, 2001)

The realization that the traditional approach (aggregate TPFDDs) used during the predictable cold war era would not effectively resolve the unpredictable challenges of the post-cold war environment (rogue nations with weapons of mass destruction, terrorism, ethnic tension, etc.), the decline of resource availability (personnel and budget), and the reduction of bases and personnel in forward based overseas locations, led to the creation of several DoD initiatives. These include the joint Precision Engagement and Focused Logistics and the Air Force's Global Engagement strategies; and the Navy's Ring of Fire, the Marine's Maneuver Warfare on Urban Terrain, the Army's Force XXI, and the USAF's AEF concepts.

The Navy implemented Ring of Fire to provide rapid, accurate, and effective fire support to joint forces. The Marines implemented Maneuver Warfare on Urban Terrain to improve flexibility and coordination of dispersed small units using maneuver warfare in urban areas (Cohen, 1998). The Army implemented the Force XXI concept to design organizations and capabilities to be light, rapidly tailored and deployed, and effectively employable in joint and multinational crises (Cohen, 1998). The USAF implemented the AEF concept to provide greater flexibility and the rapid deployment of highly capable forces with fewer resources for global engagement in peacetime, crisis, and war (Looney, 1996:5). AEF units are expected to conduct air operations independently for the first seven days until logistic sustainment is established (Godfrey, 1998: 2).

These concepts increase the criticality of selecting the best force package mix for deployment. As slack in the system, excess capacity, and redundancy of combatants and support resources are reduced, the importance of having the right forces in the first place increases. An error in planning can result in lives and resources lost as well as negatively affecting the conflict's final outcome. And, the need for rapid crisis response and deployment magnifies the need for rapidly tailorable force mix packages. Compounding the situation, timely selection of the best force package mix has become increasingly difficult with current initiatives to reduce and eliminate Unit Type Codes (UTCs) (Filcek's interview with Petersen, 2001: 12-13).

Previously, UTCs were large, general-purpose force packages that campaign planners used to tailor for specific mission requirements. They were easy to use and very convenient during short notice crises (Filcek, 2001: 12); however, no systematic rules exist for tailoring and planners must rely solely on their best guesses. The tailoring used a top down approach by taking large packages and cutting unnecessary items through intuition instead of using a systematic process that builds on the true marginal logistics requirements. This can overburden lift requirements or overlook/cut critically necessary logistics items, thus limiting the ability to rapidly deploy lethal forces. Because force mix selection seeks to get desired forces and support resources in the right amount to the right place at the right time; it may have the greatest impact on the logistics footprint size, the deployment speed, and force sustainment (Filcek, 2001: 13).

The Need for Merging.

Joint operations are maneuvers that involve coordination and cooperation of more than one US military service, while combined operations are exercises that involve more than one nation's military. Joint and combined planning are systematic processes that make joint and combined operation possible by determining the best assignment of mission preferences to tasks, tasks to resources, and resources to logistics requirements (JSOG, 1997: paragraph 500).

To maintain national security by developing the ability to respond to the full spectrum of crises, "the United States must achieve a new level of proficiency in its ability to conduct joint and combined operations...Achieving this new level of proficiency also requires improving the U.S. military's methods for integrating its forces and capabilities with those of its allies and coalition partners" (Cohen, 1998). The necessity to conduct joint and combined operations is compounded even more by several factors (JV 2010, 1997: 8):

- The reduction of military personnel to the lowest levels faced since the 1950s, while operations tempo dramatically increased
- The reduction of permanently overseas stationed forces
- Flat budgets with increasing overhead, readiness, and modernization costs
- Statutory mandate

This combination of factors has resulted in the situation where "simply to retain our effectiveness with less redundancy, we will need to wring every ounce of capability from every available resource" (JV 2010, 1997: 8).

The Goldwater-Nichols DoD Reorganization Act of 1986 sought to improve joint operations and create clear lines of authority and unified command over all services for joint operations by legally mandating jointness (Osgood, 1996). The Act was passed after investigations found joint command and control and planning shortfalls, such as the 1970 North Vietnam Sontay prison prisoner of war operation, the failed 1975 Mayaguez Iran hostage rescue operation, and the 1983 Grenada operation (Carlson et al, 1984).

Fragmented, uncoordinated joint and combined planning and strategy is eliminated with a unified war planning environment; additionally, unified planning between operations planners and logistics planners will result in more successful execution with lighter, leaner, and more lethal forces (Colvard, 2001: 8).

Nature of the Problem.

This subsection provides a specific description of the nature of the force mix value, the decision variables (force mix), and how the resource limitations restrict the force mix.

Since the purpose of this research was to develop and validate a structure and an approach, not discern a specific answer to a specific force mix and flow problem, the force mix values were notional and were represented in the Task Suitability Matrix (TSM) in Appendix D. This matrix establishes the task-weighted preferences between assets for missions. These task-weighted preferences represent the relative values, from 0 to 1, of each alternative asset/munitions configuration performing a specific task, based upon the analysis of the composite values (discussed in Chapter I). Since these values are campaign and location specific, a model would require different template values for the different scenarios. An MRR composite value of 1 would provide the commander with the best total value, based on all design and campaign specific issues. On the other hand, an MRR composite value of 0 would indicate an infeasible MRR, based upon either an infeasible design or campaign specific issue. Since each additional (feasible) MRR contributes additional benefit to the CINC, the additive composite values of the total MRRs would equal the total benefit the CINC received from an asset set. The additional benefit of each MRR is the probability of its mission success and campaign specific goodness, which is the MRR composite (task suitability) value. For example, if an F-16 deployed to conduct 2 missions a day, the total benefit per day of deploying the F-16 would be the benefit received from the F-16 performing the first mission plus the benefit received from conducting the second mission. Since a CINC receives value for each MRR assigned a task, the total value of the asset set (force mix) is the sum of all the incremental values. The values of alternative asset sets are compared within the lift and resource availability constraints. An asset mix with the highest value, within constraints, is preferred. Conversely, asset mixes with lower values, within constraints, are considered less desirable.

The decision variables represent the number of MRR asset types assigned to a task. The asset types can represent different aircraft (all services), same aircraft with different weapons configurations (all services), naval vehicles, army vehicles, or personnel with different skills and equipment. The variables represent one asset type performing one task. For example, if an F-16 is performing an Air-to-Air (AA) and a Suppression of Enemy Air Defense (SEAD) missions in the same day, this would be represented as two separate decision variables: one as an F-16 doing an AA and one as an F-16 doing a SEAD. Even if this is the same asset, it must be split by its different tasks to account for different suitability values and logistics requirements. Since a task must either be completed or not completed, partial missions are not feasible; the decision variables must be integer values.

Since assets like tanks, naval ships, aircraft, and trained personnel require a long lag time between demand time and production or training, the age of rapid deployment forces planners to use only currently available resources when developing force mixes. If there are only X number of assets/resources available in inventory and the current plan must be executed before more assets/resources will be available, the planner cannot develop a plan to deploy more than X resources/assets. Additionally, if resources needed to support an asset are not available, this limits the ability to use this asset and the usefulness of this asset. For example, if tanks are deployed for combat and no fuel for tanks is available in theater, then the tanks become useless after their initial fuel capacity is consumed. Since assets (like aircraft and tanks) require resources (such as fuel and ammunition) to accomplish missions, increasing the number of missions increases the amount of resources needed. Furthermore, resource consumption must be less than or equal to available resources. Therefore, resource availability restricts the mission supportability and selection (decision variable selection). For example, increasing a decision variable representing an F-16 conducting an AA task by a value of 1 will proportionally increase fuel and ammunition requirements. This increase in resources required must be available or the mission is not possible.

As mentioned in Chapter I, lift availability is also very limited. This forces planners to select force mixes that do not exceed the capacity of the available lift. For example, if airlift can only support deploying two tanks a day, planners cannot plan for more. Additionally, these constraints force planners to evaluate less suitable MRRs to maximize the total asset mix value. For example, assume there is only enough lift available for one more F-16 to perform an AA mission. Even though the F-15 may have a superior suitability for AA missions, the lift requirement would exceed lift availability. Although the F-15 is superior for AA missions, the resource limitation restricts the force mix to selecting the F-16 since it improves the number of sorties and value for the CINC.

Previously Used Methodologies.

As the starting point for determining what methodology to use and how to build a decision support tool for force mix optimization, a review of the methodology of previous attempts was conducted. This review focused on three major areas: Tomahawk Selection Optimization Model, the ALP and David Wakefield's Genetic Algorithm (GA) approach.

Tomahawk Selection Optimization Model (TSOM).

Scott Kuykendall's *Optimizing Selection of Tomahawk Cruise Missiles* was the most applicable and promising study found during the review of force tailoring tools. The TSOM was used to rapidly select the optimal (maximizing future strike flexibility and capability) missile location given a specified missile type for an assigned mission. TSOM was an integer program designed to support real-time single-platform (single ship or submarine) and battlegroup missile selection. It was proposed to replace the manual approach, which could require several hours to get missiles on target, and the Tomahawk Weapon Control System automatic missile selection, which often assigned missiles from sub-optimal locations (Kuykendall, 1998: 1).

The TSOM, regarding the MK 41 Vertical Launching System (see Figure 6 for this configuration), provided the following guidance (Kuykendall, 1998: 2-15):

- Each system contains eight modules which contains two half-modules each.
- Each half-module has four cells (except module 5) with a missile in each cell.
- Only one missile can be launched at a time from each half-module and both half-modules within a module can fire simultaneously
- Only missile types CIII (Block III conventional 1000-pound bullpup warheads), CII (Block II conventional), DIII (Block III sub-munition), and DII (Block II sub-munition) were considered
- Sustainment of CIII capability is preferred over CII, CII over DIII, and DIII over DII
- Selecting the correct missile type from a missile location that would leave a 16 salvo capability (ability to simultaneously fire 16 of the same missile type) for future requirements of all missile types provides the highest value or flexibility

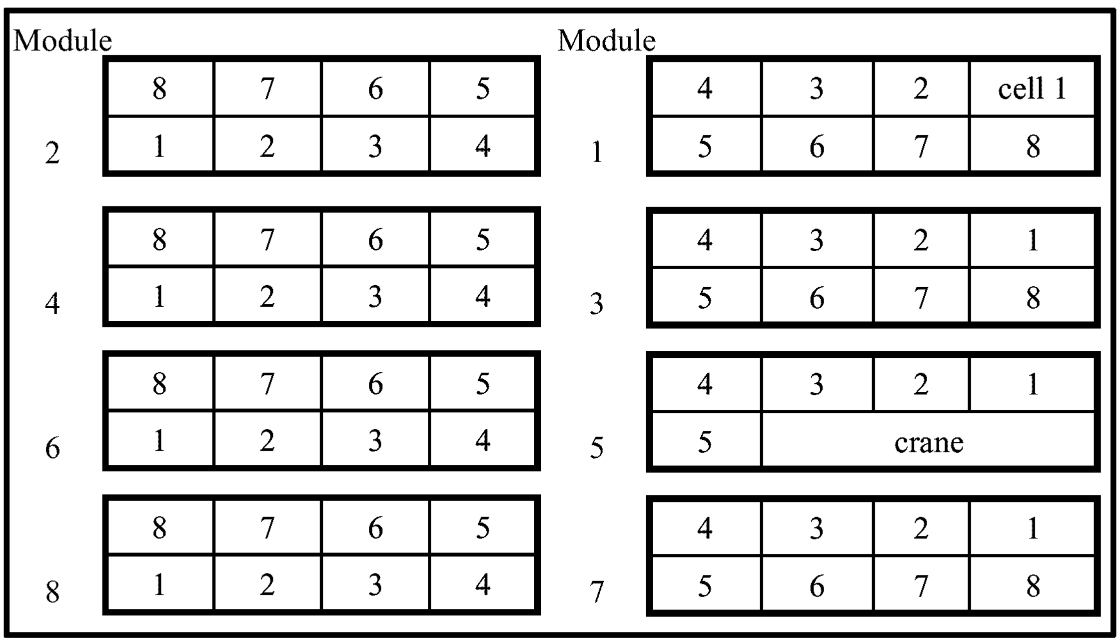


Figure 6. MK 41 Missile Location Configuration (Kuykendall, 1998: 4)

If a 16 salvo capability of all missile types is not possible, a 16 salvo of the CIII is preferred over all other missiles, the CII is preferred over the D series, and the DIII is preferred over the DII (Kuykendall, 1998: 2-15). To demonstrate optimal versus sub-optimal selection, Figure 7 provides an illustration using a given loadout. Given the

	Cell 8	7	6	5
Half-module 1	CIII	CII	CII	DIII
Half-module 2	CIII	CIII	CII	DIII
	1	2	3	4

Figure 7. Example of Two Half-Module Loadout

missile selection or tasking order of CIII and CII, there are several possible combinations that can satisfy this requirement. Selecting cell 8 for tasking 1 and cell 3 for tasking 2 is one possible solution; however, the maximum number of future CIII or CII salvo is one instead of two, which is sub-optimal according to the TSOM guidance above. If cell 1 for tasking 1 and cell 7 for tasking 2 had been selected instead, an optimal solution is generated since the greatest flexibility is achieved by allowing two simultaneous salvos of CIII (cells 2 and 8), CII (cells 3 and 6), or DIII (cells 4 and 5).

This was just one example of a launching configuration and loadout. Launching configuration and loadout can vary per ship/submarine or ship type, and a battlegroup configuration can contain any combination of these ships or submarines. Since a valid model can simply incorporate true operational loadouts/configurations and it is impossible to test all possible combinations, typical loadouts were tested for each ship class and various battlegroup combinations (Kuykendall, 1998: 21-22).

The TSOM was formulated using linear programming to maximize the objective function subject to real-world limitations and constraints (Appendix I presents Kuykendall's TSOM mathematical definitions and formulations). The objective function was designed to maximize future flexibility/capability by weighting decisions based on maximizing the value of future salvos. Penalty values were assessed for failing to accomplish primary, ready-spare, and back-up missions; as well as for selecting a submarine missile that was not preloaded in its torpedo tube. A bonus was earned for the number of remaining CIII missiles (Kuykendall, 1998: 19). This formulation encourages missile selection that leaves one of each missile type in each half-module, which promotes future flexibility/capability, while ensuring the current missions get accomplished. The constraints were designed to restrict decision alternatives to account for operating restrictions. The first constraint ensured the model represented the launching system limitation of one missile launched out of each half-module per tasking. The second constraint ensured that post-launch loadout per each missile type equaled pre-launch loadout minus missiles assigned (assumed fired) to primary missions. The third and fourth constraints ensured tracking of each missile type in each half-module available for future salvos, $SALVO_{sim}$. This was accomplished by having $SALVO_{sim}$ equal one for each half-module that contained one or more of each type of missile in the post-launch loadout or zero when a missile type has no missiles in the half-module. The $SALVO_{sim}$ binary values are multiplied by each representative missile type value and summed to create the objective function discussed above. The fifth, sixth, and seventh constraints ensured that only one missile could be assigned to a primary, ready-spare, and back-up missions and ensured that a penalty would be applied if any mission was unfilled. The eighth constraint ensured that a ready-spare is assigned to the same ship that is assigned the primary mission. The ninth constraint ensured that the backup is assigned to a different ship than the one assigned the primary mission. The tenth and eleventh constraints define the relationship among the variables for primary, ready-spare, and backup missions.

This approach provided valuable insight for the force mix problem. The model proved robust for sensitivity analysis, and the accelerated speed provided additional rationale for exploring the use of linear programming for the force mix problem. Additionally, the concept of applying a value to asset-mission assignments was directly applicable to the force mix problem. Although Kuykendall's research showed many similarities between selecting the best mix of assets to missions with constrained resources, it only evaluated the selection of a missile location and did not include evaluating an asset's suitability against specific targets or the effects of decisions on the logistics footprint, which may limit the suitability of using this approach.

The ALP.

The ALP is attempting to develop a multi-echelon, distributed computing architecture that will create a near real-time deployment planning process for military forces. The architecture attempts to automate logistics plan generation, perform real-time situation assessment, end-to-end movement control, and rapid supply (Carrico, 2000). This deployment planning architecture will enable logistics planners to significantly reduce planning process time for situation-tailored logistics plans (Carrico, 2000). This time reduction is achieved by using an automated system that integrates logistics and operations while also promoting seamless planning and execution (Filcek, 2001: 20). Although using a single system promotes seamless integration and significantly reduces process time, a single program/model could possibly yield even greater results.

As mentioned in Chapter I, the ALP was developed to speed up the sourcing and tailoring of existing TPFDDs. Although automation speeds up the process, this methodology still promotes sequential processing which takes longer than parallel processing. For example, operations and intelligence planners develop operations plans which are then sent to logistics planners. Logistics planners then determine if the plan is supportable and decide how to tailor existing TPFDDs. The plan then returns back to the operations planners for re-planning or implementation. Several iterations may be required to balance operation plans with logistics constraints. If a model could be built that incorporated operations asset requirements in parallel with logistics requirements and constraints, time could be saved.

The ALP, in its fifth year, is a single system that uses software agents to automate and manage logistics plan generation, execution monitoring, end-to-end movement control, and rapid supply and sustainment (Carrico, 2000b). Software agents, rather than an object-oriented design methodology, were pursued to facilitate a process focus instead of a data orientation. The agents are the foundation of the ALP Architecture Cluster as illustrated:

> Software agents are semi-smart, programmable pieces of software that can automate routine information processing, planning and monitoring activities. They can perform these functions relative to a set of processes and business rules appropriate for their domain and perform these actions, tailored to the requirements of the situation, in coordination with other humans, organizations and entities participating in the process. (Carrico, 2000b)

The general purpose agents and clusters, plus the domain knowledge and business rules/processes create domain specific agents that represent combat units, support units, and command and control responsibilities (Carrico, 2000b; Shaneman, 1999: 7). Agent communities are composed of several interconnected clusters (such as Wings, TRANSCOM, CENTCOM, Army AMC, Battalions, etc) and agent societies are composed of several interconnected communities (Carrico, 2000b). For the Air Force, deployment order initiation begins the ALP contingency process. Deploying unit clusters allocate resources and forces to AEF clusters and AEF clusters use decision rules to tailor requirements for the Logistics Readiness Center (LRC) community (Filcek, 2000: 20; Carrico, 2000b). The LRC uses an optimization routine to source requirements and forwards the results to the US Transportations Command (TRANSCOM) community. The final step involves TRANSCOM's balancing of these requirements with other communities (services) deploying forces while optimizing the airlift sequence and speed and reducing the logistics footprint (Shaneman, 1999: 7)

The ALP demonstrated its feasibility by building a Level 5 (highly-detailed) logistics plan for the US Army's 3rd Infantry Division in less than one hour in 1998 (Carrico, 1998: 5), and a realistic Level 6 deployment plan (including support and sustainment) for a major joint deployment in 2000 (Carrico, 2000b). The success of the 1998 test was especially noteworthy, since the demonstration was conducted using standard personal computers over standard internet bandwidth extracting real-time data from the Joint Total Asset Visibility, Global Transportation Network, and Global Decision Support System databases to build detailed plans (Carrico, 1998: 5). Timelines can be significantly reduced since all critical players have instantaneous access to detailed joint logistics plans generated in the ALP architecture.

This new model could also give operations planners visibility into how their decisions affect sustainment and are affected by logistics constraints. To take advantage of the power of ALP, researchers at AFIT proposed that building-up capability based on incremental units of capability and logistics requirements was more effective and efficient than breaking-down existing TPFDDs. Additionally, ALP has been unable to expediently select the "best" logistics plan from competing alternatives and the "best" logistics plan selection cannot begin without the selection of the best force mix (Buzo, 2000: 1). While the ALP architecture does not solve the force mix problem directly, it will (for the first time) make available in real-time the information required to formulate and solve this problem.

<u>David Wakefield's GA Approach.</u>

David Wakefield's GA modeling approach attempted to build a model that identified a preferred force mix by balancing resource suitability and lift requirements. "An Evolutionary approach was applied to a tri-objective constrained optimization problem with 15 decision variables with the goal of producing five Pareto optimal sets of force mixes corresponding to five progressively larger sortie capability levels" (Wakefield, 2001: x). The three objectives were to maximize task suitability, minimize lift weight, and minimize lift volume. Task suitability refers to the aircraft's designed effectiveness/suitability to perform specific aerospace missions. The first objective, maximize task suitability, promotes selecting an MRR set or force mix that is "best" suited for the specified missions. The force mix provides a certain suitability/capability to the CINC, but at a cost: lift resource consumption. The finite military lift capability is a key constraint on the amount and timing of resources flowing into an area of operation. Therefore, it is desirable to deliver an MRR combination that minimizes lift. Airlift is constrained not only by the amount of space available in the cargo aircraft, but also by the amount of weight a cargo aircraft can transport. Since the goal of the planners is to provide the CINC the best combination of MRR that satisfy the time phased needs for these resources and provide the greatest combat capability, the competing objectives (minimizing lift consumption and maximizing suitability) must be balanced.

The 15 decision variables included 5 different resources capable of doing 3 different tasks. Each of the 5 resources (MRR) represents a combination of an asset type and its requirements, e.g. aircraft, pilot, fuel, munitions, support equipment and personnel, etc., that has a designed suitability for a single task. To illustrate, assume that a notional aircraft F, has three configurations, F_A, F_B, and F_C, which constitutes three MRR types. The tasks represent the types of missions, targets, or tasks that the CINC desires to be resolved by the MRRs. In this research, they represent Air-to-Air, Air-to-Ground, and Precision Bombing tasks. The task mix or preference can be defined as the set of points along the resource/capability levels within a given campaign phase (see Table 1).

Table 1. Desired Capability Matrix (Wakefield, 2001: 51)

	TASK (sorties per day)			
Capability Levels	***AA***	***AG***	***PB***	**DECISION SPACE CARDINALITY**
16	10	5	1	630,630
32	10	20	2	159,549,390
75	19	45	11	$\approx 2.56 \times 10^{12}$
150	30	75	45	$\approx 1.48 \times 10^{16}$
300	60	90	150	$\approx 4.37 \times 10^{19}$

The capability/resource levels represent different levels of total activity (resource availability). To demonstrate, assume that only four notional F_A aircraft could be prepared and flown four times per day, then it would represent a capability level of 16 (16 sorties available per day). The five different capability levels are identified in Table 1. The capability levels are 16, 32, 75, 150, and 300. To illustrate, the first capability level of 16 is broken down into 10 AA, 5 AG, and 1 PB tasks. The task mix for the problem is somewhat arbitrary. For a real-world situation, the task mix is proportional to the commander's preference curve.

Since the combinatorial nature of the research was assumed to be very large, it necessitated the use of a heuristic and a relatively small set of assets with which to explore the algorithmic search for an acceptable solution (Wakefield, 2001: 5). For example, with 300 sorties per day, it was estimated that $\approx 4.37 \times 10^{19}$ different solution combinations were possible. Therefore, real world problems with even more possible solutions remain hard or nearly impossible to solve with deterministic methods, and the combinatorial nature of the problem may require heuristic approaches.

Meta-heuristics, like GA and Tabu Search, were developed to deal with these increasingly more complicated problems by balancing computational processing time with solution quality. Heuristics provide approximate solutions (good solutions with no guarantee of optimality) with less computational processing. A meta-heuristic refers to a master strategy that guides other heuristics to explore solutions beyond local optimality to find high quality global solutions (Glover, 1999: 17). GA seeks to replicate the biological phenomenon of evolutionary reproduction, where populations (potential solutions) evolve through *natural selection* and *survival of the fittest* (Glover, 1999: 1).

GAs use this population based approach to produce new potential solutions "offspring" by randomly determining which characteristics "hereditary traits" of the of existing solutions "parents" to combine and randomly selects which existing solutions will be parents (Glover, 1999: 18). Since population size is fixed; old, weak individuals are discarded or killed off. Over many generations the hereditary traits associated with higher quality solutions tend to dominate the surviving solutions. This approach uses randomization to prevent cycling, reduce dependence on memory, and increase simplicity.

Cycling, revisiting previously evaluated solutions, is undesirable since it wastes computational processing time and reduces the chance of finding high quality solutions within time constraints. Since heuristics are used because of their ability to expeditiously find *good* solutions, cycling would eliminate any timesavings. GAs use randomization to achieve a diversifying effect thus preventing cycling. Since each solution has an equal chance of being selected using randomization, problems with realistically sized solution spaces have a low probability of revisiting previously evaluated solutions (cycling). This promotes selecting unique (diverse) solutions for evaluation. Since diversification is achieved, the algorithm does not need to use exponentially large amounts of computer resources to store information and history of previously visited solutions or maintain numerical accuracy after hundreds of thousand to millions of mathematical calculations. The need for large amounts of computational memory are eliminated in GAs by only storing attributes of the best found solution and not storing data about any other previously visited solutions. The assumption that revisiting previous solutions is low supports tracking only the attributes of the best solution. Since the method is "memoryless," no special programming is needed to track solution history or attributes.

> From an abstract standpoint, there is clearly nothing wrong with equating randomization and diversification, but to the extent that diversity connotes differences among elements of a set, and to the extent that establishing such differences is relevant to an effective search strategy, then the popular use of randomization is at best a convenient proxy (and at worst a haphazard substitute) for something quite different. (Glover, 1999: 101-2)

The reader is referred to Appendix J for the mathematical formulation that David Wakefield used to solve the force mix problem. His Multiobjective Messy Genetic Algorithm (MOMGA-II) was modeled after the ALP Pilot Problem and used notional, but not arbitrary, coefficients to represent MRR suitability values, weight requirements, and volume consumption. The MOMGA-II was formulated using mathematical programming to present a Pareto optimal set of force mixes to the war planner for selecting the desired force mix within constraints. An optimal set of force mixes are presented for each capability level, instead of a single optimal force mix, since the three separate objective functions may represent three different optimal force mixes. Each Pareto optimal set must be balanced using some decision making methodology to generate a preferred MRR mix.

The first objective function was designed to maximize the suitability of the MRR set to accomplish required tasks by summing the product of the MRR suitability and associated quantity. Based on the linear weight requirement assumption, the second objective function was designed to minimize the MRR set total weight requirements, calculated by summing the product of the MRR type quantity by the MRR type weight coefficient. Based on the linear volume requirement assumption, the final objective function was designed to minimize the MRR set total volume consumption, calculated by summing the MRR type quantity by the MRR type volume coefficient.

The 10 constraints were designed to restrict the decision alternatives to model reality. The first constraint ensured that a negative number of MRR types could not be assigned to tasks. Since it would be impossible to assign a negative number of aircraft to accomplish a task, this is a relevant restriction. It is also impossible to assign a fractional number of aircraft to accomplish a mission. The second constraint remedied this restriction by ensuring MRR only assume integer values. The third, fourth, and fifth constraints ensured that one MRR was assigned to exactly one task and that tasks were accomplished in the exact proportion specified by the decision maker at each capability/resource level. Since an MRR type can only be assigned to a task if one is available, this restriction must be included in a valid model. The remaining constraints ensured that the quantity of each MRR types assigned to tasks were available to be assigned. To illustrate, assume ten F-16s are available. The F-16s can be assigned to any combination of tasks, but only if the total quantity of F-16s assigned is less than or equal to ten.

There are several reasons this method was rejected. First, processing and computational speed and computer memory capability continues to grow exponentially, making deterministic models more feasible and desirable. Second, many meta-heuristic methods are not generalizable. New problems require new procedures, since applications generally require highly problem-specific designs (Frontline, 2000: 15). Third, the criticality of selecting the best force package mix, as identified in the Cold War CONOPS Change and The Need for Merging subsections, makes finding the true optimal value more important. Since heuristic models cannot guarantee optimality, deterministic models are preferred. Fourth, "MOMGA-II does not go far enough to prevent convergence to an infeasible front" (Wakefield, 2001: 70). Finally, this methodology requires extensive knowledge in programming to create and use the model while enumerative deterministic models can be created in user-friendly software such as Excel. For example, the Excel compliant Large-Scale LP Solver "uses advanced matrix factorization methods such as LU decomposition and dynamic Markowitz refactorization, for both speed and numerical stability…capable of handling very large problems of up to 65,000 variables and 65,000 constraints" (Frontline, 2000: 12). Since a deterministic, exact approach to the problem shows promise, and has not yet been tried, further consideration was warranted.

Solution Approaches

This subsection discusses some basic aspects of the various solution approaches: modeling, mathematical programming and modeling, mathematically defining the problem, linear programming, and large integer linear programming. This research effort constructs a general modeling approach for force mix selection that is applicable over the wide range of problems on the crisis-planning spectrum illustrated in Figures 4 and 5.

Modeling.

Modeling is a management approach that uses a simplified version of a problem or object to represent the relevant characteristics to be studied. The benefits of modeling are that they are usually simplified versions of the problem, less expensive to analyze, provide information in a more-timely manner, safer in certain circumstances, provide insight into things that would be impossible to do in reality, and improve decision maker insight and understanding of the problem (Ljung, et al, 1994: 14; Ragsdale, 2001: 4-5; Smith, 1977: 7; Chang, 1990: 258).

Modeling is very important in the military for four reasons. First, timely delivery of information can mean the difference between life and death. Second, it would be impossible to perform certain important tests (such as live fire tests to determine B-2 pilot survivability against Surface to Air Missiles) since it is too risky and potentially too expensive. Third, it is less expensive to build representative models of aircraft and tanks to determine weapon system feasibility. Finally, low cost models can find flaws and avoid costly mistakes before full-scale production. For example, "it is far less costly to discover a flawed wing design using a scale model of an aircraft than after the crash of a fully loaded jet liner" (Ragsdale, 2001: 4). A precautionary statement is necessary: a model is only valid if it accurately represents the relevant characteristics of the problem in question.

There are three basic types of models: mental, visual, physical or scale, and mathematical (Ljung, et al, 1994: 14-15; Ragsdale, 2001: 2; Williams, 1985: 3). A mental model is a decision analysis approach where alternatives are evaluated within the decision maker's mind, such as mentally determining how to best utilize the workday. For complex problems like force mix determinations, a mental model would be impossible and inadequate. A visual model is a drawing or map, such as blueprints, that helps a decision maker evaluate various routes or layouts. Since force mix determinations involve evaluating how to optimize hypothetical quantities of assets to meet requirements within constraints, visual models would be infeasible. A physical or scale model is a method of analyzing or evaluating physical objects to determine their capability, feasibility, or risk. Since force mix determinations are not physical objects or concerned with determining the physical characteristics of a specific object, physical or scale modeling would be inappropriate. A mathematical model uses mathematical relationships to evaluate problems. Force mix determinations try to maximize combat capability within constrained resources, and these relationships and objectives can be represented mathematically, mathematical modeling would be appropriate. Mathematical models are usually represented in computer programs to address questions about the model's behavior (Kelton, 1998: 6). A discussion of computer modeling follows while a detailed review of mathematical programming and modeling will be expanded in the next subsection.

Today, decision makers face a fast-paced, rapidly changing competitive environment with complex problems that are not easily solved and involve numerous possible solution options. Selecting the best course of action and assessing these solution options embodies the fundamental nature of decision analysis. Trying to cope with the complexities of managing in the "real-world," managers discovered that using computer models, such as spreadsheets, was one of the most effective means of assessing the different solution options and selecting the best course of action (Williams, 1985: 229; Ragsdale, 2001: 1).

To establish the foundation of the term computer model for this research, the research uses the following definition: a set of mathematical relationships and logical assumptions (mathematical modeling) that represent the physical processes being analyzed implemented in a computer (Ragsdale, 2001: 1; Ljung, et al, 1994: 169-170).

Mathematical Programming and Modeling.

Mathematical programming (MP) is a field of management science that attempts to optimize individual or business objectives within limited resources using mathematical relationships that represent physical processes being analyzed (Smith, 1977: 7; Williams, 1985: 3-5; Ragsdale, 2001: 16). As mentioned above, mathematical models are quite different from physical or visual models. These models use mathematics, structural and quantitative approximations and assumptions, as well as logic to describe how the system works or will work. A simple example can illustrate how mathematical models can represent or describe a problem:

$$\text{REVENUE} = \text{SALES PRICE X QUANTITY SOLD} \qquad (2.1)$$

Equation 2.1 describes the relationship between revenue, sales price and quantity sold. This is a mathematical relationship that describes how revenue is a function of individual sales price and the quantity sold. Since equation 2.1 helps management evaluate alternatives to optimize revenue within quantity available constraints and uses mathematical relationships, this formula is a simple mathematical model.

MP techniques have been successfully used in solving product mix, routing and logistics, and financial planning problems (Ragsdale, 2001: 17; Williams, 1985: 64-68). Since determining manufacturing product mix (what products to make and in what quantity with a variety of different amounts of constrained resources) uses a nearly identical methodology as finding military force mixes, this provided strong, additional rationale for using this approach.

There are three categories of mathematical models: descriptive, predictive, and prescriptive (Ragsdale, 2001: 7-8). A summary of each model category and their characteristics is presented in Table 2.

Table 2. Categories and Characteristics of Mathematical Models (Ragsdale, 2001: 8)

Model Characteristics			
Category	**Form of f(·)**	**Values of Independent Variables**	**Management Science Techniques**
Prescriptive Models	known, well-defined	known or under decision-makers control	Linear Programming, Networks, Integer Programming, CPM, Goal Programming, EOQ, Nonlinear Programming
Predictive Models	unknown, ill-defined	known or under decision-makers control	Regression Analysis, Time Series Analysis, Discriminant Analysis
Descriptive Models	known, well-defined	unknown or uncertain	Simulation, Queuing, PERT, Inventory Models

Descriptive models are used when decision makers have a well-defined functional relationship between the independent and dependent variables, and the exact values of the independent variables are uncertain or unknown. These models describe outcome or behaviors of a given operation or system. Since this research assumes that the functional relationship and independent variable values are known, this method was considered unsuitable for this research.

Predictive models are used to predict the value of the dependent variables when the functional relationship is unknown, and independent variables take on a specific value. Since this research assumes that the independent values are known, predictive models may be unsuitable for this research.

Prescriptive models are used to help decision-makers decide what action to take when the functional relationship is well defined and the independent variables are known. Because of these characteristics, this approach appears well suited for this research problem.

Mathematically Defining the Problem.

Since mathematical programming attempts to optimize resource allocation, optimization problems must be able to be defined in mathematical terms or symbols to represent the decisions, constraints, and objective (Ragsdale, 2001: 18-19; Williams, 1985: 3-7). The decisions may represent the amount of different products to manufacture or the amount of different aircraft to deploy to a conflict. The values or outcomes of the optimization problem decisions can be represented by a set of variables:

$$X_1, X_2, \ldots, X_{n-1}, X_n \qquad (2.2)$$

These variables are usually referred to as the decision variables; additionally, the exact symbols used are a matter of personal preference. The constraints are the limits on the amounts of the resources available. Resource consumption can be expressed as a function of the decision variables to represent some type of limited resources that apply to the situation. The nature of the constraints can be expressed in three general ways to represent the problem in relation to a specific value: an equal to constraint, a less than or equal to constraint, or a greater than or equal to constraint. For example, if there are only 50 aircraft to fly missions, the constraint on the amount of the aircraft resource would be less than or equal to 50. The objective in optimization problems is represented by the objective function, such as:

$$\text{MAX (or MIN): } f_0(X_1, X_2, \ldots, X_{n-1}, X_n) \qquad (2.3)$$

"The objective function identifies some function of the decision variables that the decision maker wants to either MAXimize or MINimize" (Ragsdale, 2001: 19). Equation 2.3 is expressed as a function since the model can represent a linear or nonlinear relationship between decision variables with or without interactions between these variables. In general terms, this infers that the objective function is a mathematical statement describing the relationship between the objective the decision maker wishes to optimize and the value of the decision variables in the problem. This assumes that the objectives can be clearly identified, mathematically represented, and are limited in number.

The objective function is also assumed to be the sole criteria for choosing between feasible values of the decision variables. The standard mathematical optimization format, indicating both the desired outcome (objective) and resource limitations (constraints), would generally take the form of:

$$\text{MAX (or MIN): } f_0(X_1, X_2, \ldots, X_{n-1}, X_n) \quad (2.3)$$
$$\text{Subject to: } f_1(X_1, X_2, \ldots, X_{n-1}, X_n)=b_1 \quad (2.4)$$
$$f_2(X_1, X_2, \ldots, X_{n-1}, X_n)\leq b_2 \quad (2.5)$$
$$f_3(X_1, X_2, \ldots, X_{n-1}, X_n)\geq b_3 \quad (2.6)$$

This general MP model can be used with many kinds of functions and constraints, giving it great diversity. These functions, represented by f_0 in equation 2.3, can be linear or non-linear with or without interaction between the variables. The specific MP technique called linear programming, which solves optimization problems with linear objective functions and constraints, will be examined.

Linear Programming (LP). There are various methods for formulating problems as linear programs. Two specific requirements needed to formulate problems as an LP model are linear objective functions and constraints. Murty describes LP as follows:

> [It] deals with problems in which a linear objective function is to be optimized (i.e., either maximized or minimized) subject to linear equality and inequality constraints and sign restrictions on the variables. To formulate a real life problem as a linear program is an art in itself. Even though there are excellent methods for solving a problem once it is formulated as a linear program, there is little theory to help in formulating the problems in this way. (Murty, 1976: 1)

Formulating an optimization problem as a LP model requires proportionality, additivity, continuity of variation, divisibility, and deterministic coefficient assumptions (Murty, 1976: 2-3). Proportionality assumes that each variable's contribution to the value of the objective function is proportional to the level/value of the variables, and each variable is independent of each other. Additivity assumes the total objective function value can be obtained by adding the individual contributions of the different variables. Continuity of variations assumes that the decision variables can be all real values in its range of variation. To illustrate, variables that represent real world integer decisions must be modeled as an ILP. Divisibility assumes that variables can be any non-negative values including fractional ones. Deterministic coefficients assume that the decision variables values are know with certainty and do not vary. The general LP formulation according to Williams, Murty, and Ragsdale was stated as:

$$\text{MAX (or MIN): } c_1X_1 + c_2X_2 + \ldots + c_nX_n \tag{2.7}$$

$$\text{Subject to: } a_{11}X_1 + a_{12}X_2 + \ldots + a_{1n}X_n = b_1 \tag{2.8}$$

$$a_{k1}X_1 + a_{k2}X_2 + \ldots + a_{kn}X_n \leq b_k \tag{2.9}$$

$$a_{m1}X_1 + a_{m2}X_2 + \ldots + a_{mn}X_n \geq b_m \tag{2.10}$$

This model differs from the general MP model since the objective function and constraints are restricted to a linear function of the decision variables; the coefficients (c and a) must be proportional, additive, and deterministic; and each variable must be independent of each other (no interactions).

The process of taking a problem like force mix selection and describing it algebraically in LP format is known as model formulation. Ragsdale identifies five steps for formulating LP models: understand the problem, identify the decision variables, state the objective function as a linear combination of the decision variables, state the constraints as linear combinations of the decision variables, and identify any upper or lower bounds on the decision variables (Ragsdale, 2001: 20-21).

Although the "understanding the problem" step appears obvious, people's enthusiasm to solve a problem often causes them to start programming before they understand the problem. "If you do not fully understand the problem you face, it is unlikely that your formulation of the problem will be correct" (Ragsdale, 2001: 21). The decision variables are the fundamental decisions that must be made to solve the problem. This means determining what $X_1, X_2, \ldots, X_{n-1}, X_n$ represents. Stating the objective function as a linear combination of the decision variables involves creating an objective function that algebraically expresses the relationship between the variables to be maximized or minimized. For example, a $175 profit earned on each chair (X_1) sold and $200 profit for each couch (X_2) sold can be modeled as:

$$\text{MAX:} \quad 175X_1 + 200X_2$$

The function above calculates (and seeks to maximize) the total profit for whatever values X_1 and X_2 could take. However, the range of values for X_1 and X_2 is surely limited (constrained). Stating the constraints as linear combinations of the decision variables places limits on decision variable selection according to resource availability. To demonstrate, assume each chair requires 10 board-feet, each couch requires 20 board-feet, and only 3000 board-feet are available at the company. This resource limitation can be modeled as:

$$10X_1 + 20X_2 \leq 3000$$

Identifying any upper or lower bounds on the decision variables includes mathematically representing additional constraints. For example, selling a negative number of chairs or couches is not feasible/possible and can be modeled as follows:

$$X_1, X_2 \geq 0$$

The explicit purpose of linear programming is to identify an optimal solution, as represented by the values of the decision variables. The optimal solution is a solution that either maximizes or minimizes the objective function (depending on the problem objective) and is within the feasible region (Murty, 1976: 19). The feasible region is "the set of points or values that the decision variables can assume and simultaneously satisfy all the constraints in the problem" (Ragsdale, 2001: 29). There are 5 possible conditions that may arise: unique optimal solution, alternate optimal solutions, redundant constraints, unbounded solution, and infeasible solution (Williams, 1985: 21-35; Ragsdale, 2001: 31-33; Frontline, 2000b: 39).

When an optimal solution occurs at some Right-Hand Side extreme point (points in the feasible region where two or more constraint boundaries intersect) that intersects the objective function at the extreme point, it is a unique optimal solution. This is illustrated graphically in Figure 8a. The graph represents the model by presenting the objective function values at different decision variable points, the constraints as line segments, and the inequality area of the constraints as the feasible region. When more than one feasible point maximizes or minimizes the objective function value, they are alternate optimal solutions. Figure 8b shows alternate optimal solutions. When a

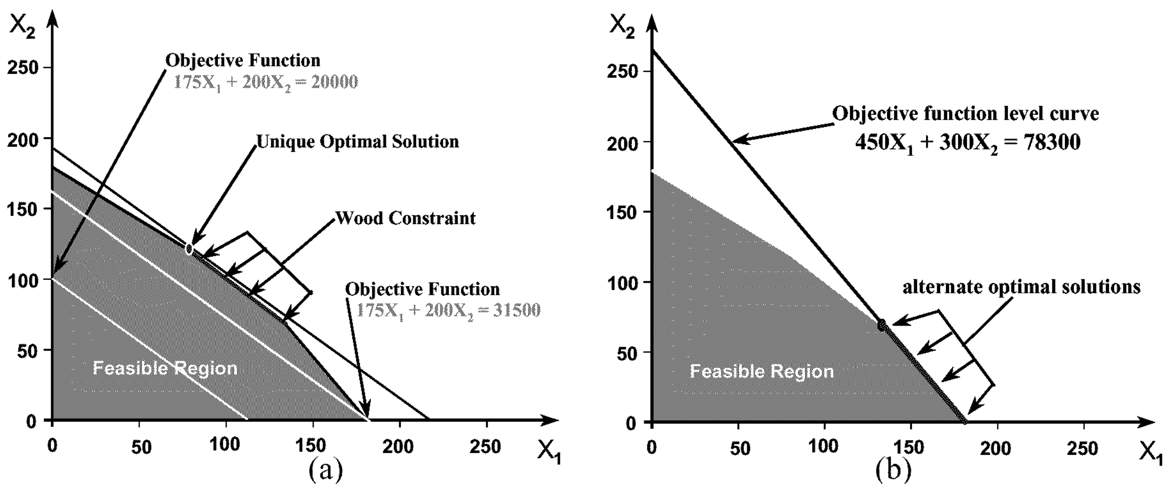


Figure 8. (a) Unique Optimal Solution. (b) Alternate Optimal Solutions

constraint is not relevant in determining the feasible region in the problem, it is a redundant constraint. Figure 9a shows this condition. When the objective function can

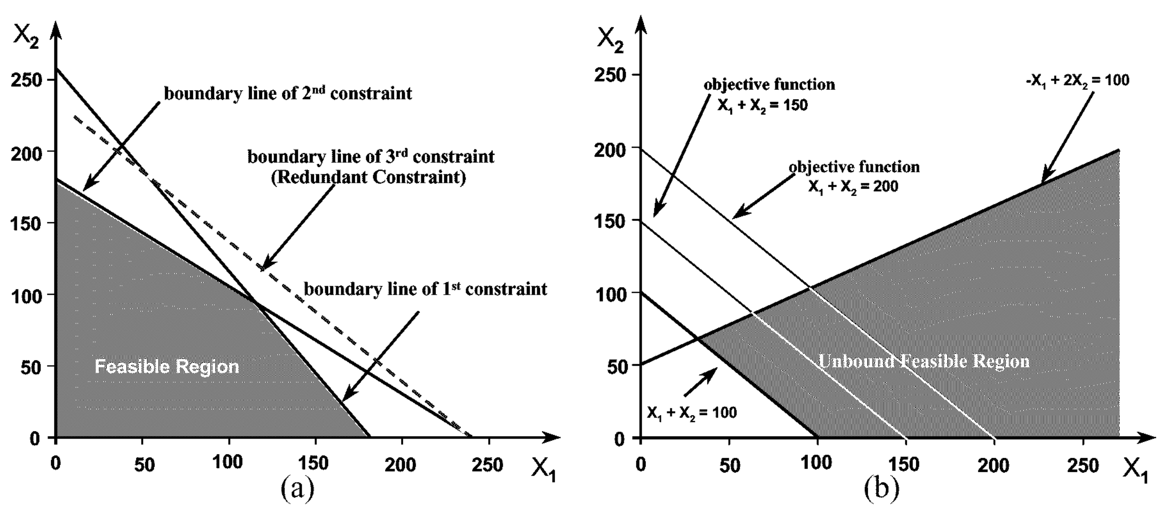


Figure 9. (a) Redundant Constraint. (b) Unbounded Solution

be made infinitely large, an unbounded solution occurs. Figure 9b shows this condition. When no solution exists that satisfies all problem constraints, it is an infeasible condition. This can be caused by the inability to solve all constraints or formulation/input errors. Figure 10 shows this condition.

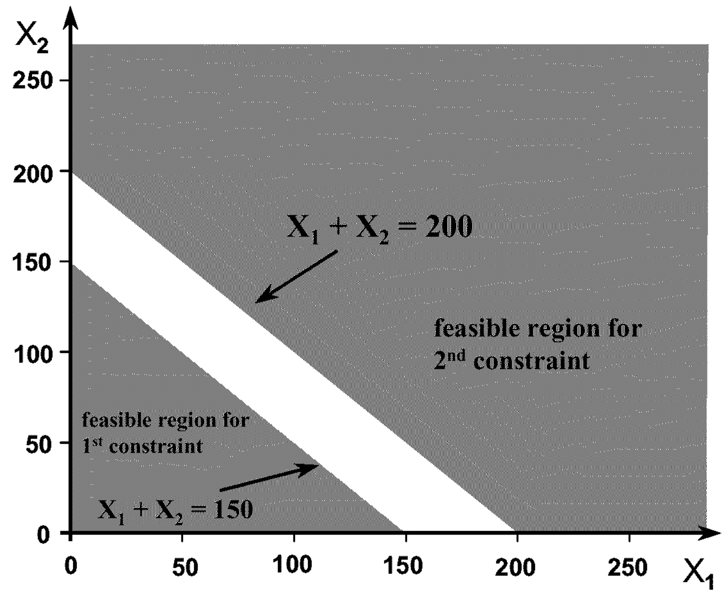


Figure 10. Infeasible Solution

There are several shortfalls to using LP techniques. First, properly defining the model can be difficult. Second, it can be very time consuming to build a valid model. Third, the model builder must have a thorough knowledge of model building and all extensive knowledge of the problem. Fourth, since the real world is very complex, it is impossible to mathematically represent all contributing factors in a model.

These shortfalls evaluated before making the decision to use this method. LP was chosen as a valid methodology for the force package mix problem for four reasons. First, the objective function of force mix problems is the sum of the different asset-task suitability values multiplied by the number of assets fulfilling these roles, making it a linear combination of the decision variables. Second, the constraints are a linear function of the decision variables. Third, modeling relevant, although not all, factors should adequately represent the problem and provide a valid tool for decision makers. Finally, off-the-shelf software, like Excel, has reduced the knowledge necessary for modeling problems. In the next subsection, the LP technique called integer linear programming, which solves optimization problems when some or all of the decision variables must assume an integer value, will be examined.

Large-Scale Integer Linear Programming (LSILP).

LSILP differs from LP on two key characteristics: large scale and the requirement for integer-only variable values.

Although there is no precise definition of what constitutes "large" scale problems, and the definition changes depending on the tool or user, Brook describes it as:

> What is a large model? The answer varies. It is one that takes a lot of time (or money) to solve. It is one that "just fits" into memory available on your machine. It is one that has more than a few hundred lines of assignment and equations when written in GAMS. Briefly, any model that is expensive to solve or difficult to manage, or whose details are so overwhelming that it is hard to keep track of them is large. (Brook, 1992: 166-167)

Large scale problems are difficult for two basic reasons: a large decision space to search and large memory requirements to track the solution search and candidate results. "Such problems require different algorithmic methods to manipulate large amounts of data, and to maintain numerical accuracy after hundreds of thousands to millions of floating-point arithmetic calculations" (Frontline, 2000: 9).

ILP is a natural extension of LP when the decision variables represent discrete choices such as planning an aircraft for a sortie or not. Since many real-world problems, activities, and resources, like people, tanks, ships, aircraft, or sorties are indivisible, they require the determination of yes-no decisions or integer values. For example, if the Army is trying to decide how many tanks to send into combat, it must obtain an integer solution because the Army cannot deploy a fraction of tanks. The Theater Attack Model is an example of large-scale linear programming's applicability to DoD problems (Jackson, 1989: 1). This model performs trade-offs analysis of the impact of aircraft and munitions effectiveness, weather, length of mission, and other factors on the effectiveness of a battle plan. According to many experts, "most optimization problems of a combinatorial nature can be formulated as integer programs" (Murty, 1976: 397).

ILP can be classified into two categories: pure, all integer programs or mixed integer programs (Murty, 1976: 397). Pure integer programs are models where all the decision variables must assume integer values. Mixed integer programs are models with some integer and some continuous decision variables.

"Although it is easy to state integrality conditions for a problem, such conditions often make a problem more difficult (and sometimes impossible) to solve" (Ragsdale, 2001: 231). The difficulty and time required to find an optimal solution are increased since the search space is expanded beyond the feasible region extreme points to all integer points within the feasible region. For example, instead of calculating the objective function values for only four extreme points in the Figure 11(a) case, under the LP method, all eleven feasible integer solutions must be identified and calculated in the Figure 11(b) case. In this simple example, the integer condition represents a 275 percent increase in complexity between the LP and ILP methods.

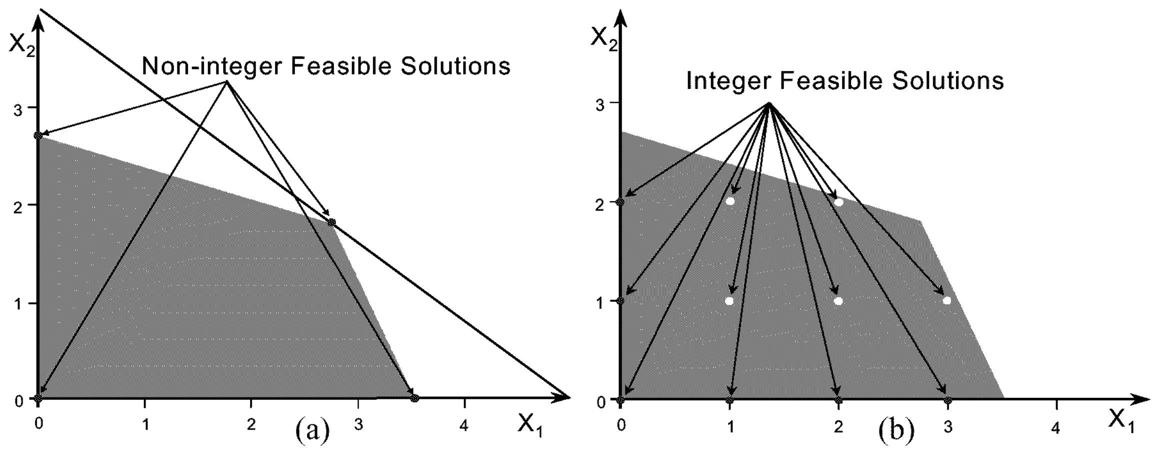


Figure 11. Integer and Non-integer Solution Space Comparison

There is an additional problem with integer conditions applied to the LP. It is not sufficient to simply "round off" the LP version of a problem in order to achieve integer results. As seen in Figure 12, using rounding to obtain an integer solution is not correct. First, this practice may yield an infeasible solution. Second, it would not guarantee selection of the optimal solution. These problems with using rounding require that an ILP cannot consider only the LP formulation as a solution starting point, resulting in (often dramatic) increases in the feasible decision space. Because of the decision space explosion caused by implementing integrality conditions, a complete enumeration of many realistic-sized problems will be required. These problems rapidly become unsolvable.

The ILP packages have several ways to deal with/avoid complete enumeration of realistic-sized problems. First, the packages' generally offer users the ability to set a tolerance that will stop the search when a solution is found within the defined (tolerance) percentage of the relaxed LP value. Second, packages generally allow users to specify a maximum runtime or maximum number of iterations for the search to find the "best"

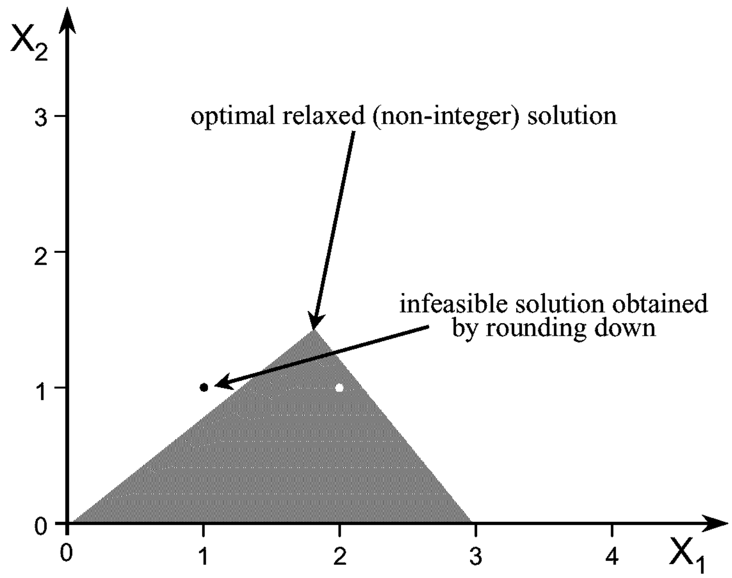


Figure 12. Infeasibility Caused by Rounding

solution up to the specified search duration. Third, packages may allow users to stop the search, without a full enumeration, if a maximum amount of time has passed without an improving solution. Fourth, since these problems are generally very simple mathematically although massive in the number of alternatives, packages may use the exponential growth in computer processing speed to rapidly manage complete enumerations. Fifth, packages may use the Simplex or Branch and Bound methods. Last, packages may use new methods to speed the search process such as advanced matrix factorization methods, improved Simplex that fully exploits scarcity in the solution space, steepest-edge pivoting strategies, or parallel searches (evaluating two or more solutions simultaneously) (Frontline, 2000b: 40).

All the disadvantages were considered before deciding to select this technique. Since force mix optimization problems are combinatorial and require integer decision variables, ILP provides a close fit to this research problem. Additionally, the integer decision variables are required to realistically model this problem.

Summary

This chapter reviewed the literature that provides a background for this research. The literature review found that the post cold war environment highlighted the critical need for a rapid method of creating lean and lethal force mixes that views DoD as a single organization. This rapid method would achieve greatest value if it could give a global vision to planners by eliminating the operations and logistics separate planning barriers. Since modeling and MP was very successfully used for many DoD problems, it warranted further study. Previous research provided valuable insights for mathematically representing the preference curve concept, the MRR suitability value, MRR deployment lift concept, and shortfalls of the other approaches. Within modeling and MP, LSILP's exact approach showed promise and had not yet been tried; therefore, further consideration was warranted. In Chapter III, a large-scale integer linear programming model is formulated and presented for the force mix campaign-planning problem.

III. Methodology

Introduction

This research uses a model, an abstract representation of a real-world problem, as its primary investigative instrument. Since the real world is too complex to model exactly, a model must make certain simplifying assumptions and only contain the level of detail necessary to accomplish its objectives. The objective of this research is to develop an efficient and realistic decision support model for force mix optimization. The model is intended to provide planners the best force mix within logistics constraints. This should help dissolve the barrier between operations, intelligence, and logistics planners, and provide a composite planner with a view of the whole decision space (indicating how the planner's asset choices directly affect or are affected by logistics constraints).

This chapter outlines the methodology used to develop the force mix optimization model. The chapter begins with the conceptual model formulation based on the ALP Pilot Problem. The second section describes the formulation of the ILP. This section also describes the development of the iterative process to solve along the preference curve to maximize the quantity of supportable missions restricted by improving suitability. The third section presents the performance evaluation measures, such as run-time and quality of solution. The fourth section presents the experimental design.

Model Formulation

The ALP Pilot Problem (Appendix B) provided the foundation for constructing this model. The MRR, the preference curve, and the Asset-Mission effectiveness value concepts were derived from this paper.

A systems approach was used to create the Mission Asset Resource Mix Optimization Tool (MARMOT) as a large-scale integer linear program model. The MARMOT Processing Model is illustrated in Figure 13. Using the systems approach, the problem was broken-down and evaluated according to the systems approach components: input, process, and output. The inputs consist of the Task Preference Matrix (TPM), the MRR Task Suitability Matrix (TSM), the MRR Weight Matrix, Deployment Lift

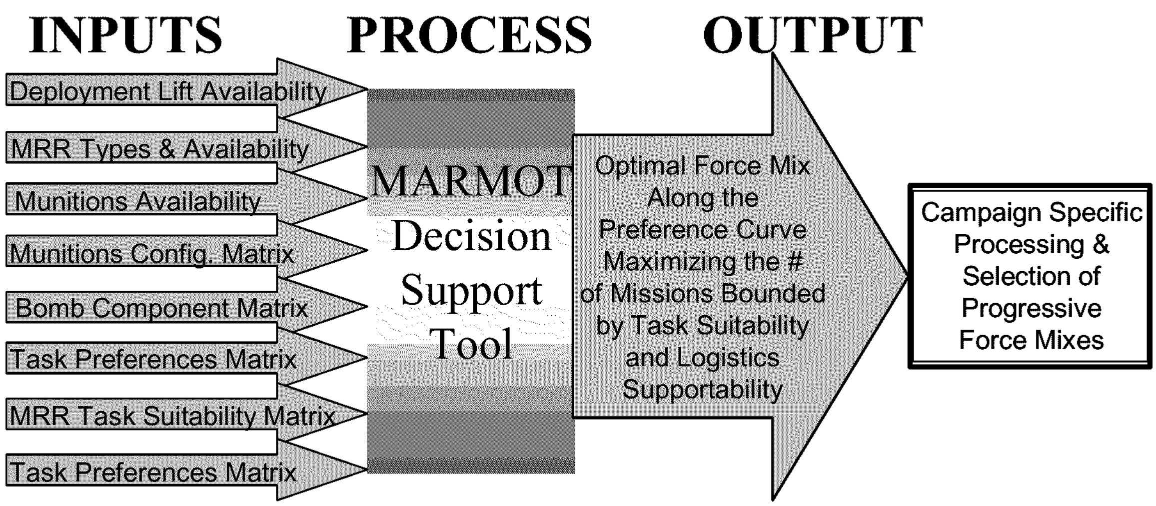


Figure 13. MARMOT Processing Model

Availability, MRR Types and Availability (different asset/munitions configurations and quantity of mission quantities), Munitions Configuration Matrix (MCM), Bomb Component Matrix (BCM), and Munitions and Component Availability. The process step, conducted in the MARMOT, converts the inputs into usable output. These inputs will be described in the next several paragraphs. The MARMOT Decision Support Tool uses the methodology extracted from the literature discussed in Chapter II and is described in the next section. The output should provide the war planner an optimal force mix that balances logistics constraints and operational requirements. The model solution presents the planners the best mix of MRRs, the highest number of supportable tasks, the total number of supportable daily missions, the amount of deployment lift consumed, the number of assets to deploy, and the amount of bomb components and munitions to deploy.

The TPM is based on the ALP Pilot Problem's Sortie/Mission Mix Preference Inflection Points (over Resource Levels) concept (Table 4). In this formulation, the integer values for each incremental mission are included in matrix form, eliminating the need for inflection points and linear relationship assumptions. The TPM is designed to represent the commander's preference for certain missions over others during an operation when a limited number of total missions are available. For example, during the initial deployment when few sorties are available and gaining air superiority is a priority, a CINC may prefer a large proportion of the total missions to be AA (20 of the 40 sorties available), a fairly large proportion to be SEAD (10 of 40) and INT (10 of 40), and no CAS. As more total missions become available and/or the campaign transitions to ground support priorities, the CINC may prefer a larger portion of CAS (600 of 1000 sorties), a fairly large number of INT (250 of 1,000), and fairly low number of AA (100 of 1,000) and SEAD (50 of 1,000). This is illustrated in Figure 14. The TPM then represents the relative priority of tasks. The TPM helps to rapidly identify force mixes by matching mission preferences to tasks. Appendix C presents an example of the notional TPM used for this model.

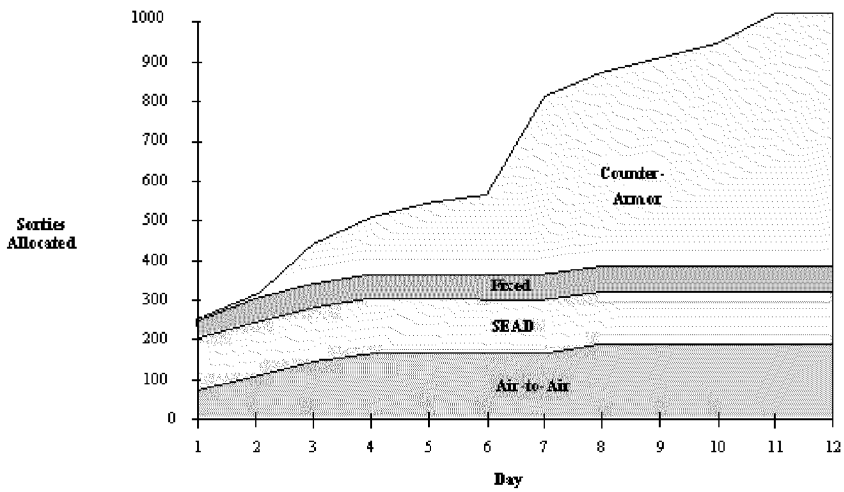


Figure 14. Task Preference Over Mission Levels (RAND, 1997:18)

The TSM is based on the Pilot Problem's Asset-Mission Task Preference Matrix (A-M TPM) concept (Table 3). This research uses a slightly different definition of the effectiveness values. In the Pilot Problem Formulation, the A-M TPM value of .5 implies that two sorties are required to achieve the same battlefield result of a single sortie of the best asset for that mission. In this formulation, the TSM value is a "mission suitability" value assigned by military targeting experts. The TSM represents the relative effectiveness of an MRR performing a task. As mentioned in Chapters I and II, it is assumed that the effectiveness is composed of the intrinsic suitability (specific asset and munitions capability against a specific target) and the extrinsic suitability.

Notional, yet realistic, data was used for this suitability matrix to keep this research unclassified. The real values would be based on the expert assessments of war planners and intelligence people, who would assign effectiveness values for specific assets and munitions combinations (MRRs) against specific target types (tasks). The TSM therefore establishes the mission-weighted preferences between assets (MRRs) and missions and is listed in Appendix D.

The MRR concept, as a basic deployment unit of measure, represents a departure from classical logistics approaches. Therefore, logistics support requirements (including deployment cargo weights) have not been formulated or calculated using this unit of measure. It was therefore necessary to construct approximate values for the support of an MRR based on existing (non-MRR) data. Notional values for the MRR weight matrix are used due to this lack of data. Determining the relationship between MRRs and lift requirements is beyond the scope of this research; however, this must be accomplished prior to real-world application of the model. Since actual values could be easily substituted into the model when they are determined, the use of notional values poses no risk to internal model validity. An extension to this model to calculate MRR logistics support requirements could then be used to calculate the exact deployment weight needed and create the TPFDD.

The MCM lists the specific feasible munitions configurations for each MRR, and is used to calculate the total number of munitions required. For example, each A10-1 MRR requires 2 AIM-9s, 2 AGM-65s, and 4 GBU-12s according to its configuration. The complete MCM is listed in Appendix E. The A10 and F16 configurations are based on the standard configurations from the FAS website listed in the Bibliography section. The others are based on notional data.

Several munitions are built from shared components. Examples include gravity bomb assemblies such as MK82s, MK84s, or GBU-24s. Since the model is being built to determine concept feasibility and is not for full implementation, a representative sample (not a complete enumeration) of second tier resources (components) were programmed in the model. The BCM lists these components in Appendix F. The components are referred to as second tier resources because their demand/usage is a derived demand from the munitions end item (i.e. MK-82) in the MCM determined by the number of MRR using that munition selected. For example, both the MK-82 and the GBU-12 munition use 500-pound bomb bodies as components. The need for 500-pound bomb bodies is therefore derived from the number of MK-82s and GBU-12s needed. The number of MK-82s and GBU-12s is derived from the number of A10-1s, A10-9s, A10-11s, F16-3s, F16-11s, F18s, F19s, B1s, B2s, and B3s selected. The BCM is based on the component list from the FAS and Eglin websites listed in the Bibliography section.

The Deployment Lift Availability, MRR Types and Availability, and Munitions and Component Availability are all hard constraints based on the resources available to the planner using this model and situation specific circumstances. The planners will input these constraints based on their specific circumstances. To evaluate the validity of this model, notional values were selected and the values were fixed while evaluating the performance measures during the experimental design.

ILP Formulation

The model was constructed in three phases: basic model with single point on preference curve mapped, expanded model (basic plus munitions configuration and bomb components) with single point on preference curve mapped, and expanded model mapped along the entire preference curve.

Basic Model.

The first phase of MARMOT involved creating a basic model with a single preference curve point. A small-scale model was built to determine the feasibility of pursuing this research methodology. This modeling process began by using the five steps identified by Ragsdale: understand the problem, identify the decision variables, state the objective function as a linear combination of the decision variables, state the constraints as linear combinations of the decision variables, and identify any upper or lower bounds on the decision variables (Ragsdale, 2001: 20-21).

Understand the Problem. A thorough knowledge of the problem is essential to accurately model its relevant characteristics as simplistically as possible. This is important, since the "best model is the simplest model that accurately reflects the relevant characteristics or essence of the problem being studied" (Ragsdale, 2001: 9). The problem in this research is fairly easy to understand: how many of each MRR asset types to assign to each required task to maximize the total suitability value, while using no more than the amount of all resources available, and meeting the mission distributions according to the preference curve.

Identify the Decision Variables. In this problem, the fundamental decision campaign planners face is how many of each MRR asset type to assign to satisfy the CINC's task preferences based on total mission levels. For this problem, $X_{i,j}$ represents the number of MRRs using asset type *i* to complete task *j* at daily total mission level "m." For example, $X_{1,1}$ represents the number of A10-1 munitions configured aircraft assigned to AA sorties and $X_{30,10}$ represents the number of B5 munitions configured aircraft assigned to PB sorties. To test the applicability and robustness of this model to the real world, 300 decision variables were created (30 MRRs by 10 task types). This is substantively expanded over the 15 decision variables used in David Wakefield's study.

Objective Function. Since the decision variables have been determined, the next step is to create an objective function that is a linear combination of these 300 decision variables. Assuming that a daily task is satisfied by exactly one MRR and that no interactions exist between different MRR asset types, then the suitability (*S*) for all MRRs assigned to all tasks is defined by

$$S = \sum C_{i,j} X_{i,j} \qquad (3.1)$$

where $C_{i,j}$ is the suitability of MRR *i* for Task *j* (according to the TSM in Appendix D) and $X_{i,j}$ is the number of MRRs *i* assigned to task *j*.

Constraints. As mention in Chapter II, the decision variables usually have some limitations on the values that they can assume. These limitations must be identified and stated as a linear combination of the decision variables. For the basic model, there are four constraints: integrality condition, deployment weight availability, MRR asset availability, and preference curve requirements.

Often, real-world situations may require all or some of the variables in the formulation to assume integer values since fractional values are not feasible/possible. In this model, the decision to assign an MRR asset to a task/mission is based on a yes/no decision and is not realistically divisible. This mandates that all the MRRs must assume an integer value, such as

$$X_{i,j} = \text{Integer} \qquad (3.2)$$

Since the logistics footprint limits the amount of assets that can be deployed, the total weight consumed by the MRR selection must be within lift availability. This portion of the model is perhaps the weakest, because it includes the least realistic assumptions. In reality, a deploying force requires lift for both re-usable (and shared among MRRs) equipment, and consumed materials (Goddard, 2001: 15). This model assumes a single, independent weight requirement for each MRR. This represents a serious limitation of the model formulation, but a true determination of lift requirement is unavailable and outside the scope of this thesis.

Assuming that no interactions exist between different MRR weight values and an MRR weight across tasks is the same, the total weight (W) for all MRRs is defined by

$$W = \sum_{i=1}^{n}(P_i \sum_{j=1}^{m} X_{i,j}) \tag{3.3}$$

where

i = MRR 1,…,n,
j = task 1,…,m,
P_i = weight of MRR i (according to the weight vector in Appendix G),
n = final MRR asset type,
m = last task,
$X_{i,j}$ = number of MRRs i for task j.

Since the total weight (W) must be less than or equal to the available deployment weight (AW) the formula is written:

$$W \leq AW \tag{3.4}$$

Instead of using the quantity of assets available to represent limited resources and to constrain MRR selection, the number of missions an asset could perform daily was selected. The number of asset-missions was the selected unit of measure since it more accurately represented the MRR concept. One MRR equals one task/mission and any individual asset has the ability to perform several missions in a day. In addition, logistics consumption (for example, fuel, munitions, or spares) was assumed to have a stronger correlation to the number of sorties than the number of aircraft. Ultimately, a given resource can be classified as either "reusable" or "consumable." Reusable assets (like aircraft) can be used for many missions. Consumable resources are used once and discarded. The maximum number of missions per day (MMPD) for a specific asset class (A) is defined by

$$MMPD_A = Q_A(turn\ rate_A) \tag{3.5}$$

where Q_A is the quantity of asset class A. The term asset class refers to identical base assets with different configurations. For example, the A10 asset class represent the A10-1 to A10-11 MRR set since all A10 configurations use the same airframe, and the total of all A10 configurations at any time must be less than or equal to the total number of A10s available. Given that asset class A has Z configurations, the assigned daily missions (ADM) is formulated as

$$ADM_A = \sum_{i=1}^{Z_A} \sum_{j=1}^{m} X_{i,j} \tag{3.6}$$

where

i = MRR 1,…,n,
j = task 1,…,m,
m = last task,
n = final MRR asset type,
$X_{i,j}$ = number of MRRs i assigned to task j,
Z = final configuration,
Z_A = asset class A with 1,…,Z configurations.

Since a mission can only be performed if the configured asset is available, the $MMPD_A$ is an upper limit for the ADM_A and the constraint can be mathematically defined by

$$ADM_A \leq MMPD_A \tag{3.7}$$

This constraint ensures that no MRR can be allocated a number of tasks during the time period that exceeds asset availability. The number of constraints resulting from equation

(3.7) is equal to the number of MRR asset classes. This is important to the programmer who must ensure this constraint is appropriately modeled.

Since the decision maker precisely defines task/mission preference based on the total mission level, L (according to the preference curve) the relationship between the preference curve tasks (PC) and MRR assigned tasks ($X_{i,j}$) must be based on an equality constraint. The constraint that all tasks $j=1,..,m$ must be satisfied at a particular total mission level is:

$$PC_{j,L} = \sum_{i=1}^{n} X_{i,j} \tag{3.8}$$

where

- i = MRR 1,…,n,
- j = task 1,…,m,
- L = mission level or total number of missions,
- n = final MRR asset type,
- $X_{i,j}$ = number of MRRs i assigned to task j.

The number of constraints resulting from equation (3.8) is equal to the number of m tasks. These constraints ensure that the number of sorties for Task i is equal to the commander's preference curve at that level of total missions.

Bounds. Frequently, decision variables may have upper or lower bounds that should be viewed as additional constraints. In this model, there are simple lower bounds of zero on the $X_{i,j}$ decision variables since it is impossible to generate a negative number of missions. These are referred to as non-negativity constraints, stated as:

$$X_{i,j} \geq 0 \tag{3.9}$$

The basic model was built in Excel using these mathematical formulations and Frontline System's Large Scale Linear Programming Solver Add-in. Since initial runs indicated that an optimal solution could be identified in a few seconds, the second phase was started.

Expanded Model with a Single Preference Curve Point.

The second phase involved creating an expanded model with a single preference curve point. The term *expanded* is used to represent the basic model plus the munitions and bomb component constraints. For this model to be of use to campaign planners, the computer run time for finding the optimal solution must be short (less than a minute).

Having a combat asset (people or vehicles) without munitions has no combat value, and munitions are a limited resource. Therefore, munitions make up an additional important set of constraints. Bomb components are the basic unit that makes up different bomb configurations and these components are limited. A shortage of the components prevents bomb availability which precludes combat asset availability.

The munitions end item (a fully configured munition; AIM-9, MK-82, etc.) requirement, U_t for all munitions end items $t = 1,..e$, is determined by

$$U_t = \sum_{i=1}^{n} MCM_i \sum_{j=1}^{m} X_{i,j} \tag{3.10}$$

where

i = MRR 1,…,n,
j = task 1,…,m,
m = last task,
n = final MRR asset type,
$X_{i,j}$ = number of MRRs i assigned to task j,
MCM_i = number of munitions end items for MRR over all tasks (according to Appendix E).

The U_t must be less than the munitions available, $MUNS_t$,

$$U_t \leq MUNS_t \tag{3.11}$$

Since the munitions configurations are specifically defined, any munitions shortages within the configuration eliminates the configuration as a feasible alternative. For example, the A10-1 MRR requires 2 AIM-9s, 2 AGM-65, and 4 GBU-12s. If there are only 3 GBU-12s or a shortfall in any of the other munitions, no further A10-1s can be selected. If the reduced munitions configurations are a feasible planning alternative, they must be included as a separate configuration.

The bomb component usage BC_d for all bomb components $d = 1,..f$, is determined by

$$BC_d = \sum_{t=1}^{e} BCM_{d,t} U_t \qquad (3.12)$$

where

d = bomb component $1,\ldots,f$,
e = last munitions end item,
f = last bomb component,
t = munitions end items $1,\ldots,e$,
$BCM_{d,t}$ = number of bomb component items for each bomb end item (according to the BCM in Appendix F),
U_t = number of munitions end items consumed.

The BC_d must be less than the component available, CA_d,

$$BC_d \leq CA_d \qquad (3.13)$$

Since the bomb configurations are specifically defined, any component shortages within the bomb configuration eliminates the configuration as a feasible alternative. For example, shortages in the M904E2 fuse eliminates any further selection of MK-82s or MK-84s and any combat assets using the MK-82s or MK-84s munitions.

The expanded model, using the same Excel and Frontline System's Large Scale Linear Programming Solver Add-in, incorporated these additional mathematical formulations. Since an optimal solution could be identified in less than five seconds, the third phase was started.

Expanded Model Along a Preference Curve.

In the final phase, a model was created to iteratively follow a preference curve to not only determine the optimal force mix at a specific point (total mission level), but also determine the optimal number of total supportable missions. This iterative process was intended to determine the maximum number of supportable missions, while fulfilling the CINC's exact task preference at that total mission level, and optimizing the selection of MRRs to complete these tasks. The preference curve matching process is illustrated in Figure 15. The visual basic code used to achieve this process in the model, the Preference Curve Mapping Macro (PCMM), is listed in Appendix H.

The first step in the process involved determining if the current iteration has a total suitability value (S) greater than or equal to the previous iteration (the initial default value is zero). An important choice exists at this point. There is a fundamental question as to which has greater value: the suitability of an asset set assigned to a number of missions, or the total number of missions achievable for a given asset set. Given a finite amount of lift and resources, a tradeoff exists between the ability to prosecute fewer missions with better aircraft, or more missions with less capable aircraft. This tradeoff, which occurs when less capable assets require less lift and/or less constrained resources than better-suited assets, has not been suitably explored in previous literature and remains a critical area for future research.

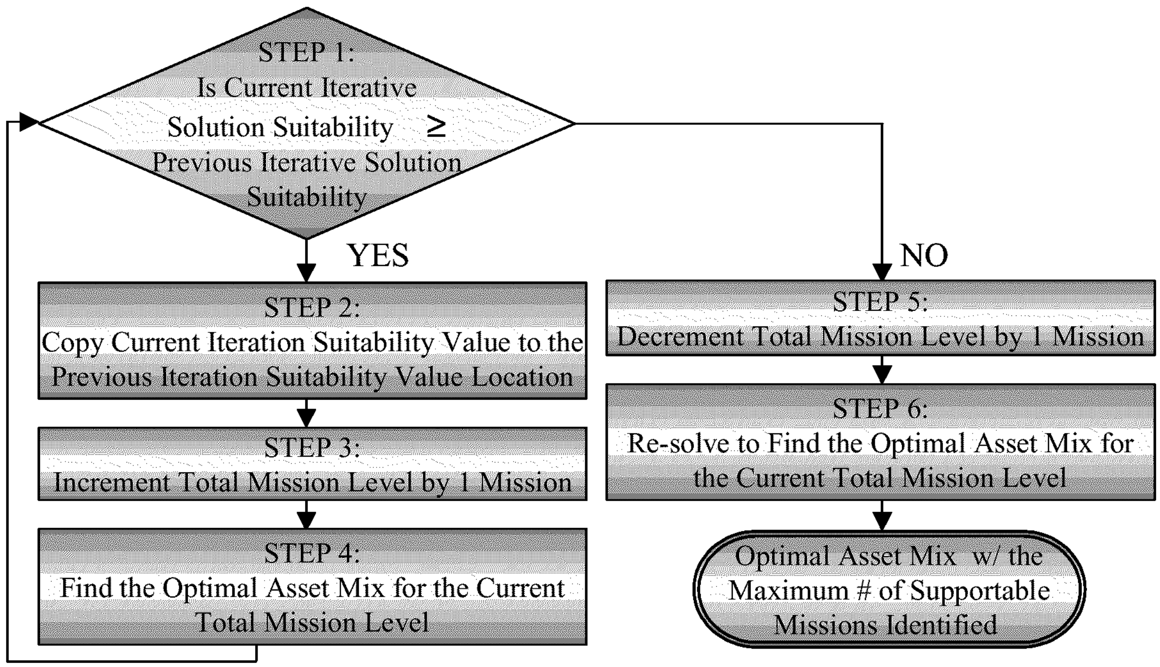


Figure 15. Preference Curve Mapping Process

Total suitability (*S*), instead of the number of missions, was selected as the primary objective. Setting the primary objective as maximizing the number of missions would subjugate the importance of task suitability to selecting assets with the lowest deployment lift requirements, since lift availability was a major factor limiting the number of MRR missions. For example, since F16-11s have the lowest weight requirements, they would be substituted for better-suited aircraft to increase the number of missions while keeping within the available weight. If the optimal aircraft mix was selected based on an asset capability and the maximum amount of weight was consumed, it was likely that in many circumstances (including the present model formulation) the only way to increase the number of missions would be to substitute better suited assets for lighter, less-effective assets such as sending F16-11s (instead of bombers) for precision bombing tasks. Therefore, it was decided to prefer asset set suitability over total number of missions for this current model.

In the decision diamond of Figure 15, the greater than or equal to condition has two parts. The greater than portion of the condition forced the model to explore increasing mission levels along the preference curve as long as the total asset set suitability value improved. Since a mix with the same total suitability value and higher quantity of total missions (mission level) was assumed to be preferred over the lower mission level mix, the equal to part of the condition (identified in the decision diamond of Figure 15) was incorporated into the macro. These two sub-conditions helped find the maximum number of supportable missions with assets performing tasks they are well suited for. If the current solution suitability value was greater than or equal to the previous iterative solution value, the macro completed the following tasks; otherwise, the macro skipped to the “decrement total mission level by 1 mission” task in Figure 15 and completes the task that follow.

The second step copied the value of the current total suitability value for an optimal asset mix at the most recent total mission level to the previous suitability value location. This provided the comparison between the next solution result and this result for step one.

The third step increased the Total Mission Level by one. This caused the task preference quantities to be updated from the TPM to exactly match the CINC’s task preference at that Total Mission Level. This step was the backbone of the iterative process to map along the preference curve.

The fourth step ran the solver add-on to generate an optimal total suitability value for the current Total Mission Level. This sequence of steps ensured that the new suitability value (from step four) could be compared to the previous suitability value (from step two) in step one.

Eventually, the first four steps caused the model to find a solution that was one mission beyond the best solution. At that point the suitability value began to decrease. The fifth step then forced the model to move back to the optimal mission quantity. This was initiated by decrementing the number of total missions by one. This caused the task preference quantities to be updated from the TPM to exactly match the CINC's task preference at that Total Mission Level.

After step five was completed, the final task of the macro began. The final step identified the optimal mix of assets at the greatest number of supportable missions. This step also initiated the generation of an optimal total suitability value for the current Total Mission Level. After solver completed its optimization, the process was done.

Performance Measures

The total time to find the "best" solution, total runtime, is calculated as finish time minus the start time.

$$total\ runtime = finish1 - start1 \tag{3.14}$$

The total runtime calculation was automated by including it in the Preference Curve Mapping Macro (see Appendix H). As seen in Appendix H, the macro recorded the time when the optimal solution was found (finish1) and subtracted it from the time recorded at the beginning of the search process (start1) to calculate total runtime (cell W62 in the worksheet).

Programming the runtime computation into the macro was designed to promote consistency through all iteration runs and between the different scenarios/trials; however, variance due to the computer's processing of other programs, the allocation/priority of computer random access memory, the computer's use of resources to keep the computer from overheating, etc. caused minor differences in the time to run the same scenario. These variances hide the "true" runtime; therefore, the screensaver was disabled and all other computer processes/tasks (the ones that were able to be ended using the Window's Task Manager) were ended to remedy this effect and focus the runtime.

Automating the runtime was designed to promote consistency by

- Ensuring measurement validity (measures what it is supposed to) and reliability (consistently measures results accurately)
- Eliminating variance in human response rate to measure time and between time keepers (internal consistency and interrater reliability)
- Ensuring time was measured from the time the computer began computing until the last computation exactly the same way at the same start and stop points for all trials and scenarios

Model Validation

The validation process focused on three areas: general performance, preference curve compliance, and macro functionality.

The general performance validation ensured that the model would select the correct assets (highest suitability values) and properly consume resources based on MRR usage and the munitions and bomb component matrices. This was accomplished by solving small problems with known solutions. For example, the model was tested against five sorties with AA only preference and unconstrained resources. Since all five sorties were performed by F18s (the best selection) and resource consumption was as expected, the model performance was validated.

Preference curve compliance was examined in two areas: monotonic assurance and exact mapping. First, it was necessary to ensure that all curves were monotonic. This was accomplished by creating a cell in Excel that compared the sum of each task at each level to the mission level total. If the value of the two numbers were equal, “TRUE” was recorded in the cell, otherwise “FALSE.” No “FALSE” values were present, so the curves were monotonic. Second, the look-up function that maps the task quantities to the preference curves was tested. Different values were put into the total mission level block and the associated task values were compared to the values in the matrix worksheet. Since the values matched exactly, the look-up function ensured that the tasks at each mission level were the same as the task mix on the preference curve.

The functionality of the macro was evaluated using two approaches. First, the solutions for curve 1 using the single and five-batch approach were calculated using manual runs and the results were compared to the macro solutions. This demonstrated that the macro would find the best solution since both approaches yielded the same MRR mix, mission totals, and suitability values. Second, comparing the macro-calculated time versus a stopwatch time on five occasions tested the macro’s ability to accurately track model runtime. Since the times were within 100th of a second and any difference was likely due to human reaction time delay, the macro’s runtime calculation was validated.

Experimental Design

Computational Environment.

The Frontline Systems Premium Solver Platform and Large-Scale LP Solver add-ons for use with Microsoft Excel 2000 were selected to execute this Large ILP methodology. They require limited programming skills, they are capable of handling problems with up to 65,000 variables and 65,000 constraints, the Excel program is user friendly and compatible with other presentation software such as PowerPoint, and they guarantee selection of the best/optimal solution. The MARMOT was executed on a Dell Inspiron 7500 equipped with a Pentium III 650MHz processor, 128MEG RAM, 18.6GB hard drive, and using *Windows 2000 Professional*.

Factors and Levels.

To measure model sensitivity, robustness, and generlizability, 16 different factor-level combinations were evaluated. These factor-level combinations were represented by four different preference curves, two different solution space search methods, and two different deployment lift batching approaches. The preference curves, search methods and batching approach factors and levels will be discussed in greater detail in the following sub-sections.

<u>Preference Curves.</u>

In order to test the performance of the complete model, a series of preference curves were developed. This series of preference curves were created to represent the different extreme possibilities (high, medium, or no task differences of a single line segment and multiple line segments with various slopes) that the preference curves could assume. Only the extreme points were evaluated since it was assumed that this isolated each factor for study, eliminated interactions for evaluation, and efficiently/effectively represented all possible complicating factors. Four different curves were selected in order to represent the effects of multiple line segments and represent the high, medium, and no task differences of a single line segment.

The first preference curve was constructed to represent changing proportions of task preference over different total mission levels without assuming linearity of the entire curve. Creating a preference "curve" with multiple line segments of various lengths and slopes up to 1,600 total missions represented this characteristic. Sixteen hundred was selected as the ending total mission level for all four curves, since it was large enough to allow the model to search the curves without exhausting the entire search space and it standardized the curves at the same ending mission level. For preference curve one, the ending value for each task was 130, 210, 165, 155, 105, 140, 175, 125, 175, and 220. These values were selected for the following reasons:

1. They provide for a "medium task difference" since some task preferences were higher than others without a strong bias
2. They summed to 1,600
3. They represented a relatively wide range of proportions

The second preference curve was constructed to determine if differential preference was a factor that affected runtime or solution accuracy. This was modeled by assigning a higher preference for half the tasks and lower preference for the other half, while maintaining linearity of the entire curve and relatively constant proportions of tasks. Richard Antoine's Excel linear approximation program, which used an algorithm developed by Air Force Institute of Technology researchers, was used to create a very flat slope (high task difference) with a single linear segment (Antoine, 2001: 45-48). The Excel program generated monotonic integer preference curve values to 1,600 total missions with 270, 270, 270, 270, 270, 50, 50, 50, 50, and 50 ending task values. These values were selected for the following reasons:

1. They represented high task differences since half the values were 5.4 times larger than other half
2. They summed to 1,600
3. They represented high task concentration with only two different proportions

The third preference curve was constructed to represent relatively constant proportions of tasks with minor difference in preference between tasks while assuming linearity of the entire curve. The same Excel program used for preference curve two was used to creating a medium slope (medium task differences) with a single linear segment. The same ending values for preference curve 1 were used for this curve. These values were selected for the following reasons:

1. They represented medium task differences, since some task preferences were higher than others without a strong bias
2. They summed to 1,600
3. They represented a diverse number of proportions
4. They model linearity/constant proportions for comparison against curve one

The fourth preference curve was constructed to represent relatively constant proportions of task with no difference in preference between tasks while assuming linearity of the entire curve. The same Excel program used for preference curve two and three was used to creating a high slope (no task differences) with a single linear segment.

All ten tasks were programmed to have 160 as their ending values. These values were selected for the following reasons:

1. They represented no task differences (indifferent task preferences)
2. They summed to 1,600
3. They represented equal proportions

Search Methods.

The second major investigative factor was the method used to sequence the optimization algorithm. Two methods were created to navigate the solution space: step-wise and jump-wise. The step-wise method involved incrementing the total missions/resource level by one each time as it searched the solution space for the greatest suitability value. The search algorithm stopped when the suitability value began to decrease, and the algorithm then returned to the previously optimal value. The jump-wise method incremented the total missions by ten as it searched the solution space. The search algorithm stopped when the suitability value of the current iteration was lower than the value of the previous iteration. The method then decremented the total missions by twenty (two iterations) and proceeded step-wise. It was found that by only retracing to the previous jump, it was possible to miss the optimal solution.

The jump-wise search was performed at the end points of the jump interval. The optimal point would most likely be at the top of a curve that decreased to the right and left of that point. When the placement of the interval caused values explored on the right side of the optimal point to be greater than the values explored on the left side, a false (local) optima could be picked up by the search. Therefore, a retrograde by two jumps was required to ensure optimality. After the algorithm decremented twenty missions, the solution space search concluded by using the same approach as the step-wise method.

Batching Approaches.

The third major investigative factor was the method used to aggregate deployment weight for the lift constraint. Two batching approaches were used to model the weight constraint: a single batch of 18,000 short tons, and five batches of 3,600 short tons (totaling 18,000 short-tons). The single batch would be analogous to developing a plan for aggregating a week's worth of lift into a single movement, while the five batch approach would represent the development of a plan to optimize the force mix constrained by 5 smaller daily (disaggregate) movements.

Summary of Factors and Levels.

Model performance was evaluated by using the two search methods with the two batching approaches against the four preference curves. Since the search methods, batching approaches, and preference curves represent various extremes, the 16 different factor-level combinations (2 methods times 2 approaches times 4 curves) should provide an accurate measure of model sensitivity, robustness, and generalizability.

Experiments.

Two primary experiments were conducted: first, the model searched the preference curve until the maximum number of missions that yield the maximum suitability value was reached, and second, the model searched the preference curve to a maximum of 775 missions.

The first experiment was designed to evaluate quality of solution, convergence on known optima, and attainment of the exact same MRR mix and suitability value for step-wise and jump-wise test within each batch quantity and preference curve combination. To illustrate, the step-wise results for the single batch and preference curve 1 combination would be compared to the jump-wise results for the same combination.

The second experiment was designed to evaluate runtime performance to determine if the jump-wise approach would be faster than the step-wise approach regardless of the batching method or nature of the preference curve.

In the first experiment, ten repetitions of each of the 16 factor-level combinations (160 total tests) were conducted while allowing the model to search the preference curve until the maximum number of missions with improving total suitability is reached. After each test was run, the iteration runtime, number of missions, and suitability values were recorded for analysis in Chapter IV.

In the second experiment, the same procedures were followed; however, only five iterations of each test were conducted and the preference curve search space was restricted to 775 total missions. The 775-mission value was selected since it was the lowest number of tasks attained by all tests in experiment one, and it ensured that all tests ended at the same point. Ending the model at the same point eliminated potential bias caused by evaluating different sized solution spaces and allowed a valid comparison of runtimes between the different tests.

Summary

This chapter addressed the third investigative question (Can the effectiveness and efficiency of the selected methodology be tested?) by describing the experimental design and testing methodology. The ALP Pilot Problem described in Chapter II provided the foundation to mathematically represent the force mix problem to be modeled as a LSILP. The performance measures were used in conjunction with the experimental design to demonstrate how the different methodologies could be tested. Chapter IV answers the remaining investigative questions by evaluating the results of the tests.

IV. Results

Introduction

The previous chapter presented the experimental design used to assess the MARMOT against two experiments with 16 different tests each. This chapter uses the results of these tests to evaluate the performance of the MARMOT.

Experiment One Output and Analysis

The complete results of the first experiment, 160 iterations, are presented in Appendix K and the summary results are presented in Table 3. To ensure that the search methods (step-wise and jump-wise) were comparable, the final MRR mixes and suitability values for preference curve one, the first iteration of the single batch, were matched to ensure exactly the same results. The same was done for the 5 batch approach for preference curve 1, iteration 1, using the step-wise and jump-wise search methods. The MRR mix is presented in Appendix M. Both tests demonstrated that each method yielded exactly the same MRR mixes and suitability values.

Table 3. Summary of Experiment One Results

	Single Batch								5 Batches							
	Step-Wise				Jump-Wise				Step-Wise				Jump-Wise			
	Mean	Std Dev	Tasks	Value	Mean	Std Dev.	Tasks	Value	Mean	Std Dev	Tasks	Value	Mean	Std Dev.	Tasks	Value
Preference Curve 1	1020.092	2.079	779	683.58	126.620	0.633	779	683.58	851.515	3.031	790	683.21	188.857	3.295	790	683.21
Preference Curve 2	1386.524	3.908	789	710.88	251.426	0.974	789	710.88	941.877	0.993	790	694.78	238.432	0.582	790	694.78
Preference Curve 3	938.479	1.833	775	656.19	113.053	0.124	775	656.19	841.518	1.618	782	651.3	172.149	0.668	782	651.3
Preference Curve 4	823.512	0.842	777	669.22	108.271	0.318	777	669.22	842.553	2.895	784	664.51	176.388	2.978	784	664.51

The quality of solution was evaluated by determining if each iteration within each of the 16 tests produced the same suitability value and number of tasks, and that the step-wise suitability value and mission quantities in each batch quantity and preference curve combination matched the jump-wise results in the same combination. As seen in Appendix K, all iterations yielded the same suitability values and quantity of missions within each test. As seen in Table 3, all step-wise suitability values and mission quantities for each batch quantity and preference curve combination matched the jump-wise results in the same combination. For example, the step-wise suitability value of 683.58 and mission quantity of 779 for the single batch and preference curve 1 combination equaled the jump-wise results for the single batch and preference curve 1 combination. The quality of solution and comparability evaluations demonstrated that both approaches resulted in the same answer for the war planner regardless of the underlying nature of the preference curve or batching method. Since both methods result in the same solution, the method that yields the solution the fastest is the best method, as long as both methods converge on the same known optima. Since manually solving at each iterative point for the single and five batch approach for preference curve one had the same solutions as discussed in the comparability sub-section and Appendix M, it is assumed that the two methods would always converge on the same optimal solution.

Statistical Analysis

The complete results of the second experiment, 80 iterations, are presented in Appendix L and the summary results are presented in Table 4. The number of missions (tasks) were kept constant in this experiment to provide an accurate means of judging the effects of the preference curve design, batching method, and search methods on runtime.

These tests will help determine the robustness of the model and search methods and if one of the search methods is preferred.

Table 4. Summary of Experiment Two Results

	Single Batch								5 Batches							
	Step-Wise				Jump-Wise				Step-Wise				Jump-Wise			
	Mean	Std Dev.	Tasks	Value	Mean	Std Dev.	Tasks	Value	Mean	Std Dev.	Tasks	Value	Mean	Std Dev.	Tasks	Value
Preference Curve 1	1014.452	1.518	775	682.1	154.037	0.300	775	682.1	842.987	1.490	775	675.11	178.402	0.372	775	675.11
Preference Curve 2	1324.859	1.203	775	707.52	628.241	0.812	775	707.52	872.664	2.112	775	685.41	208.375	0.295	775	685.41
Preference Curve 3	949.387	0.426	775	656.19	126.100	0.647	775	656.19	835.885	1.590	775	647.66	179.277	0.426	775	647.66
Preference Curve 4	830.283	1.627	775	668.27	112.528	0.126	775	668.27	832.665	0.948	775	660.48	189.191	0.251	775	660.48

The sample data in Appendix L suggests that the population distributions for the 16 tests are not normal and require non-parametric methods for statistical comparison (see Figures 16-19). The assumption of normalcy, necessary for parametric tests, would not be appropriate since several of the sample distributions appear to be bi-modal, exponential, or uniform.

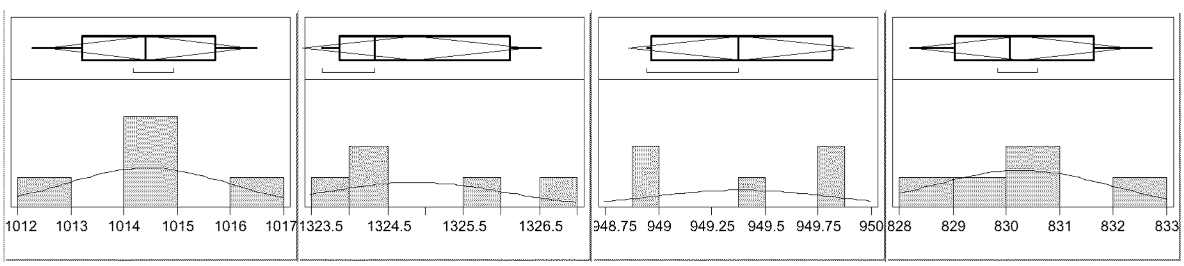


Figure 16. Single Batch Step-Wise Runtime Distributions for Curves 1 – 4

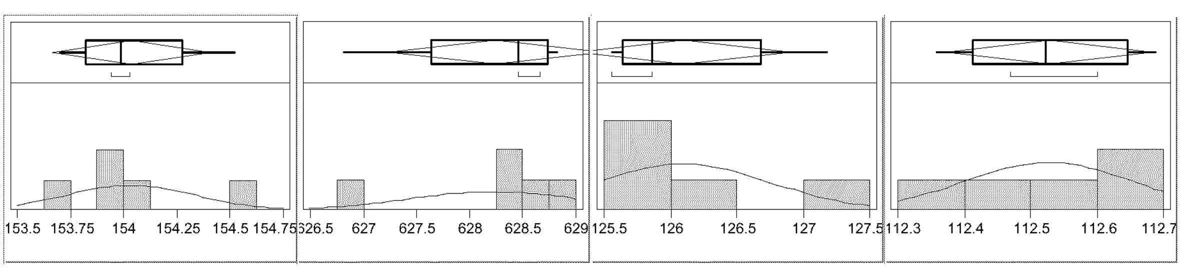


Figure 17. Single Batch Jump-Wise Runtime Distributions for Curves 1 – 4

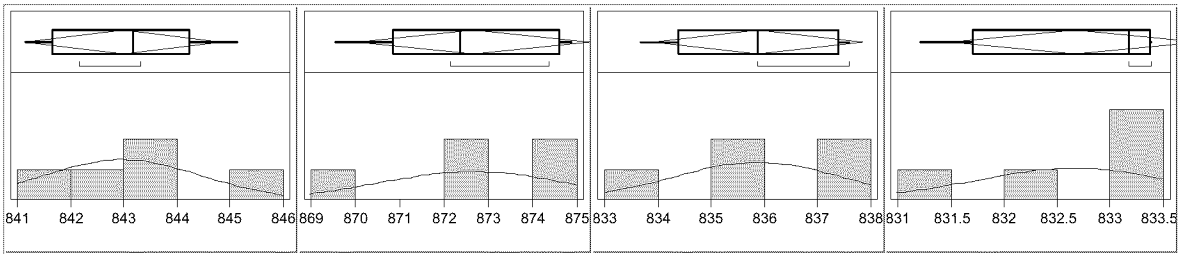


Figure 18. Five Batch Step-Wise Runtime Distributions for Curves 1 – 4

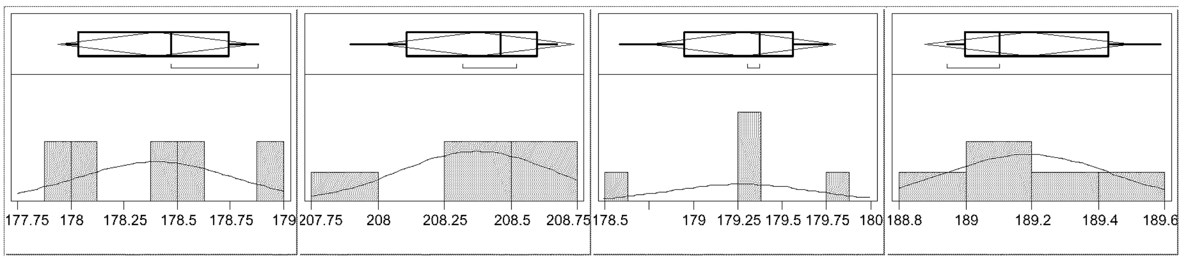


Figure 19. Five Batch Jump-Wise Runtime Distributions for Curves 1 - 4

The Wilcoxon Signed-Rank Test for Paired Observations and the Friedman Multiple-Block Rank Comparison Test were the nonparametric statistical techniques used to evaluate the runtimes. The Wilcoxon test assumes that the differences between the pairs are continuous and symmetric, which is significantly less restrictive than the assumption of normalcy (Devore, 2000). The Friedman test assumes that the results in each block are mutually independent (they do not influence the results in other blocks) and the observation in each block may be ranked according to some criterion of interest (Conover, 1980: 296-299). These assumptions are satisfied, so the following hypotheses are tested:

H_0: The step-wise and jump-wise sampled populations have identical probability distributions.

H_A: The step-wise probability distribution is shifted to the right of (slower than) the jump-wise probability distribution.

Since the number of ranks, *n*, is greater than 20 and the Sign-Rank table only provided critical values for level α test when n ≤ 20, the Wilcoxon Large-Sample Approximation test was used (Devore, 2000: 655). In Appendix N, the test was accomplished using Excel 2000 to calculate the test statistic

$$Z = \frac{S_+ - n(n+1)/4}{\sqrt{n(n+1)(2n+1)/24}}. \tag{4.1}$$

According to Devore, this test statistic can be justifiably applied to the standard normal (*Z)* distribution (Devore, 2000: 656). At the 0.01 significance level for the upper-tailed test, the observed value of 5.5109 indicates that there is sufficient evidence to reject H_0.

This result demonstrated that the jump-wise method is significantly faster than the step-wise approach regardless of the underlying nature of the preference curve and batching approach. This result was also recognized by inspection, since all jump-wise time were several times faster than the corresponding step-wise version. By visual inspection, it also appears that the jump-times for curves 1, 3, and 4 are within a minute of each other while curve 2 tends to be several minutes slower. This will be statistically analyzed below:

H_0: Each ranking of the random variables within a curve is equally likely.

H_A: At least one of the curves tends to require longer processing time than at least one other curve.

The Friedman test in Appendix O was used to evaluate these four curves. The test was accomplished using Excel 2000 to calculate the test statistic

$$T_2 = \frac{(b-1)[B_2 - bk(k+1)^2/4]}{A_2 - B_2} \tag{4.2}$$

where

$$A_2 = \frac{bk(k+1)(2k+1)}{6} \tag{4.3}$$

and

$$B_2 = \frac{1}{b}\sum_{j=1}^{k} R_j^2 \ . \tag{4.4}$$

At the 0.01 significance level, the null was rejected since T_2 exceeded the 1-α quantile of the F distribution from Table A26 with $k1 = k-1$ and $k2 = (b-1)(k-1)$ (Conover, 1980: 300). A curve is considered different if

$$\left|R_j - R_i\right| > t_{1-\alpha/2}\left[\frac{2b(A_2 - B_2)}{(b-1)(k-1)}\right]^{\frac{1}{2}} \tag{4.5}$$

is satisfied where $t_{1-\alpha/2}$ is the 1-α/2 quantile of the t distribution from Table A25 with $(b-1)(k-1)$ degrees of freedom (Conover, 1980: 300). The results, illustrated in Appendix O, indicate that Curve 2 is significantly different from all the other curves, Curve 1 is significantly different from Curve 4 and 2, and Curve 3 is not significantly different from Curves 4 or 1. Although Curve 1 and Curve 4 statistically differ, the jump-wise difference of 2/3 of a minute, at most, in reality likely has no practical difference. Curve 2's jump-wise 1/2 a minute to less than nine minute difference may also have no practical difference for war planners who are accustomed to waiting hours to days for less detailed information.

The final hypothesis was designed to test the models against batching methods:

H_0: Each ranking of the random variables within a batch method is equally likely.

H_A: At least one of the batch methods tends to require longer processing time than the other.

The Friedman test in Appendix P was used to evaluate these two batching methods. The test was accomplished using Excel 2000 to calculate the test statistic

$$T_2 = \frac{(b-1)[B_2 - bk(k+1)^2/4]}{A_2 - B_2} \tag{4.6}$$

where

$$A_2 = \frac{bk(k+1)(2k+1)}{6} \tag{4.7}$$

and

$$B_2 = \frac{1}{b}\sum_{j=1}^{k} R_j^2 \,. \tag{4.8}$$

At the 0.01 significance level, the null was not rejected since T_2 did not exceed the 1-α quantile of the F distribution from Table A26 with *k*1 = *k*-1 and *k*2 = (*b*-1)(*k*-1) (Conover, 1980: 300). The previous test and this test demonstrated the robustness of the model by showing that batching and the nature of the preference curve did not affect the quality of solution or the ability of the model to rapidly find an optimal solution.

Summary

This chapter answered the fourth research question: what are the results of this test (runtime, quality of solution, do you get an answer, does it converge on known optima)? The results revealed that MARMOT was robust and yielded quality solutions regardless of the underlying preference curve or batching method. The results also demonstrated that the jump-wise approach was superior in all tests. The tests did reveal that there were statistical difference in runtimes for curves, particularly curves with large differences in tasks (flat slope); however, these time differences were less than nine minutes which would be insignificant in reality. This chapter analyzed the test results and Chapter V presents the conclusions, limitations, recommendations, and areas for future research.

V. Conclusion

Introduction

Chapter I and II discussed the motivation of this research by stating that the Defense Advanced Research Projects Agency (DARPA) seeks to use information technology to transport campaign planning into the 21st century with near real-time logistics information and the ability to develop and compare multiple deployment plans. Campaign planners will be able to develop and compare multiple scenarios, compare competing sets of combat assets based upon their designed suitability and theater specific issues, and respond to crises with greater effectiveness and efficiency than previously possible.

The Mission-Resource Value Assessment Tool, the front-end DARPA component, is intended to reduce the deployment timeline and footprint by selecting the force mix that maximizes the combined designed suitability and campaign specific issues value within logistics constraints. The primary goal of this research was to determine if there was a method to accurately represent these conflicting goals and find optimal force mixes within a reasonable amount of time. To accomplish this goal, five research questions were answered:

1. What is the underlying nature and structure of the problem being studied?
    a. What are we trying to maximize?
    b. What are our constraints?
2. What are the solution methodologies that best fit this problem structure?
    a. What are the key characteristics (to solution type) of the problem?

   b. What are the matching solution types?

3. Can the effectiveness and efficiency of the selected methodology be tested?

4. What are the results of this test (runtime, quality of solution, do you get an answer, does it converge on known optima)?

5. What test would be performed and what inferences could/should be drawn from the test results?

These questions were answered in three phases. The first phase included a literature review of the problem structure and the modeling field. The problem structure was reviewed to determine the interactions within the problem, the problem constraints, and what should be maximized. The review identified the need for total visibility between operations and logistics for planning, faster planning to cope with asymmetric warfare, and joint deployment planning to deal with limited resources and budgets. This answered question one. The modeling field was reviewed to determine the most appropriate modeling approach with which to model this problem. The review focused on metaheuristics, mathematical, linear, integer, and large-scale programming to determine the best methodology to fit the problem structure. This answered question two.

The second phase included data collection. The MRR suitability and deployment lift consumption were based on notional values from the ALP Pilot problem since actual mission specific resource capabilities were classified.

The third phase included creation of the Mission Asset Resource Mix Optimization Tool (MARMOT) by incorporating David Wakefield's formulation of the problem and the ALP Pilot Problem concepts into the integer linear programming formulation. The ALP Pilot Problem concepts include the Mission Ready Resource as the basic combat asset/munitions building block of suitability and lift consumption, the Task Preference Curve/Matrix, force mix feasibility defined by decision space and problem constraints, and force mix desirability base on task preference/suitability. MARMOT was built to evaluate preference curves with different underlying characteristics and different batching methods. One curve was created to represent multiple line segments of various lengths. The other three curves were created to represent a single line segment with high, medium, and no differences between ending task quantities. This phase answered question 3.

The fourth phase evaluated the alternative methodologies using two experiments with 16 tests each based on five objectives. The first objective was to determine if each iteration of the same test yields the same solution. The second objective was to determine if the jump-wise approach yields the same solution as the step-wise approach within a batch quantity and preference curve combination. The third objective was to determine if each test finds a known optima. The fourth objective was to determine if the jump-wise approach found a solution faster. The fifth objective was to determine if the model was robust. This phase answered the remaining questions and found that the MARMOT using the jump-wise approach was appropriate for rapidly finding the optimal force mix within logistics constraints. The results also show MARMOT's robustness. The exponential growth of computer processing speed and memory provide additional support to the viability of this approach to provide combat planners near-real time deployment solutions.

Conclusions and Significance of Research

The results of this research showed that integer linear programming was a useful and promising approach to investigating the force mix selection problem. The significance of this research is identified in the results that:

1. Proved the concept of using integer linear programming to optimize force mix selection for realistically sized problems in a reasonable amount of time (within minutes).

2. Verified that force mix tailoring could be automated using a systematic and objective approach rather than subjective educated guesses.

3. Showed that a force mix model could incorporate intelligence, operations, and logistics visibility/planning into a single system providing a global/system-wide view rather than a local stove-piped view.

4. Elevated the logistics footprint from an afterthought to an integral part of the force mix decision.

5. Demonstrated the MARMOT could provide real-time combat asset and logistics tradeoff analysis.

6. Demonstrated the methodology could maximize rapid combat capability deployment by optimizing the entire system rather than only sub-systems.

These results verified the viability of using a MARMOT-style model to generate optimal force mixes that maximizes combat capability by balancing combat sustainment and capability.

Limitations

First, the assumption of a single, independent weight requirement for each MRR represents a serious weakness of the model. If this simplifying assumption is not valid, the weight consumption for the force mix could be overstated or understated. Overstating the weight would cause the planner to deploy less combat capability due to the inefficient use of lift. Conversely, understating the weight would cause the war planner to select an unsupportable force mix. In either case, inaccurately accounting of lift consumption would likely result in the selection of a sub-optimal force mix. This means that the model does not accurately represent the problem and should not be used. Additionally, if the model could not assume a single, independent value for weight, the use of specific business rules could cause the solution to be intractable with runtime increasing exponential due to the combinatorial nature of this problem. This would make this approach inappropriate for this type of problem.

Second, the assumption that extrinsic and intrinsic suitability can be captured in a single number is a limitation. If a single number were not possible, then the model would not properly represent the tradeoff between these two factors. This could cause the selection of assets that are not allowed in an area or the overlooking of assets with a total package that is superior, such as low logistics footprint due to shared components with a host nation. Additionally, this could cause manual intervention, which would decrease the ability to rapidly and objectively build deployment packages.

Third, the model assumes a linear and continuous relationship between the decision variables and constraints and between the decision variables and the objective function. If both relationships are not linear and continuous, linear programming cannot be used. Thus, a different methodology would be required to solve the problem. Continuing to use this approach when the assumptions are not satisfied could result in ineffective and inefficient deployment of combat power at the least or campaign failure and the needless loss of life at the other extreme.

Fourth, the model assumes that no assets are destroyed. This could overstate the number of assets available, which overstates the number of missions and tasks able to be supported. Overstating the missions assets can perform could cause the selection of force mixes that cannot meet planning objectives or cause campaign failure. The probability of asset destruction could be accounted for by adjusting each asset's turn rates to reflect the probability of accretion.

Fifth, a limitation centers on the data retrievability assumption. The model assumes that the composite suitability values can be calculated on an MRR basis and deployment lift can be calculated as a function of the MRR values. Since the MRR concept is a new concept within DoD, a cultural and infrastructure shift would be required to re-align DoD to take advantage of this concept. Even though this concept may revolutionize and dramatically improve combat planning, people's resistance and adversity to change could prevent its acceptance and implementation. This resistance could affect the quality of data gathered, which would cause the model to select less than optimal or disastrous force mixes. Additionally, if the data cannot be gathered on an MRR basis, then the data and model would not represent the real-world problem. This could cause the model select an infeasible or dangerous force mix.

Sixth, the assumption that the sum of the suitability values is more important than the sum of tasks is another limitation. This could cause the selection of a force mix that does not meet the needs of the combatant commander. Since building plans that meet the needs of the combatant commander is the primary objective of force mix planning, the model would pursue the wrong objective and select a less than optimal solution.

The final limitation deals with problem size and the number of resources modeled. Although the model was built for a realistically sized problem, adding more munitions, bomb components, or asset types may cause runtime to exceed an acceptable limit. This would cause the model to generate solutions too late or not at all. Since adding the munitions and bomb components to the basic model had very little effect on runtime, the models robustness may cause this limitation to be irrelevant.

Although MARMOT currently can solve real-world problems, it is populated with notional data which requires actual values before being operationally applied. This means that the model cannot be used until the data is gathered.

Recommendations and Future Research

Since the success of the MARMOT cannot be achieved until real-world data is used, research must be done to collect lift business rules or single value. This research must investigate if a single, independent weight value accurately represents the MRR weight consumption or what the business rules are for each support item on an MRR basis, and what that value (either single value or business rule value) is for each MRR asset type. This would also include evaluating logistics requirements for any interactions, continuity, and linearity between MRRs.

In addition to evaluating the weight consumption, research must determine if the intrinsic and extrinsic factors can be represented as a single, independent composite value, if there are any interactions between the different MRR that effect the values, how to combine the two major factors, and what the composite values are. This would also include evaluating the composite value for continuity and linearity. If the two factors cannot be incorporated into a single factor, research must be done to determine how to include these two factors separately into the force mix determination model.

Since the MARMOT model is reliant on the linear and continuity assumption of the objective function and the constraints, additional research is needed to validate this assumption. The objective function and weight constraint are discussed in the first two recommendations, but the other constraints, such as munitions end items and components, must be evaluated. Are all relevant constraints identified? Are the consumption values accurate? Do the consumption values change? When?

Since the MARMOT model assumed that maximizing the suitability value was the goal, does this accurately represent the combatant commander's needs? Is maximizing the number of missions a better measure? Is there some combination of these two objectives that is preferred? How can these two objectives be balanced? This area has not been suitably explored in previous research and remains a critical area of exploration.

The modular nature of the MARMOT model and formulas make it compatible for implementation into the Mission-Resource Value Assessment Tool and DARPA project, and consistent with previous research in the area. However, small-scale demonstrations and further testing are needed to evaluate integratability and appropriateness.

Further research is also necessary to determine the most effective and efficient jump value. The value was set at 10, but is there a value which is universally better? Is there a value that is better for preference curves of a specific nature (high, medium, or low slope)?

Additional research is also needed to determine the underlying nature of preference curves and the best or easiest way to obtain preference curve values. Are the curve tasks monotonic? Are the curves linear? Are the curves piece wise linear segments? Are the curves constant during a combat phase? Can the data on the entire curve be enumeratively gathered from the combatant commander? Does gathering inflection points for the curve accurately represent the combatant commander's true needs?

Finally, research is necessary to determine how sensitive the model is to more platform and munitions/bomb alternatives and the impact of the simplifying assumptions on the solution fitness. Does runtime increase exponentially with each addition? What increases runtime more adding platforms, munitions, bomb components, or some combination? How much of an increase in platforms, munitions, components, or some combination are needed before the model becomes unsolvable in a reasonable amount of time?

Summary

The MARMOT model developed in this research provides a method to rapidly evaluate and select force mixes based on task/target suitability, mission preference and capability, and resource/lift availability. It forms the foundation of the M-R VAT and DARPA architecture and provides a means for instantaneously computing force mix value, global planning visibility, and optimizing the deployment of combat capability.

This research and the MARMOT decision support tool presents several advancements over earlier works in the war planning and deployment field and provides a stepping-stone for future research. The most important advancement includes the quantification of force mix selection so that consistent, expert decisions will be made regardless of a planner's experience level.

Appendix A. Acronyms and Definitions

AA	Air-to-Air--missions to clear the area and patrol incase of a launch of enemy aircraft
AEF	Aerospace Expeditionary Force--a composite organization of aerospace capabilities from which a tailored Aerospace Expeditionary Task Force, composed of AEWs, AEGs, and AESs, is created to provide forces o meet theater CINC requirements. An AEF is not a discrete warfighting unit (AFI 10-400).
AEW	Aerospace Expeditionary Wing--a wing or wing slice assigned or attached to an Aerospace Expeditionary Task Force or an in-place NAF by MAJCOM G-series orders. Normally, the Aerospace Expeditionary Task Force or in-place NAF commander also exercises OPCON of AEWs. An AEW is composed of the wing command element and some groups. The AEW commander reports to a COMAFFOR (AFI 10-400).
AFIT	Air Force Institute of Technology, Wright-Patterson, Ohio
ALP	Advanced Logistics Project-- a joint Defense Advanced Research Projects Agency and Defense Logistics Agency research project, was developed to speed up the sourcing and tailoring of existing TPFDDs.
APS	Advanced Planning System
BCM	Bomb Component Matrix--a listing of the number of shared components used to make a specific bomb.
CAS	Close Air Support--missions flown against hostile targets in close proximity to friendly forces. Bombers have effectively been used in the CAS role. However, missions have to be carefully planned to avoid fratricide and loss of the bomber to enemy fire.
CINC	Commander in Charge/Combatant Commander--the commander in charge of all operations within a unified theater.
CONOPS	Concept of Operations
CONUS	Continental United States--the 48 contiguous states.
DARPA	Defense Advanced Research Projects Agency

DLA	Defense Logistics Agency
FAM	Functional Area Managers--the person at the MAJCOM that is responsible for ensuring the accuracy of the data in UTCs.
ILP	Integer Linear Programming--LP with some or all of the decision variables assuming integer values.
JMPS	Joint Mission Planning System
LP	Linear Programming--MP with linear objective function and constraints.
LSILP	Large-Scale Integer Linear Programming--an ILP that has many decision variables, constraints, or both and is so complex that normal ILP procedures are inadequate to deal with the problem.
MAJCOM	Major Command
MARMOT	Mission Asset Resource Mix Optimization Tool
MCM	Munitions Configuration Matrix--a matrix detailing the munitions load for a specific aircraft configuration.
MP	Mathematical Programming--using mathematics that describe how systems work or will work to find the optimal use of limited resources.
MRR	Mission Ready Resource--composed of a resource type and its logistics requirements, i.e. aircraft, pilot, fuel, munitions, support equipment and personnel, etc., that has a certain task suitability.
M-R VAT	Mission-Resource Value Assessment Tool--a tool, using MRRs, is designed to rapidly identify alternative force mixes by matching mission preferences to tasks, tasks to resources, and resources to logistics requirements.
MTW	Major Theater Wars
PFPS	Portable Fighting Planning System
PCMM	Preference Curve Mapping Macro--Macro in Appendix H

SEAD	Suppression of Enemy Air Defense--missions to destroy or disable radar-guided, surface-to-air missile sites and anti-aircraft artillery.
SSC	Small Scale Contingency--unpredictable challenges such as conflict against rogue nations with weapons of mass destruction, terrorism, ethnic tension, etc., that are not part of the MTW preplanning concept that require rapid response of limited forces.
TPFDD	Time-Phased Force Deployment Data--Joint Operation Planning and Execution System data base portion of an operation plan with time-phased force data, non-unit-related cargo and personnel data, and movement data for the operational plan, including: (a) in-place units, (b) units to be deployed to support the operation plan with a priority indicating the desired sequence for their arrival at the port of debarkation, (c) routing of forces to be deployed, (d) movement data associated with deploying forces, (e) estimates of non-unit related cargo and personnel movements to be conducted concurrently with the deployment of forces, and (f) estimate of transportation requirements that must be fulfilled by common-user lift resources as well as those requirements that can be fulfilled by assigned or attached transportation resources (JP 1-02).
TPM	Task Preference Matrix--theater commander task preference based on number of total missions available.
TSM	Task Suitability Matrix--relative effectiveness of an MRR performing a specific task/mission and establishes the mission weighted preferences between MRRs and missions.
US	United States
USAF	United States Air Force
USTRANSCOM	US Transportation Command, Scott AFB, Illinois
UTC	Unit Type Code--modular, scalable capability to provide combat and support operations across the spectrum of operations.

Appendix B. ALP Pilot Problem (Swartz, 1999)

ALP Pilot Problem and Derivation of Mathematical Model

I. Measuring the Relative Utility of Resource Sets

Preferences or Relative Utility of Assets for Various Tasks

Assume two basic aircraft: a bomber, with two potential configurations; B-A and B-B; and a fighter, with three potential configurations; F-A, F-B, and F-C. The aircraft are assigned to a unit tasked with providing four missions or tasks: SEAD, AA, CAS, and INT. Certain configurations of the different aircraft can provide higher or lower levels of effectiveness in applying a single sortie to the various missions according to the Asset-Mission Task Preference Matrix (A-M TPM). The "favorite" or best asset to apply to a particular mission is assigned a value of unity, and the effectiveness of alternative assets is weighted accordingly. A value of .5 implies that two sorties of the aircraft would be able to achieve equivalent results to a single sortie of the best aircraft for the mission.

Table 5: A-M TPM

	SEAD	AA	CAS	INT
B-A	.5	0	.2	1
B-B	.8	0	.6	.8
F-A	.3	1	.2	0
F-B	.5	.1	1	.5
F-C	1	.1	.3	.2

In essence, the "scorer" of the relative effectiveness of these resources is assessing how many additional sorties would have to be flown by a non-preferred asset in order to achieve the same mission outcome as the preferred asset. In some cases, there is no way an asset could create the same battlefield effect as another; in those instances, the non-preferred asset is assigned a value of 0.

This matrix establishes our sortie-weighted preferences between assets for missions. These preferences can be set aside for the moment; the A-M TPM will become critical in a later section.

Preferences or Relative Utility of Various Tasks for the Campaign

Over the course of a campaign, commanders have preferences for certain missions over others; these preferences represent the relative value of one mission type over another. For example, out of an arbitrary number of total missions required for a given day, a commander may prefer that most of them be AA (60%), a fairly large number be SEAD (30%), no CAS, and a small amount (10%) for INT. At this level of total activity, for this given day of the campaign, these relative ratios represent the relative weight of importance or priority between the mission types. Of course, the relative priority and numbers of missions would be different for different days of the campaign. They would also probably be different for different levels of total activity (resource availability) possible. Let's examine the latter case further, then address the former.

Utility in Response to Resource Level. The relative balance or share will probably not remain constant over varying levels of total resource availability. At any given level of resources available, the actual relative balance or share of mission type out of total missions may or may not remain constant. For example, a certain mission type may be critical at a low level of total missions, but as resources increase, it's marginal value may taper and another mission type becomes paramount. For example, consider the following mission requirements based on the preference ratios just described:

SEAD: 20 AA: 40 CAS: 0 INT: 10

The relative ratios 2/4/0/1 between the numbers of missions by type for campaign operations can be used to provide an index of criticality or desirability; (preference or utility assumed to be weighted by the number of sorties specified). This relative utility assumes a certain level of resources available. These utility ratios may change as the level of resources increases or decreases. For example, as the total number of sorties possible gets larger and larger, the increase in the number of SEAD and AA sorties may taper (diminishing returns); while the value of some CAS or extra INT sorties may increase. Conversely, if the total number of sorties decreases, the INT sorties desired may decrease rapidly, and the commander may be more reluctant to reduce the SEAD missions than the AA missions. At varying levels of total resource availability, the relative marginal utility of one sortie or mission type over another may change. The inflection points of the diminishing/increasing utility curves can be captured using the expert knowledge of the campaign commanders and planners. The commander can be asked to determine relative utility given various levels of resourcing. For example, by identifying the preferred sortie/mission mix for the "nominal" case (above) and then providing alternative mixes representing points where the relative priorities change as resource levels go up or down, we could represent shifts in relative marginal utility or priority.

Table 6: Sortie/Mission Mix Preference Inflection Points (over Resource Levels)

	Crit	Nom	Hi
SEAD	5	20	30
AA	30	40	50
CAS	0	0	10
INT	0	10	25

In this table, we can see that for resource levels between 0 and 35 total sorties/day, commanders would prefer a sortie mix in the ration of 5/30/0/0. In other words, as resources flow into the theatre of operations, the commander would prefer 6 AA sorties for every 1 SEAD sortie- and isn't interested in any CAS or INT sorties at all. Between 35 and 70 total sorties, the relative utility can be represented by the ratio 15/10/0/10. Between 70 and 115 total sorties/day, our preference ratios would be 10/10/10/15. As no inflection point was identified above 115 total sorties/day, it could be assumed that the relative utilities above that level would be equal; with relative priority ratios of 1/1/1/1. These three inflection points could be associated with "Critical," "Nominal," and "High" levels of resources.

Utility in Response to Time. The relative balance or share (relative preference) of missions will probably not remain constant over time. Priorities will change as the campaign progresses. The time-phased preference changes can be represented by system states or phases at some arbitrary (but preferred) level of total missions. For example, assume that our campaign can be conducted in three phases: an air supremacy phase, a strategic bombing phase, and finally a close air support phase. Given a scenario (including enemy configuration), a commander could provide the following preferences for numbers of sorties at the "nominal" level of missions.

Table 7: Sortie/Mission Mix Preference Inflection Points (over Time)

	I	II	III
SEAD	20	20	10
AA	40	16	16
CAS	0	10	40
INT	10	30	10

Note that the actual time value associated with the phase (state) change is not indicated. It could be specified in advance when the state/phase changes are anticipated to occur. For example, state/Phase I type operations will begin on "Day 0" and the change to state/Phase II will occur on "Day 15," and Phase III operations are planned for "Day 30." Also, while this example indicates the presence of only three states or phases, a very large number of nuanced states could be defined. Of course, for each phase of the campaign, the commander would also be able to complete the specification of relative preferences or utility at various levels of resources as in the previous paragraphs.

Preferences for missions would therefore vary along two dimensions: the dimension of time or phases and the dimension of resource availability. The resulting three-dimensional matrix will indicate relative marginal utility of missions or tasks over both time and level of resources. This can be represented by the Mission-Resource-Time Task Preferences Matrix (M-R-T TPM). This matrix, when combined with the previously described Asset-Mission Task Preference Matrix (A-M TPM), can be used to construct a map of the relative "value" of different sets of assets in the theatre against the missions that are preferred during the campaign.

Table 8: M-R-T TPM

	I			II			III		
	Crit	Nom	Hi	Crit	Nom	Hi	Crit	Nom	Hi
SEAD	5	20	30	15	20	30	10	10	15
AA	30	40	50	5	16	20	5	16	20
CAS	0	0	10	0	10	10	30	40	60
INT	0	10	25	15	30	60	5	10	20

The M-R-T TPM represents the relative utility or preference the campaign commander has for the set of available missions, for the resources at his or her disposal, over various levels of resource availability and over the duration of the campaign. Each combination of sorties or tasks under any specified resourcing level and campaign phase represents a "preferred sortie set" or point in *n* dimensions where *n* is the number of sortie or task types. The preferred mix of sorties or tasks can be described as the set of points along the resource levels (with inflection points specified) within a given campaign phase. Starting from "0" resources, the preferred mix of sorties or tasks is the vector between inflection points as resource levels increase.

Calculating the Relative Value of Preferred (Benchmark) Asset Sets

Assume, for the purpose of discussion, that we only assign the "best" asset to each task or sortie as described by the A-M TPM. Assume further that the total set of assets increases or decreases in a linear fashion between the inflection points described in the

M-R-T TPM, within each time phase. We could then easily establish the rank order or relative value of each asset set according to its position along the line between points. For example, in Phase I we start with having no assets and seek to achieve sufficient assets to reach the "Critical" task or sortie set of 5-30-0-0. The best SEAD asset is the F-C, and the best AA asset is the F-A. Since we desire 6 AA sorties for every 1 SEAD sortie (and since we desire 0 CAS or INT sorties); and since we are interested in integer values only, a sample relative value ranking would involve the following asset sets (from least to most) shown in Table 9.

Table 9: Asset-Set Points Along the Sortie/Mission Preference Vector

B-A	B-B	F-A	F-B	F-C
0	0	0	0	0
0	0	1	0	0
0	0	2	0	0
0	0	3	0	0
0	0	4	0	0
0	0	5	0	0
0	0	6	0	0
0	0	6	0	1
0	0	7	0	1
0	0	8	0	1
0	0	9	0	1
0	0	10	0	1
. . .	. . .	. . .	. . .	. . .
0	0	30	0	5

For each increment, we are able to state confidently that the increment (asset set) is better than or preferable to the increment that preceded it; and is not as good or is not preferred to the one that follows it. We are unable, however, to ascertain by “how much” any increment is preferred over another. This issue will be addressed momentarily. Currently, the concept of rank preference is sufficiently useful to continue the analysis.

Using the tabular information (representing a vector through the asset set space), the position of any asset combination (point in the asset space) could be related to the previously developed value or preference vector. For example, the point (1, 0, 6, 0, 0) is not on the value vector- but the point (0, 0, 6, 0, 0) is. What then can we say about the relative value of (1, 0, 6, 0, 0) vs. (0, 0, 6, 0, 0)? Has the combatant commander gained anything by having an additional B-A sortie?

The positive argument says yes, the commander has obviously gained; he or she has an additional sortie above and beyond what was desired. The negative argument says no, the commander has not gained; this additional sortie is unwanted and therefore provides no value. Both arguments seem to have merit; but which is correct? Consider a key fact: the commander has already indicated preferences for sorties. While it may be true that the additional sortie does add value to the package, it would be difficult to claim that the value added is greater than the value represented by the next point along the vector. While it may be true that $(0, 0, 6, 0, 0) < (1, 0, 6, 0, 0)$, it is also true that $(1, 0, 6, 0, 0) < (0, 0, 6, 0, 1)$. If given the choice between an additional B-A sortie and an F-C sortie, the commander has already spoken. Indeed, the commander prefers an additional F-C sortie over any number of any combination of additional sorties. The value of any off-vector combination of sorties is greater than the smallest vector point that is completely satisfied by all elements of the off-vector combination; and less than the value of the next higher vector point. For the time being, we can ignore the incremental value of the off-vector asset sets and consider their value to be equal to the value of the lowest completely satisfied on-vector asset sets.

Calculating the Relative Value of Non-Preferred Asset Sets

In the previous discussion, we have assumed that only the preferred assets (relative A-M TPM value of "1.0") would be considered. We were then able to assume a "one for one" correlation between the individual sorties or tasks (from the commanders mission preferences described in the M-R-T TPM) and the assets assigned to perform those sorties. During that discussion, the terms "sortie set" and "asset set" were synonymous. If non-preferred assets are being used, the value of the sorties provided must be adjusted relative to what would be achieved using the preferred assets. We must "calibrate" between sorties and assets to account for the use of less capable assets. Of course, we already have a way to do this: the A-M TPM provides us with our correction factors.

The asset set represents the number of mission-ready assets available to be tasked against sorties on the specified day. The sortie preference sets represent the points along the commander's preference vector. Relative position (relative to the preferred sortie sets vectors) can be calculated by multiplying the respective A-M TPM preference weight by the number of assets available for any given day. This weighted asset set can then be compared to the "nominal" (or one-one previously described) sortie set vector, and assigned a relative position based on the minimum sortie set satisfied by the weighted asset set. An example may be instructive.

Assume we have available at our disposal 12 F-C assets. These assets can be used in any combination against SEAD, AA, CAS, and INT sorties in the weights of 1, 0.1, 0.3, and 0.2. By assigning all assets to a single mission, we could achieve 12 SEAD, 1.2 AA, 3.6 CAS, or 2.4 INT weighted sortie equivalents maximum. Therefore, we have 4 possible integer allocations of these 12 assets to these 4 missions using the "all or nothing" principle, and weighting for effectiveness/preference of the asset type against the mission type. The four preference weighted sortie sets for the allocation of these assets would be: (12, 0, 0, 0), (0, 1, 0, 0), (0, 0, 3, 0), or (0, 0, 0, 2). Truncation is used instead of rounding; as a partial asset is meaningless for the performance of a sortie on a given day. The relative value of these preference weighted, asset allocated sortie sets could then be assigned against the preferred sortie set vector as previously discussed.

Bounded Possibilities; the Relative Values of All Combinations of Sortie Sets

Following the procedures described above, it would now be possible to enumerate all possible combinations of asset sets, bounded by the total numbers of assets available within each type. These asset sets could then be converted into preference weighted sortie sets using the A-M TPM. The (asset) preference weighted sortie sets could be assigned relative value positions along the (mission) preference weighted sortie set vector described by the M-R-T TPM. At this point, many of the asset sets could share the same value positions. We have lost the ability to differentiate among asset sets sharing the same sortie set value; even though differences may exist. Also, we have no ability to assign absolute values (vice relative values) to any of the sortie sets or asset sets. The enumerated asset set-relative value pairs could be considered to describe a response surface. The Dependent (outcome) variable of the surface is the relative value. The

Independent Variables are the numbers of assets by each type; an asset feasible combination of assets (an asset set) represents an *m* dimensional point along the surface where *m* is the number of different asset types.

Table 10: Relative Values of Asset Set Points along the Mission Preference Vector

V	Sortie Mix (S, A, C, I)	Equivalent Asset Sets (B-A, B-B, F-A, F-B, F-C)
0	(0, 0, 0, 0)	(0, 0, 0, 0, 0)
1	(0, 1, 0, 0)	(0, 0, 1, 0, 0):(0, 0, 0, 10, 0):(0, 0, 0, 0, 10):(0, 0, 0, 9, 1):(0, 0, 0, 8, 2) . . . etc.
2	(0, 2, 0, 0)	(0, 0, 2, 0, 0):(0, 0, 1, 10, 0):(0, 0, 1, 0, 10):(0, 0, 1, 9, 1):(0, 0, 1, 8, 2) . . . etc.
3	(0, 3, 0, 0)	(0, 0, 3, 0, 0):(0, 0, 2, 10, 0) . . . etc.
4	(0, 4, 0, 0)	(0, 0, 4, 0, 0) . . . etc.
5	(0, 5, 0, 0)	(0, 0, 5, 0, 0) . . . etc.
6	(0, 6, 0, 0)	(0, 0, 6, 0, 0) . . . etc.
7	(1, 6, 0, 0)	(0, 0, 6, 0, 1):(0, 0, 5, 10, 1) . . . (0, 0, 0, 0, 61):(0, 2, 6, 0, 0) . . . etc.
8	(1, 7, 0, 0)	(0, 0, 7, 0, 1) . . . etc.
9	(1, 8, 0, 0)	(0, 0, 8, 0, 1) . . . etc.
10	(1, 9, 0, 0)	(0, 0, 9, 0, 1) . . . etc.
11	(1, 10, 0, 0)	(0, 0, 10, 0, 1) . . . etc.
12	(1, 11, 0, 0)	(0, 0, 11, 0, 1) . . . etc.
13	(1, 12, 0, 0)	(0, 0, 12, 0, 1) . . . etc.
14	(2, 12, 0, 0)	(0, 0, 12, 0, 2) . . . etc.
15	(2, 13, 0, 0)	(0, 0, 13, 0, 2) . . . etc.
16	(2, 14, 0, 0)	(0, 0, 14, 0, 2) . . . etc.
17	(2, 15, 0, 0)	(0, 0, 15, 0, 3) . . . etc.
. . .	. . .	. . .
Max	TBD	All assets of each type

It must be noted at this time that the response surface just described would change between phases of the campaign. As the commander's mission preference vectors change between phases, the relative value of any given asset set would change as well. An asset set that gives you the "best" answer for Phase I might not (probably will not) give you the "best" answer for either Phase II or Phase III. In order to achieve better results over the duration of the campaign, the conduct of any individual phase may need to be subordinated or sub-optimized. This issue will need to be addressed by whatever solution methodology is employed to choose among potential asset sets.

The problem is not only bound by the total numbers of assets available, but is constrained by the total amount of lift available. The initial delivery and subsequent support of an asset (used to produce a sortie) consumes materials. These support materials need to be brought into theater. Each item transported into the theater consumes lift along two dimensions: cube (volume) and tare (weight). Lift is made available in finite amounts over time; and the amounts of cube and tare available will limit the amount of material support that can be delivered. The nature of the bounds and how to restrict the feasible set according to the bounds will be discussed in the next section.

II. Bounding Utility: Converting Asset Sets to Lift Requirements

<u>The Nature of the Bounds: Finite Assets</u>

As was just mentioned, the region of all possible combinations of asset sets could be generated using the commander's preferences for sorties by type, matched against the applicability of assets to missions. Of course, these possibilities are bound by the assets available and mission-ready at the operating location, on the day required. The total availability of resources required to make up the mission ready assets, available at the operating location, will limit the missions that can be serviced by sorties. In turn, there are two considerations that will bound the amount of resources available at the operating location: the total number of resources available worldwide, and the amount and timing of lift required to transport them.

The total number of resources available worldwide (and this number may change over time as resources are either generated through production or consumed through attrition) forms a "pool" of potential resources that sets the absolute "ceiling" for **potential** resource availability. The amount of lift available over time, combined with what is already present at the operating location at any given moment in time, will dictate the **actual** availability. The characterization of resources by consumption type (renewable or consumable) and failure type (frequency and predictability) are important considerations in determining resource availability. If a resource is renewable (vice consumed during use), it places no load on the lift process aside from the initial movement. However, even otherwise renewable resources may fail and be "consumed." When a failure occurs, this creates a demand for a replacement which must be satisfied through the lift process. These distinctions between renewable and consumable resources, and between failure frequencies and types, will be explored further in our discussion of resource types.

In our example, we had two basic aircraft: the "B" and the "F." The "B" airframe can be combined with other resources (fuel, munitions, equipment, etc.) to create two basic configurations, the B-A and the B-B. Similarly, the F airframe can be configured into three distinct assets, the F-A, F-B, and F-C. If we have only "X" Fs available, then any feasible asset set combination must have the total number of F-As, F-Bs, and F-Cs strictly less than or equal to "X." This strict bound would apply to each asset, and to each of the constituent resources that make up the asset. An asset (mission-ready asset) may consist of several resources bound in this manner. The number of basic airframes would certainly bound the possibilities. Auxiliary equipment, fuel, and munitions may also bound specific classes of assets as well. All of these resources can be classified as either being renewable or consumable. Rules for the use and availability of renewable and consumable assets can be developed to describe the nature of the bounds each will place on the feasibility of the potential asset sets.

Renewable Resources. Renewable resources are those which could be used over and over again for many missions, while perhaps suffering some minor degradation or wearout. Perhaps more importantly, these are resources that can be characterized by failures that are either rare, or unpredictable, or both. Three types of logistics resources could be classified as "renewable:" Aircraft, Equipment, and Personnel. Each of these will be discussed in turn.

The total number of aircraft (the F airframe, for example) in use at any given time could not exceed the total number available. However, any individual F could be used repeatedly over time for any number of F-A, F-B, or F-C missions. Upon return from a sortie, the individual aircraft could be repaired, refueled, and reconfigured for any suitable task. In this sense, the aircraft as a resource would be renewable; i.e., it could be used over and over again during the course of the campaign.

There would, of course, be a time delay involved between the time the aircraft completed it's previous task and when it would be available for the next task. Also, there would be a number of additional resources that may be needed (borrowed or consumed) by the aircraft in order to restore it to a mission ready configuration. Items such as spares, ammunition, and fuel would need to be installed (consumed) on the aircraft prior to it's being available for another task. Items such as power units, bomb loaders, and toolboxes would need to used (borrowed) during the restoration actions.

Another type of renewable resource includes items of support equipment. Tools, test equipment, powered and nonpowered aerospace ground equipment are all resources that could be used over and over again in support of the campaign.

The last category of renewable resources includes people. Operations, maintenance and support skill sets would all be included in this category.

The rules governing the use of renewable resources would be generally straightforward: calculate how many items you would need to support a given level of tasks, and then account for failure and attrition. As long as you have a sufficient number of each item available to meet the activity level, you will be able to accomplish the tasks specified. For example, to fly X sorties of F-A and Y sorties of F-B, you would need to have X+Y F airframes on hand. You would (perhaps) need X+Y maintenance people of type "Aircraft Crew Chief" as well. The rule for a different skill set (e.g. "Life Support") may be totally different; 1 person needed for the first 20 aircraft assigned, then one additional specialist for every 5 more aircraft. There may be a type of powered ground support equipment used to start the aircraft engines. This "start cart" is used on each aircraft for just 5 minutes, and then moved to the next aircraft. The start cart may only by needed in a ratio of 1 for every four sorties. In this case, then, the "rule" for the resource would result in a calculation of (X+Y)/4; rounded up to the next highest integer value. Another example might involve a ground refueling hydrant. After the sorties are flown, the aircraft need to be recovered. Recovery involves refueling; which requires the availability of a ground refueling hydrant. Here, the requirements rule for hydrants may be a complicated step function: the first hydrant will handle up to 4 sorties, two hydrants can service up to six aircraft, and three hydrants are sufficient to refuel up to 16 sorties- with similar increments beyond that number. Overall, however, the business rules for determining the requirements for renewable resources involve a comparison of activity level to capability of the resource.

Consumable Resources. Consumable resources are those that could not be used over and over again for many missions. While they (as in the case of spares) may indeed last for more than one mission, repeated use will result in additional demand for more resources. Generally, consumables can be characterized by failures or demands that are either predictable, or common (frequent), or both. Three types of logistics resources could be classified as "consumable:" Fuel, Munitions, and Spares. Each of these will be discussed briefly in turn.

Fuel

Munitions

Spares

The rules governing the use of consumable resources would be similar to the rules for renewable resources in that they would be based on the activity level supported. However, the calculation of the requirement would (in many cases) be a straightforward summation of use, without complex

Resource-Activity Integration.

The fundamental question of lift for this problem is "what to move into the operating location and when does it need to be there?" This question has two fundamental components that must be reconciled: first, what resources are needed on any given day;

and second, what will already be on-hand for that day. The shortage of material between the two makes up what needs to be delivered on that day. The delivery schedule can then be used to identify sources of material, and schedule the lift of that material from origin to destination.

Consider the restoration of an F returning from an F-A task, which must be made available for an F-B task. In order to create a mission-ready F-B, a host of prerequisites must be satisfied. First, an F airframe must be borrowed.

Appendix C. Task Preference Matrix (TPM) Example

Number of Missions	Task/Mission Preferences									
	AA	*AB*	*AC*	*AD*	*AE*	*AF*	*AG*	*NB*	*SB*	*PB*
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
1	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
2	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
3	**3**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
4	**4**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
5	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
6	**6**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
7	**7**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
8	**8**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
9	**9**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
10	**10**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
11	**10**	**1**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
12	**10**	**2**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
13	**10**	**3**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
14	**10**	**4**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
15	**10**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
16	**11**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
17	**12**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
18	**13**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
19	**14**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
20	**15**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
21	**16**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
22	**17**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
23	**18**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
24	**19**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
25	**20**	**5**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
26	**20**	**6**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
27	**20**	**7**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
28	**20**	**8**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
29	**20**	**9**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
30	**20**	**10**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
31	**21**	**10**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**
32	**22**	**10**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**

Appendix D. Task Suitability Matrix (TSM)

Asset Type & Munitions Configuration	TASK SUITABILITY									
	AA	*AB*	*AC*	*AD*	*AE*	*AF*	*AG*	*NB*	*SB*	*PB*
A10-1	0.45	0.78	0.96	0.95	0.97	0.91	0.94	0.01	0.40	0.01
A10-2	0.52	0.87	0.90	0.92	0.93	0.92	0.88	0.01	0.80	0.02
A10-3	0.52	0.80	0.90	0.89	0.91	0.92	0.97	0.01	0.70	0.02
A10-4	0.51	0.87	0.91	0.86	0.89	0.93	0.85	0.01	0.01	0.20
A10-5	0.51	0.86	0.92	0.83	0.87	0.93	0.85	0.01	0.01	0.20
A10-6	0.51	0.84	0.93	0.98	0.85	0.94	0.85	0.01	0.01	0.20
A10-7	0.51	0.82	0.94	0.96	0.87	0.94	0.85	0.01	0.01	0.20
A10-8	0.51	0.80	0.95	0.95	0.87	0.95	0.85	0.01	0.01	0.20
A10-9	0.51	0.78	0.96	0.94	0.87	0.95	0.85	0.01	0.01	0.20
A10-10	0.51	0.85	0.97	0.93	0.85	0.96	0.85	0.01	0.01	0.20
A10-11	0.51	0.88	0.98	0.98	0.87	0.96	0.85	0.01	0.01	0.20
F16-1	0.80	0.85	0.90	0.63	0.61	0.53	0.79	0.01	0.01	0.01
F16-2	0.76	0.64	0.76	0.85	0.85	0.58	0.53	0.01	0.01	0.01
F16-3	0.67	0.79	0.78	0.66	0.74	0.85	0.90	0.01	0.05	0.01
F16-4	0.70	0.70	0.77	0.66	0.70	0.64	0.84	0.01	0.02	0.04
F16-5	0.71	0.86	0.63	0.69	0.69	0.65	0.72	0.01	0.10	0.10
F16-6	0.69	0.86	0.62	0.63	0.62	0.65	0.80	0.01	0.01	0.01
F16-7	0.68	0.84	0.78	0.57	0.80	0.69	0.85	0.01	0.20	0.01
F16-8	0.67	0.79	0.84	0.66	0.66	0.67	0.73	0.01	0.01	0.04
F16-9	0.68	0.75	0.79	0.54	0.71	0.70	0.80	0.01	0.05	0.01
F16-10	0.68	0.88	0.80	0.78	0.84	0.78	0.84	0.01	0.03	0.01
F16-11	0.69	0.89	0.88	0.78	0.76	0.52	0.70	0.01	0.02	0.05
F18	0.85	0.85	0.85	0.84	0.76	0.62	0.94	0.01	0.06	0.02
F19	0.63	0.60	0.77	0.90	0.54	0.75	0.72	0.01	0.09	0.10
F20	0.72	0.75	0.63	0.86	0.53	0.67	0.68	0.01	0.01	0.01
B1	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.50	0.40	0.80
B2	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.40	0.50	0.70
B3	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.80	0.60	0.60
B4	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.70	0.70	0.50
B5	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.60	0.80	0.40

Appendix E. Munitions Configuration Matrix (MCM)

	Missiles										Guided Bombs				Unguided Bombs			Cluster Bombs			Targeting Pods		
	Air Intercept			Air to Ground							Laser Guided		GPS Guided										
	AIM-7	AIM-9	AIM-120	AGM-65	AGM-84	AGM-86C	AGM-88	AGM-130	AGM-154	AGM-158	GBU-10	GBU-12	GBU-24 E/B	GBU-31	MK-66	MK-82	MK-84	CBU-87	CBU-89	CBU-97	AAS-35	ASQ-213	LANTIRN
A10-1	0	2	0	2	0	0	0	0	0	0	0	4	0	0	0	0	0	0	0	0	0	0	0
A10-2	0	2	0	4	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0
A10-3	0	2	0	2	0	0	0	0	0	0	0	0	0	0	14	0	0	0	0	0	0	0	0
A10-4	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0
A10-5	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0
A10-6	0	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0
A10-7	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0
A10-8	0	2	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0
A10-9	0	2	0	2	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0
A10-10	0	2	0	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
A10-11	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	12	0	0	0	0	0	0	0
F16-1	0	0	6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F16-2	0	2	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F16-3	0	2	2	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0	0	0	0	0
F16-4	0	2	2	0	0	0	0	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	1
F16-5	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0
F16-6	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0
F16-7	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	4	0	0	0	0	0
F16-8	0	2	2	0	0	0	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
F16-9	0	2	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
F16-10	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	0	2	0	0	0	0	0	0
F16-11	0	2	2	0	0	0	0	0	0	0	0	0	0	0	0	6	0	0	0	0	0	0	0
F18	2	0	2	0	2	1	1	0	0	1	0	1	0	1	0	0	0	1	1	1	0	0	0
F19	4	0	0	0	1	0	1	0	1	0	0	0	0	0	1	0	0	0	0	0	1	1	1
F20	2	0	2	0	1	1	0	2	0	0	0	0	1	1	0	0	0	0	0	0	2	2	1
B1	0	0	0	0	1	1	1	1	1	1	1	0	0	0	1	1	0	0	0	0	1	1	1
B2	0	0	0	0	2	2	2	2	2	1	1	1	1	1	0	0	0	1	1	1	0	0	0
B3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	0	1	1	1
B4	0	0	0	0	1	1	1	1	1	0	0	0	1	1	0	0	0	1	1	1	0	0	0
B5	0	0	0	2	2	2	2	2	2	1	1	0	1	1	0	0	0	1	1	1	0	0	0

Appendix F. Bomb Component Matrix (BCM)

	Missiles										Guided Bombs				Unguided Bombs		
	Air Intercept			Air to Ground							Laser Guided		GPS Guided				
	AIM-7	AIM-9	AIM-120	AGM-65	AGM-84	AGM-86C	AGM-88	AGM-130	AGM-154	AGM-158	GBU-10	GBU-12	GBU-24 E/B	GBU-31	MK-66	MK-82	MK-84
Constraints: Fuze Options																	
M904E2								0			0	0	0	0	0	1	1
M905								0			1	1	0	0	0	1	1
FMU-26A/B								0			0	0	0	0	0	1	1
FMU-26B/B								0			1	1	0	0	0	1	1
FMU-54/B,A/B								0			0	0	0	0	0	0	0
FMU-56 D/B								0			0	0	0	0	0	0	0
FMU-72/B								0			0	0	0	0	0	1	1
FMU-81/B								0			1	1	1	0	0	0	0
FMU-110/B								0			0	0	0	0	0	0	0
FMU-113/B								0			0	0	0	0	0	1	1
FMU-124A/B,B/B								1			0	0	0	0	0	0	0
FMU-139A/B								0			1	1	1	0	0	1	1
FMU-143/B,B/B								0			0	0	0	0	0	0	0
FMU-152/B								1			1	1	1	1	0	0	0
MK43 TDD								0			0	0	0	0	0	0	0
FZU-39/B								0			0	0	0	0	0	0	0
Body Options																	
500 LB								0			0	1	0	0	0	1	0
1000 LB								0			0	0	0	1	0	0	0
2000 LB								1			1	0	1	0	0	0	1
4000 LB								0			0	0	0	0	0	0	0

Appendix G. MRR Weight Matrix

Asset Type & Munitions	Weight (ST)
A10-1	25.3
A10-2	26.8
A10-3	23.5
A10-4	25.1
A10-5	24.5
A10-6	24.5
A10-7	25.1
A10-8	24.5
A10-9	23
A10-10	24.1
A10-11	23.5
F16-1	24.2
F16-2	24.6
F16-3	22.5
F16-4	25.4
F16-5	24.4
F16-6	26.5
F16-7	24.4
F16-8	26.8
F16-9	29.5
F16-10	21.4
F16-11	24.2
F18	22.9
F19	23.9
F20	24.3
B1	22.1
B2	23.9
B3	25.5
B4	22
B5	22.1

Appendix H. Preference Curve Mapping Macro Code

```
Sub JW()
 Start1 = Timer
 Do
   If Worksheets("ILP").Range("v47") >= Worksheets("ILP").Range("w47") Then
      Worksheets("ILP").Range("L69") = Worksheets("ILP").Range("L69") + 10
      Worksheets("ILP").Range("w47") = Worksheets("ILP").Range("v47")
      SolverSolve (True)
   Else
      End
      Exit Do
   End If
   Loop Until Worksheets("ILP").Range("v47") < Worksheets("ILP").Range("w47")   '
Exit outer loop immediately.
If Worksheets("ILP").Range("v47") < Worksheets("ILP").Range("w47") Then
      Worksheets("ILP").Range("L69") = Worksheets("ILP").Range("L69") - 20
      SolverSolve (True)
      Worksheets("ILP").Range("w47") = 0
   Else
      End
   End If
Do
   If Worksheets("ILP").Range("v47") >= Worksheets("ILP").Range("w47") Then
      Worksheets("ILP").Range("L69") = Worksheets("ILP").Range("L69") + 1
      Worksheets("ILP").Range("w47") = Worksheets("ILP").Range("v47")
      SolverSolve (True)
   Else
      End
      Exit Do
   End If
   Loop Until Worksheets("ILP").Range("v47") < Worksheets("ILP").Range("w47")   '
Exit outer loop immediately.
If Worksheets("ILP").Range("v47") < Worksheets("ILP").Range("w47") Then
      Worksheets("ILP").Range("L69") = Worksheets("ILP").Range("L69") - 1
      SolverSolve (True)
      Finish1 = Timer
      Worksheets("ILP").Range("w62") = Finish1 – Start1
   Else
      End
   End If
'JW Macro
'Macro recorded 11/30/2001 by cpunches
End Sub
```

Appendix I. TSOM Definitions and Formulas (Kuykendall, 1998: 16-19)

Indices:

s	ships {e.g., DD-987, DDG-53, CG-54, SSN-720}
i	half-module, dependent on type of ship {e.g., h1-h16 for DD-987}
j	cell, each half-module contains four cells {c1-c4} or {c5-c8}

Note: the valid (s,i,j,) tuples are called *missile locations*.

m	missile type loaded in cell, {e.g., CIII, DII, ASROC, etc…}
n	mission number, total missions known to require tasking {e.g., n1, n2, n3,…} Each mission corresponds to a single requested missile firing.

Given Data:

$load_{sijm}$	equals 1 if initial loadout in location (s,i,j) is a missile of type m, 0 otherwise
$order_{nm}$	equals 1 if mission n calls for missile m, 0 otherwise
rs_n	equals 1 if mission n calls for a ready-spare, 0 otherwise
$bkup_n$	equals 1 if mission n calls for a back-up, 0 otherwise
val_m	relative value for missile m
primepen	elastic penalty for not completing a primary mission
rspen	elastic penalty for not completing a ready-spare mission
bkpen	elastic penalty for not completing a back-up mission
torppen	elastic penalty for assigning a missile not currently loaded in torpedo tubes (refers only to submarine assignments)
$notintube_{sij}$	equals 1 if location (s,i,j) is not in torpedo tubes, 0 otherwise (refers only to submarine)

Derived Data:

ok_{sijn}	equals 1 if missile in location (s,i,j) can be assigned as a primary, ready-spare or back-up for mission n; $ok_{sijn} = \sum_m load_{sijm} * order_{nm}$

Variables:

Missile firing and Assignment

X_{sijn}	equals 1 if missile in location (s,i,j) is fired for primary mission n, 0 otherwise
W_{sijn}	equals 1 if missile in location (s,i,j) is assigned as ready-spare for mission n, 0 otherwise
Z_{sijn}	equals 1 if missile in location (s,i,j) is assigned as back-up for mission n, 0 otherwise
Y_{sij}	equals 1 if missile in location (s,i,j) is fired for a primary mission, 0 otherwise
V_{sij}	equals 1 if missile in location (s,i,j) is assigned as a ready-spare or back-up mission, 0 otherwise

Missile Counting

$HMOD_{sim}$ residual number of missile m on ship s, in half-module i after firing

$SALVO_{sim}$ equals 1 if ship s, half-module i contains one or more missiles of type m after firing, 0 otherwise

Incomplete Missions

$UNABLE_n$ equals 1 if no missile is selected for primary mission n, 0 otherwise

$RSUNABLE_n$ equals 1 if no missile is assigned as ready-spare for mission n, 0 otherwise

$BKUNABLE_n$ equals 1 if no missile is assigned as back-up for mission n, 0 otherwise

Notes on Variable Definitions:

1) X_{sijn}, W_{sijn}, and Z_{sijn} are not defined if $ok_{sijn} = 0$.
2) Y_{sij} and V_{sij} are not defined if $\sum_m load_{sijm} = 0$.
3) $HMOD_{sim}$ is meant to be general integer, but can be treated as continuous since it must equal the sum of binary variables in Constraint (3).
4) $HMOD_{sim}$ and $SALVO_{sim}$ are not defined if $\sum_j load_{sijm} = 0$.
5) $UNABLE_n$, $RSUNABLE_n$, and $BKUNABLE_n$ are meant to be binary variables, but are treated as continuous since they must equal 1 minus the sum of sum of binary variables in Constraints (6) – (8).

Formulation:

1a) MAXIMIZE $\sum_s \sum_i \sum_m val_m * SALVO_{sim}$

1b) $- primepen * \sum_n UNABLE_n$

1c) $- rspen * \sum_n rs_n * RSUNABLE_n$

1d) $- bkpen * \sum_n bkup_n * BKUNABLE_n$

1e) $- torppen * \sum_{sij} notintube_{sij} * (Y_{sij} + V_{sij})$

1f) $+ \sum_s \sum_i HMOD_{s,i,CIII}$

Subject to:

2)	$\sum_j (Y_{sij} + V_{sij}) \leq 1$	$\forall s,i$
3)	$\sum_j load_{sijm} - \sum_j load_{sijm} * Y_{sij} = HMOD_{sim}$	$\forall s,i,m$
4)	$HMOD_{sim} \geq SALVO_{sim}$	$\forall s,i,m$
5)	$\sum_m SALVO_{sim} \leq 1$	$\forall s,i$
6)	$\sum_s \sum_i \sum_j X_{sijn} + UNABLE_n = 1$	$\forall n$
7)	$\sum_s \sum_i \sum_j W_{sijn} + RSUNABLE_n = 1$	$\forall n$ s.t. $rs_n = 1$
8)	$\sum_s \sum_i \sum_j Z_{sijn} + BKUNABLE_n = 1$	$\forall n$ s.t. $bk_n = 1$
9)	$\sum_i \sum_j X_{sijn} \geq \sum_i \sum_j W_{sijn}$	$\forall s,n$ s.t. $rs_n = 1$
10)	$\sum_i \sum_j (X_{sijn} + Z_{sijn}) \leq 1$	$\forall s,n$ s.t. $bk_n = 1$
11)	$Y_{sij} = \sum_n X_{sijn}$	$\forall s,i,j$
12)	$V_{sij} = \sum_n (rs_n * W_{sijn} + bkup_n * Z_{sijn})$	$\forall s,i,j$

Appendix J. MOMGA Formulation (Wakefield, 2000: 52)

The complete MOP formulation is as follows:

Decision variables: Number of MRR j assigned to Task $i = \{x_{1,1} \ldots x_{i,j}\}$

Maximize:

$$S = 0.8x_{1,1} + 0.3x_{1,2} + 0.6x_{1,3} + 0.001x_{1,4} + 0.001x_{1,5} + 0.4x_{2,1} + 0.8x_{2,2} + 0.6x_{2,3} \\ +0.001x_{2,4} + 0.001x_{2,5} + 0.001x_{3,1} + 0.001x_{3,2} + 0.1x_{3,3} + 0.8x_{3,4} + 0.4x_{3,5} \tag{5.1}$$

Minimize:

$$W = 20.2(x_{1,1} + x_{2,1} + x_{3,1}) + 28.5(x_{1,2} + x_{2,2} + x_{3,2}) + 35.7(x_{1,3} + x_{2,3+}x_{3,3}) \\ +19.9(x_{1,4} + x_{2,4} + x_{3,4}) + 22.5(x_{1,5} + x_{2,5} + x_{3,5}) \tag{5.2}$$

$$V = 1650(x_{1,1} + x_{2,1} + x_{3,1}) + 2475(x_{1,2} + x_{2,2} + x_{3,2}) + 2887.5(x_{1,3} + x_{2,3} + x_{3,3}) \\ +1705(x_{1,4} + x_{2,4} + x_{3,4}) + 2200(x_{1,5} + x_{2,5} + x_{3,5}) \tag{5.3}$$

Subject to:

$$x_{1,1}, \ldots, x_{3,5} \geq 0 \tag{5.4}$$

$$\{x_{1,1}, \ldots, x_{3,5}\} \in I \tag{5.5}$$

$$x_{1,1} + x_{1,2} + x_{1,3} + x_{1,4} + x_{1,5} = RLtask_{m,1} \tag{5.6}$$

$$x_{2,1} + x_{2,2} + x_{2,3} + x_{2,4} + x_{2,5} = RLtask_{m,2} \tag{5.7}$$

$$x_{3,1} + x_{3,2} + x_{3,3} + x_{3,4} + x_{3,5} = RLtask_{m,3} \tag{5.8}$$

$$x_{1,1} + x_{2,1} + x_{3,1} \leq RLmrr_{m,1} \tag{5.9}$$

$$x_{1,2} + x_{2,2} + x_{3,2} \leq RLmrr_{m,2} \tag{5.10}$$

$$x_{1,3} + x_{2,3} + x_{3,3} \leq RLmrr_{m,3} \tag{5.11}$$

$$x_{1,4} + x_{2,4} + x_{3,4} \leq RLmrr_{m,4} \tag{5.12}$$

$$x_{1,5} + x_{2,5} + x_{3,5} \leq RLmrr_{m,5} \tag{5.13}$$

where m is the Resource Level index for the current problem.

Appendix K. Trial Runs to Maximum Missions that Improves Total Suitability

		DESIGN APPROACHES/METHODS																					
		Single Batch to 18,000 ST						5 Batches to 18,000 ST (Each 3600)															
		Step-Wise			Jump-Wise			Step-Wise								Jump-Wise							
		Time	Tasks	Value	Time	Tasks	Value	1st Time	2nd Time	3rd Time	4th Time	5th Time	Total Time	Tasks	Value	1st Time	2nd Time	3rd Time	4th Time	5th Time	Total Time	Tasks	Value
Preference Curve 1	Iteration 1	1025.676	779	683.58	126.883	779	683.58	167.871	174.301	173.969	167.41	165.961	849.512	790	683.2	31.594	38.426	38.043	31.543	50.793	190.399	790	683.21
	Iteration 2	1018.965	779	683.58	126.82	779	683.58	167.25	172.219	173.246	167.414	167.574	847.703	790	683.2	31.496	38.027	38.184	31.344	51.082	190.133	790	683.21
	Iteration 3	1019.586	779	683.58	125.18	779	683.58	167.551	172.039	174.277	168.313	167.129	849.309	790	683.2	31.746	38.434	37.824	31.316	51.164	190.484	790	683.21
	Iteration 4	1018.348	779	683.58	126.711	779	683.58	167.48	172.727	173.891	168.262	166.449	848.809	790	683.2	31.688	37.973	37.855	31.359	51.332	190.207	790	683.21
	Iteration 5	1019.848	779	683.58	126.195	779	683.58	168.313	172.746	175.09	167.91	167.961	852.02	790	683.2	31.504	38.234	37.824	31.344	51.594	190.5	790	683.21
	Iteration 6	1019.125	779	683.58	126.641	779	683.58	167.859	172.289	174.43	168.305	168.383	851.266	790	683.2	31.637	38.266	38.113	31.594	51.234	190.844	790	683.21
	Iteration 7	1019.977	779	683.58	126.523	779	683.58	169.195	174.523	174.89	168.422	167.828	854.858	790	683.2	31.75	38.031	37.836	31.32	51.047	189.984	790	683.21
	Iteration 8	1019.023	779	683.58	127.359	779	683.58	168.094	172.492	174.57	166.359	168.664	850.179	790	683.2	31.594	38.188	38.07	31.32	51.305	190.477	790	683.21
	Iteration 9	1020.883	779	683.58	126.461	779	683.58	169.016	177.492	173.891	168.102	168.438	856.939	790	683.2	30.531	36.414	36.281	30.188	47.844	181.258	790	683.21
	Iteration 10	1019.492	779	683.58	127.43	779	683.58	168.344	174.43	174.781	168.609	168.391	854.555	790	683.2	31.695	38.016	36.25	30.492	47.828	184.281	790	683.21
Preference Curve 2	Iteration 1	1382.133	789	710.88	251.539	789	710.88	174.977	172.208	175.102	182.522	236.97	941.779	790	694.8	37.711	36.883	37.07	46.281	81.211	239.156	790	694.78
	Iteration 2	1383.906	789	710.88	251.617	789	710.88	174.611	174.942	174.461	182.413	235.98	942.407	790	694.8	37.398	36.781	37.063	45.125	81.469	237.836	790	694.78
	Iteration 3	1381.172	789	710.88	251.555	789	710.88	173.8	174.921	174.651	182.392	236.29	942.054	790	694.8	37.734	36.758	37.125	44.82	81.563	238	790	694.78
	Iteration 4	1381.328	789	710.88	250.57	789	710.88	175.122	174.481	175.202	182.633	236.38	943.818	790	694.8	37.383	36.75	37.461	45.172	81.531	238.297	790	694.78
	Iteration 5	1390.234	789	710.88	250.93	789	710.88	173.99	174.461	174.581	180.64	236.37	940.042	790	694.8	37.352	36.719	37.656	45.25	81.641	238.618	790	694.78
	Iteration 6	1388.055	789	710.88	250.852	789	710.88	174.42	172.778	173.81	183.664	236.55	941.222	790	694.8	37.352	36.734	37.141	44.914	81.609	237.75	790	694.78
	Iteration 7	1390.273	789	710.88	253.5	789	710.88	174.311	174.071	174.251	182.903	235.889	941.425	790	694.8	37.461	36.773	37.461	45.266	81.477	238.438	790	694.78
	Iteration 8	1389.977	789	710.88	252.586	789	710.88	173.89	174.371	175.182	181.832	236.14	941.415	790	694.8	37.344	37.078	37.453	44.898	81.578	238.351	790	694.78
	Iteration 9	1388.588	789	710.88	250.531	789	710.88	174.09	173.75	175.332	181.841	236.97	941.983	790	694.8	37.313	36.734	37.359	44.805	82.039	238.25	790	694.78
	Iteration 10	1389.578	789	710.88	250.578	789	710.88	174.001	174.001	175.021	182.742	236.86	942.625	790	694.8	37.945	36.734	38.672	44.797	81.477	239.625	790	694.78
Preference Curve 3	Iteration 1	939.07	775	656.19	113.172	775	656.19	175.112	169.394	167.771	162.604	167.751	842.632	782	651.3	38.258	31.641	33.586	36.258	33.141	172.884	782	651.3
	Iteration 2	939.27	775	656.19	112.977	775	656.19	173.259	168.433	169.123	162.482	166.89	840.187	782	651.3	38.227	31.289	33.492	36.492	32.336	171.836	782	651.3
	Iteration 3	936.566	775	656.19	113.023	775	656.19	173.9	168.934	169.252	162.676	167.781	842.543	782	651.3	38.258	31.273	33.5	36.563	32.57	172.164	782	651.3
	Iteration 4	942.758	775	656.19	112.969	775	656.19	173.738	169.125	167.113	162.332	168.313	840.621	782	651.3	38.094	31.258	33.273	36.422	32.695	171.742	782	651.3
	Iteration 5	937.207	775	656.19	113.141	775	656.19	174.672	168.699	169.18	162.715	167.152	842.418	782	651.3	38.313	31.266	33.273	36.367	32.234	171.453	782	651.3
	Iteration 6	936.637	775	656.19	112.906	775	656.19	173.039	167.441	169.14	160.531	167.93	838.081	782	651.3	38.086	31.273	33.344	36.602	32.602	171.907	782	651.3
	Iteration 7	939.051	775	656.19	113.289	775	656.19	173.941	169.109	168.875	160.898	168.23	841.053	782	651.3	38.43	31.273	33.375	36.469	32.484	172.031	782	651.3
	Iteration 8	937.086	775	656.19	113	775	656.19	173.883	169.043	169.266	162.715	168.145	843.052	782	651.3	38.031	31.25	33.172	36.453	32.344	171.25	782	651.3
	Iteration 9	938.602	775	656.19	113.133	775	656.19	174.078	169.113	167.863	162.203	167.965	841.222	782	651.3	38.125	31.32	33.242	37.68	32.742	173.109	782	651.3
	Iteration 10	938.547	775	656.19	112.922	775	656.19	173.809	170.809	168.68	162.453	167.617	843.368	782	651.3	39.695	31.336	33.156	36.797	32.125	173.109	782	651.3
Preference Curve 4	Iteration 1	823.754	777	669.22	108.336	777	669.22	167.43	172.969	169.641	163.711	171.508	845.259	784	664.5	29.93	35.156	31.266	54.859	33.523	184.734	784	664.51
	Iteration 2	823.242	777	669.22	108.227	777	669.22	167.25	171.469	168.648	164.109	171.266	842.742	784	664.5	29.797	33.867	30.633	49.914	32.234	176.445	784	664.51
	Iteration 3	822.805	777	669.22	108.039	777	669.22	166.648	171.82	169.008	164.625	171.766	843.867	784	664.5	28.922	33.945	30.188	49.789	31.906	174.75	784	664.51
	Iteration 4	824.531	777	669.22	108.086	777	669.22	167.352	173.461	167.508	162.914	171.727	842.962	784	664.5	29.078	33.844	30.336	49.938	32.016	175.212	784	664.51
	Iteration 5	824.336	777	669.22	108	777	669.22	167.281	173.281	169.07	164.453	171.492	845.577	784	664.5	28.844	33.836	30.141	50.289	31.914	175.024	784	664.51
	Iteration 6	823.805	777	669.22	108.172	777	669.22	167.336	171.219	168.742	163.07	171.477	841.844	784	664.5	28.859	33.891	30.305	49.852	32.297	175.204	784	664.51
	Iteration 7	822.93	777	669.22	108.039	777	669.22	166.609	170.969	167.148	162.813	169.32	836.859	784	664.5	28.836	33.867	30.18	50.234	32.625	175.742	784	664.51
	Iteration 8	824.742	777	669.22	108.148	777	669.22	167.438	171.219	167.273	161.852	171.68	839.462	784	664.5	29.258	33.875	30.367	49.953	32.602	176.055	784	664.51
	Iteration 9	822.484	777	669.22	108.961	777	669.22	167.242	171.125	169.242	163.813	169.523	840.945	784	664.5	28.82	33.867	30.18	49.945	32.266	175.078	784	664.51
	Iteration 10	822.492	777	669.22	108.703	777	669.22	167.789	174.266	168.922	164.266	170.773	846.016	784	664.5	28.836	33.836	30.461	49.859	32.648	175.64	784	664.51

Appendix L. Trial Runs to 775 Missions

		DESIGN APPROACHES/METHODS																			
		Single Batch to 18,000 ST						5 Batches to 18,000 ST (Each 3600)													
		Step-Wise			Jump-Wise			Step-Wise								Jump-Wise					
		Time	Tasks	Value	Time	Tasks	Value	1st Time	2nd Time	3rd Time	4th Time	5th Time	Total Time	Tasks	Value	1st Time	2nd Time	3rd Time	4th Time	5th Time	Total Time	Tasks	Value
Preference Curve 1	Iteration 1	1014.398	775	682.1	154.523	775	682.1	170.516	176.625	176.742	169.945	149.336	843.164	775	675.1	31.938	38.742	38.219	31.75	37.82	178.469	775	675.11
	Iteration 2	1014.93	775	682.1	154.031	775	682.1	170.883	176.195	176.25	171.359	150.461	845.148	775	675.1	32.023	38.367	38.211	31.68	37.695	177.976	775	675.11
	Iteration 3	1014.18	775	682.1	153.945	775	682.1	171.113	177.156	174.891	170.977	149.18	843.317	775	675.1	31.938	38.344	38.797	31.867	37.648	178.594	775	675.11
	Iteration 4	1016.484	775	682.1	153.703	775	682.1	171.398	174.375	174.664	169.063	151.648	841.148	775	675.1	32.477	38.359	38.195	32.047	37.805	178.883	775	675.11
	Iteration 5	1012.266	775	682.1	153.984	775	682.1	171.297	174.289	176.344	170.93	149.297	842.157	775	675.1	32.016	38.352	38.227	31.68	37.813	178.088	775	675.11
Preference Curve 2	Iteration 1	1325.695	775	707.52	628.461	775	707.52	175.641	175.641	175.289	183.514	164.276	874.361	775	685.4	37.719	37.078	37.414	45.32	50.367	207.898	775	685.41
	Iteration 2	1323.634	775	707.52	628.453	775	707.52	176.063	173.289	176.145	182.613	164.256	872.366	775	685.4	38.188	37.078	37.406	45.484	50.367	208.523	775	685.41
	Iteration 3	1324.113	775	707.52	626.813	775	707.52	175.523	176.064	176.584	182.723	163.986	874.88	775	685.4	37.797	37.508	37.82	45.352	50.195	208.672	775	685.41
	Iteration 4	1326.52	775	707.52	628.656	775	707.52	175.262	173.473	175.855	181.492	163.465	869.547	775	685.4	37.68	37.508	37.414	45.383	50.336	208.321	775	685.41
	Iteration 5	1324.332	775	707.52	628.82	775	707.52	175.059	173.426	175.742	183.973	163.965	872.165	775	685.4	37.797	37.602	37.414	45.352	50.297	208.462	775	685.41
Preference Curve 3	Iteration 1	948.945	775	656.19	125.563	775	656.19	175.914	169.535	170.426	163.695	158.035	837.605	775	647.7	38.586	31.766	33.883	36.828	38.242	179.305	775	647.66
	Iteration 2	948.984	775	656.19	125.711	775	656.19	175	168.48	170.125	162.574	157.508	833.687	775	647.7	38.867	32.063	33.695	36.875	38.258	179.758	775	647.66
	Iteration 3	949.816	775	656.19	126.188	775	656.19	174.77	170.617	168.555	163.859	158.078	835.879	775	647.7	38.688	31.758	33.695	36.797	38.43	179.368	775	647.66
	Iteration 4	949.816	775	656.19	127.18	775	656.19	175.043	170.344	168.543	164.996	158.25	837.176	775	647.7	38.516	31.75	33.703	36.813	37.805	178.587	775	647.66
	Iteration 5	949.375	775	656.19	125.859	775	656.19	174.863	170.324	168.59	163.555	157.746	835.078	775	647.7	38.539	31.969	33.766	36.836	38.258	179.368	775	647.66
Preference Curve 4	Iteration 1	828.211	775	668.27	112.359	775	668.27	167.461	172.238	170.094	164.586	158.797	833.176	775	660.5	30.133	35.492	31.555	54.172	37.695	189.047	775	660.48
	Iteration 2	832.727	775	668.27	112.602	775	668.27	167.836	172.289	170.406	165.305	157.547	833.383	775	660.5	30.047	36.086	31.555	54.086	37.813	189.587	775	660.48
	Iteration 3	829.844	775	668.27	112.469	775	668.27	167.367	172.234	169.664	165.375	157.555	832.195	775	660.5	30.039	35.547	31.578	54.125	37.656	188.945	775	660.48
	Iteration 4	830.063	775	668.27	112.523	775	668.27	168.461	172.219	169.695	165.094	157.891	833.36	775	660.5	30.164	35.484	31.555	54.391	37.508	189.102	775	660.48
	Iteration 5	830.57	775	668.27	112.688	775	668.27	168.398	172.484	170.078	163.164	157.086	831.21	775	660.5	30.023	36.086	31.547	54.078	37.539	189.273	775	660.48

Appendix M. MRR Mix for the 5 Batch/Curve 1 Test

Asset Type & Munitions	Missions									
	AA	*AB*	*AC*	*AD*	*AE*	*AF*	*AG*	*NB*	*SB*	*PB*
A10-1	0	0	0	0	2	0	0	0	0	0
A10-2	0	0	0	0	0	0	0	0	0	0
A10-3	0	0	0	0	1	0	85	0	0	0
A10-4	0	0	0	0	0	0	0	0	0	0
A10-5	0	0	0	0	0	0	0	0	0	0
A10-6	0	0	0	0	0	0	0	0	0	0
A10-7	0	0	0	0	0	0	0	0	0	0
A10-8	0	0	0	0	0	0	0	0	0	0
A10-9	0	0	41	0	0	75	0	0	0	6
A10-10	0	0	0	0	0	0	0	0	0	0
A10-11	0	0	39	60	0	0	0	0	0	0
F16-1	0	0	0	0	0	0	0	0	0	0
F16-2	0	0	0	0	0	0	0	0	0	0
F16-3	0	0	0	0	0	0	0	0	0	0
F16-4	0	0	0	0	0	0	0	0	0	0
F16-5	0	0	0	0	0	0	0	0	0	0
F16-6	0	0	0	0	0	0	0	0	0	0
F16-7	0	0	0	0	0	0	0	0	0	0
F16-8	0	0	0	0	0	0	0	0	0	0
F16-9	0	0	0	0	0	0	0	0	0	0
F16-10	0	115	0	0	52	0	0	0	0	0
F16-11	0	0	0	0	0	0	0	0	0	0
F18	100	0	0	0	0	0	10	0	0	0
F19	0	0	0	0	0	0	0	0	0	0
F20	0	0	0	0	0	0	0	0	0	0
B1	0	0	0	0	0	0	0	0	0	42
B2	0	0	0	0	0	0	0	0	0	42
B3	0	0	0	0	0	0	0	11	0	25
B4	0	0	0	0	0	0	0	24	18	0
B5	0	0	0	0	0	0	0	0	42	0
Total Missions:	100	115	80	60	55	75	95	35	60	115

Appendix N. Wilcoxon Large Sample Approximation Test on Search Methods

CPU Run-Times			Step-Wise	Jump-Wise	Difference
Single Batch	Curve 1	Iteration 1	1014.398	154.523	859.875
		Iteration 2	1014.93	154.031	860.899
		Iteration 3	1014.18	153.945	860.235
		Iteration 4	1016.484	153.703	862.781
		Iteration 5	1012.266	153.984	858.282
	Curve 2	Iteration 1	1325.695	628.461	697.234
		Iteration 2	1323.634	628.453	695.181
		Iteration 3	1324.113	626.813	697.3
		Iteration 4	1326.52	628.656	697.864
		Iteration 5	1324.332	628.82	695.512
	Curve 3	Iteration 1	948.945	125.563	823.382
		Iteration 2	948.984	125.711	823.273
		Iteration 3	949.816	126.188	823.628
		Iteration 4	949.816	127.18	822.636
		Iteration 5	949.375	125.859	823.516
	Curve 4	Iteration 1	828.211	112.359	715.852
		Iteration 2	832.727	112.602	720.125
		Iteration 3	829.844	112.469	717.375
		Iteration 4	830.063	112.523	717.54
		Iteration 5	830.57	112.688	717.882
5 Batches of 3600 ST	Curve 1	Iteration 1	843.164	178.469	664.695
		Iteration 2	845.148	177.976	667.172
		Iteration 3	843.317	178.594	664.723
		Iteration 4	841.148	178.883	662.265
		Iteration 5	842.157	178.088	664.069
	Curve 2	Iteration 1	874.361	207.898	666.463
		Iteration 2	872.366	208.523	663.843
		Iteration 3	874.88	208.672	666.208
		Iteration 4	869.547	208.321	661.226
		Iteration 5	872.165	208.462	663.703
	Curve 3	Iteration 1	837.605	179.305	658.3
		Iteration 2	833.687	179.758	653.929
		Iteration 3	835.879	179.368	656.511
		Iteration 4	837.176	178.587	658.589
		Iteration 5	835.078	179.368	655.71
	Curve 4	Iteration 1	833.176	189.047	644.129
		Iteration 2	833.383	189.587	643.796
		Iteration 3	832.195	188.945	643.25
		Iteration 4	833.36	189.102	644.258
		Iteration 5	831.21	189.273	641.937

		Signed Rank	
Single Batch	Curve 1	Iteration 1	37
		Iteration 2	39
		Iteration 3	38
		Iteration 4	40
		Iteration 5	36
	Curve 2	Iteration 1	23
		Iteration 2	21
		Iteration 3	24
		Iteration 4	25
		Iteration 5	22
	Curve 3	Iteration 1	33
		Iteration 2	32
		Iteration 3	35
		Iteration 4	31
		Iteration 5	34
	Curve 4	Iteration 1	26
		Iteration 2	30
		Iteration 3	27
		Iteration 4	28
		Iteration 5	29
5 Batches of 3600 ST	Curve 1	Iteration 1	16
		Iteration 2	20
		Iteration 3	17
		Iteration 4	12
		Iteration 5	15
	Curve 2	Iteration 1	19
		Iteration 2	14
		Iteration 3	18
		Iteration 4	11
		Iteration 5	13
	Curve 3	Iteration 1	9
		Iteration 2	6
		Iteration 3	8
		Iteration 4	10
		Iteration 5	7
	Curve 4	Iteration 1	4
		Iteration 2	3
		Iteration 3	2
		Iteration 4	5
		Iteration 5	1
			820

α = 0.01

$Z_{Critical}$ = 2.33 Table A.3 Standard Normal Curve Area (Devore, 2000: 723)

z = 5.5109 **The null is rejected since $z > Z_{critical}$**

Appendix O. Friedman Multiple Rank Test on the Preference Curves

			CPU Run-Times			
			Curve 1	Curve 2	Curve 3	Curve 4
Single Batch	Step-Wise	Iteration 1	1014.4	1325.7	948.945	828.211
		Iteration 2	1014.93	1323.63	948.984	832.727
		Iteration 3	1014.18	1324.11	949.816	829.844
		Iteration 4	1016.48	1326.52	949.816	830.063
		Iteration 5	1012.27	1324.33	949.375	830.57
	Jump-Wise	Iteration 1	154.523	628.461	125.563	112.359
		Iteration 2	154.031	628.453	125.711	112.602
		Iteration 3	153.945	626.813	126.188	112.469
		Iteration 4	153.703	628.656	127.18	112.523
		Iteration 5	153.984	628.82	125.859	112.688
Five Batches of 3600 ST	Step-Wise	Iteration 1	843.164	874.361	837.605	833.176
		Iteration 2	845.148	872.366	833.687	833.383
		Iteration 3	843.317	874.88	835.879	832.195
		Iteration 4	841.148	869.547	837.176	833.36
		Iteration 5	842.157	872.165	835.078	831.21
	Jump-Wise	Iteration 1	178.469	207.898	179.305	189.047
		Iteration 2	177.976	208.523	179.758	189.587
		Iteration 3	178.594	208.672	179.368	188.945
		Iteration 4	178.883	208.321	178.587	189.102
		Iteration 5	178.088	208.462	179.368	189.273

CPU Run-Times Rank Ordered			
Curve 1	Curve 2	Curve 3	Curve 4
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
3	4	2	1
1	4	2	3
1	4	2	3
1	4	2	3
2	4	1	3
1	4	2	3
51	80	39	30

A_2 =	600	k_1 =	3
B_2 =	571.1	k_2 =	57
T_2 =	46.7439	α =	0.01

$T_{Critical}$ = 4.16 Table A26 (Conover, 1980: 485)

The null hypothesis is rejected $T_2 > T_{critical}$

The F distribution is used to evaluate the test statistic, T_2, since it was found to be superior to previous chi-square approximation. (Conover, 1980: 300)

Multiple Comparisons: using the *t* distribution with (*b*-1)(*k*-1) degrees of freedom

$t_{1-\alpha/2}$ = 2.667 Table A25 (Conover, 1980: 480)

Any test more than 12.0106 units apart may be regarded as unequal.

Ordering and underscoring yields

Curve 4	Curve 3	Curve 1	Curve 2
30	39	51	80

*Underscoring indicates Curves that do not statistically differ

Appendix P. Friedman Multiple Rank Test of Batching Methods

Runtimes			Single Batch	Multi Batch
Step-Wise	Curve 1	Iteration 1	1014.398	843.164
		Iteration 2	1014.93	845.148
		Iteration 3	1014.18	843.317
		Iteration 4	1016.484	841.148
		Iteration 5	1012.266	842.157
	Curve 2	Iteration 1	1325.695	874.361
		Iteration 2	1323.634	872.366
		Iteration 3	1324.113	874.88
		Iteration 4	1326.52	869.547
		Iteration 5	1324.332	872.165
	Curve 3	Iteration 1	948.945	837.605
		Iteration 2	948.984	833.687
		Iteration 3	949.816	835.879
		Iteration 4	949.816	837.176
		Iteration 5	949.375	835.078
	Curve 4	Iteration 1	828.211	833.176
		Iteration 2	832.727	833.383
		Iteration 3	829.844	832.195
		Iteration 4	830.063	833.36
		Iteration 5	830.57	831.21
Jump-Wise	Curve 1	Iteration 1	154.523	178.469
		Iteration 2	154.031	177.976
		Iteration 3	153.945	178.594
		Iteration 4	153.703	178.883
		Iteration 5	153.984	178.088
	Curve 2	Iteration 1	628.461	207.898
		Iteration 2	628.453	208.523
		Iteration 3	626.813	208.672
		Iteration 4	628.656	208.321
		Iteration 5	628.82	208.462
	Curve 3	Iteration 1	125.563	179.305
		Iteration 2	125.711	179.758
		Iteration 3	126.188	179.368
		Iteration 4	127.18	178.587
		Iteration 5	125.859	179.368
	Curve 4	Iteration 1	112.359	189.047
		Iteration 2	112.602	189.587
		Iteration 3	112.469	188.945
		Iteration 4	112.523	189.102
		Iteration 5	112.688	189.273

Ranks	Single Batch	Multi Batch
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	2	1
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	2	1
	2	1
	2	1
	2	1
	2	1
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	1	2
	60	60

$A_2 =$ 200 $\alpha = 0.01$ $k_1 =$ 1

$B_2 =$ 180 $k_2 =$ 39

$T_2 =$ 0.0000

$T_{Critical} =$ 7.34 Table A26 (Conover, 1980: 485)

The null hypothesis is <u>not</u> rejected $T_2 < T_{critical}$

Bibliography

Aspin, Les. *Interim Report of the Committee on Armed Services, House of Representatives*. 30 March 1992. Memorandum for Members. n. pag. 3 August 2001. http://es.rice.edu/projects/Poli378/Gulf/aspin_rpt.html.

Air Force Studies and Analyses Agency. *Combat Forces Assessment Model (CFAM) Training Manual*. Contract DASW01-94-D-0060 with SRS Technologies ASI Division. Washington DC, 28 August 1997.

Brooke, A., Kendrick D. and Alexander Meeraus. *GAMS: User Guide, Release 2.25*. Redwood City CA: The Scientific Press, 1992.

Buzo, Christopher D. *A Decision Support Tool to Aid Campaign Planners in Selecting Combat Aircraft for Theater Crisis*. MS thesis, AFIT/GEE/ENS/00M-02. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2000.

Carlson, James R., Michael J. Sierra, and Philip A. King. *Strategy, Policy, and Contingency Planning: the US Defense Planning Process*. U.S. Army War College, Carlisle Barracks PA, 1984.

Carrico, Todd M. "ALP Overview: Background." *Advanced Logistics Project*. United States, Defense Advanced Research Projects Agency. Excerpt from unpublished article. 6 April 2000. n. pag. 23 June 2001. http://www.darpa.mil/iso/alp/.

-----. *Introduction to the Advanced Logistics Project (ALP)*. Advanced Logistics Project Workshop VII. United States, Defense Advanced Research Projects Agency. Excerpt from unpublished presentation. 6-8 December 2000b. n. pag. 23 November 2001. http://www.cougaar.org/documents/presentations/workshop/Intro_to_ALP.pdf.

-----. "Automated Logistics Plan Construction to be Shown at ALP '98 Demonstration." Joint Logistics Technology Pipeline 1: 5 (October 1998)

Chang, Shoou-Yuh. "A System Optimization Model for Low-Level Nuclear Waste Disposal," in *Mathematical and Computer Modeling*, Volume 14. New York: Pergamon Press, 1990.

Cioppa, Thomas M. *Force Tailoring Tools*. Technical Document TRAC-TD-0196. TRADOC Analysis Center-Study and Analysis Center, Study Directorate, Fort Leavenworth KS, April 1996.

Cohen, William S. *Annual Report to the President and Congress*. 1998. n. pag. 3 November 2001. http://www.dtic.mil/execsec/adr98.

Colvard, Michael J. *An Analysis of the Interaction Between the J3 and J4 War Planning Staffs During the Phases of Crisis Action Planning*. MS thesis, AFIT/GLM/ENS/01M-06. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2001.

Conover, W. J. *Practical Nonparametric Statistics*. Second Edition. New York: John Wiley & Sons, 1980.

Crump, Pete. "Portable Flight Planning System (PFPS)." Excerpt from unpublished article. Fort Walton Beach FL: 2000. n. pag. 21 November 2001. http://www.tybrin.com/services/federal/missionplan/mpproduct_jmps.htm.

Department of the Army. *Army Vision 2010*. AV 2010. Washington: Army Chief of Staff, 1 October 1996. n. pag. 3 November 2001. http://www.army.mil/2010/default.htm.

Department of Defense (DoD). "Department of Defense Chemical and Biological Defense Program." *Annual Report to Congress and Performance Plan*. Washington: Secretary of Defense, July 2001. n. pag. 3 November 2001. http://www.defenselink.mil/pubs/chem_bio_def_program/2001_CBDP_Annual_Report.pdf.

-----. *Joint Staff Officers Guide*. AFSC Pub 1. Washington: GPO, 1997.

-----. *Joint Vision 2010*. Washington: CJCS, 1997.

Devore, Jay L. *Probability and Statistics for Engineering and the Sciences*. Pacific Grove CA: Duxbury, 2000.

Director, Operational Test and Evaluation (DOT&E). "Joint Mission Planning System (JMPS)." FY99 Annual Report. Washington: DOT&E, 1999. n. pag. 21 November 2001. http://www.dote.osd.mil/reports/FY99/navy/99jmps.html.

Eglin. http://wmnet.eglin.af.mil/weapons/appendix_a.doc. 15 October 2001.

FAS. http://www.fas.org/man/dod-101/sys/ac/equip/index.html. 18 October 2001.

Filcek, Paul G. *A Quantitative Decision Support Model to Aid Selection of Combat Aircraft Force Mixes for Contingency Deployment*. MS thesis, AFIT/GLM/ENS/01M-10. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2001.

Frontline Systems, Inc. *Premium Solver Platform Solver Engines*. User Guide. Incline Village NV: Frontline Systems, Inc, 2000.

-----. *Premium Solver Platform for use with Microsoft Excel*. User Guide. Incline Village NV: Frontline Systems, Inc, 2000b.

Glover, Fred and Manuel Laguna. *TABU SEARCH*. Norwell MA: Kluwer Academic Publishers, 1999.

Goddard, Matthew W. *Estimating Deployed Airlift and Equipment Requirements for F-16 Aircraft in Support of the Advanced Logistics Project*. MS thesis, AFIT/GLM/ENS/01M-11. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2001.

Godfrey, Brian M. *A Comparison of Proposed Air Expeditionary Force Packages Using the Thunder Campaign Simulation Program*. MS thesis, AFIT/GLM/LAL/98S-6. School of Acquisitions and Logistics, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1998.

Holleran, Gail. "United States Air Force UTC Refinement Effort." *Unpublished Presentation*. 3 October 2000. n. pag.

Hughes, David. "New Planning Software Aids Bosnian Airdrops," Aviation Week and Space Technology, 138(17): 59-61 (April 1993).

Jackson, Jack A. Jr. *A Taxonomy of Advanced Linear Programming Techniques and the Theater Attack Model*. MS thesis, AFIT/GST/ENS/89-7. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, September 1989.

Judge, Paul J. *The Potential Influence of Advanced Logistics on Defense Air Transportation*. School Of Logistics And Acquisition Management, Air Force Institute Of Technology, Wright-Patterson AFB OH, June 1998. (AD-A354260).

Johnson, Alan W. and Stephen M. Swartz. "Mission-Resource Value Assessment Technique." *Unpublished briefing*. ALP Winter Conference. dec00.5L.ppt December 2000.

Koleszar, George E., James N. Bexfield, and Frederic A. Miercort. *A Description of the Weapon Optimization and Resource Requirements Model (WORRM)*. IDA Document D-2360. Institute for Defense Analyses, Alexandria VA, August 1999.

Kuykendall, Scott D. *Optimizing Selection of Tomahawk Cruise Missiles*. MS thesis. Department of Operations Research, Naval Postgraduate School, Monterey CA, March 1998.

Ljung, Lennart and Torkel Glad. *Modeling of Dynamic Systems*. Englewood cliffs, NJ: P T R Prentice Hall, 1994.

Looney, William R. "The Air Expeditionary Force: Taking the Air Force into the Twenty-first Century." *Airpower Journal*. 4-9, Winter 1996.

Murty, Katta G. *Linear and Combinatorial Programming*. New York: John Wiley and Sons, 1976.

Osgood, John. "The Goldwater Nichols Act – Managing the Defense Department." 1998. Excerpt from unpublished article. n. pag. 3 December 2001. http://pw1.netcom.com/~jrosgood/w16.htm.

Pagonis, William G. and Jeffrey L. Cruikshank. *Moving Mountains: Lessons in Leadership and Logistics from the Gulf War*. Boston: Harvard Business School Press, 1992.

Ragsdale, Cliff T. *Spreadsheet Modeling and Decision Analysis*. Third Edition. Cincinnati: South-Western Publishing, 2001.

RAND. "Rapidly Halting an Armored Invasion: EFX '98." Project AIR FORCE. *Unpublished Presentation*. December 1997. Slide 1-33. 18 November 2001. http://fas.org/man/dod-101/usaf/docs/rand_ed/sld001.htm.

Shaneman, Keith S. "Improving AEF Deployment Planning." Joint Logistics Technology Pipeline 2: 6-7 (February 1999).

Smith, Jon M. *Mathematical Modeling and Digital Simulation for Engineers and Scientists*. New York: A Wiley-Interscience Publication, 1977.

Swartz, Stephen M. Assistant Professor of Logistics Management, Department of Operational Sciences, Air Force Institute of Technology, Wright-Patterson AFB OH. Personal interview. 30 July 2001.

-----. "ALP Pilot Problem and Derivation of Mathematical Model." *Unpublished report*. Wright-Patterson AFB OH, 1999.

Wakefield, David J, Jr. *Identification of Preferred Operational Plan Force Mixes Using a Multiobjective Methodology to Optimize Resource Suitability and Lift Cost*. MS thesis, AFIT/GLM/ENS/01M-24. School of Engineering and Management, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, March 2001.

Williams, H. P. *Model Building in Mathematical Programming*. Second Edition. New York: A Wiley-Interscience Publication, 1985.

Vita

On 13 December 1985, then Private Punches enlisted in the US Army and entered training as a Nuclear, Biological, and Chemical Warfare Operations Specialist. He earned a Bachelor of Science in Business degree with a major in Business Administration from Emporia State University, Kansas in 1992. He graduated Summa Cum Laude with Honors, earned 7 national academic honors such as Who's Who of American Universities, and had one article published.

In October 1995, he entered Officer Training School where he earned distinction as the Outstanding Graduate. Upon commissioning, he served as a supply officer at Ellsworth AFB, South Dakota. During his two-and-a-half year stay, he was a flight commander for the Fuels Flight and an OIC for the Combat Operations Flight. He was also recognized as the Eighth Air Force Junior Supply Manager of the Year, the Wing Company Grade Officer of the Quarter, and the Logistics Group Company Grade Officer of the Year. In August 1998, Captain Punches was assigned as the Materiel Storage and Distribution Flight Commander and later the Management and Systems Flight Commander for the 36th Air Base Wing at Andersen AFB, Guam. While at Andersen, he was recognized as the 13th Air Force Lance P. Sijan Leadership Award winner and the Pacific Air Force Junior Supply Manager of the Year nominee.

In August 2000, he entered the Graduate Logistics Management program at the Air Force Institute of Technology. His superior academic achievement earned him selection into Sigma Iota Epsilon, the National Honorary and Professional Management Fraternity. Upon graduation, he will be assigned to HQ PACAF, Hickam AFB, Hawaii.

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 074-0188*

1. REPORT DATE *(DD-MM-YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From – To)*
20-03-2002	Master's Thesis	1 Mar 2001 – 20 Mar 2002

4. TITLE AND SUBTITLE A LARGE SCALE INTEGER LINEAR PROGRAM AS A DECISION SUPPORT TOOL FOR FORCE MIX SELECTION	**5a. CONTRACT NUMBER** **5b. GRANT NUMBER** **5c. PROGRAM ELEMENT NUMBER**
6. AUTHOR(S) Punches, Craig A., Capt, USAF	**5d. PROJECT NUMBER** **5e. TASK NUMBER** **5f. WORK UNIT NUMBER**
7. PERFORMING ORGANIZATION NAMES(S) AND ADDRESS(S) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 P Street, Building 640 WPAFB OH 45433-8865	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT/GLM/ENS/02-15
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES) Dr. Todd Carrico DARPA/ISO 3701 North Fairfax Drive Arlington, Virginia 22203-1714 (703) 526-6616	**10. SPONSOR/MONITOR'S ACRONYM(S)** **11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION/AVAILABILITY STATEMENT
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT
In the post cold war environment, the rapid deployment of combat capability is critical. Deployment lift capability is limited, however, so the real-time selection of the optimal combat asset mix that balances capability provided and sustainment required has become paramount. In this model, the value of a force mix is determined by the sum of the individual weapon system "suitabilities" against their assigned missions. The value is constrained by the numerical limits on the items required to create and support the force mix, and the lift required to move these items.

The research considered heuristic and complete enumeration methods against the problem structure to develop a decision support model that expedites the selection of the best overall force mix. War planners are provided a decision support tool that objectively compares alternative force mix packages and selects the optimal asset mix in a reasonable amount of time while explicitly considering logistics constraints. This demonstrates the feasibility of an approach that integrates intelligence, operations, and logistics issues into a single decision support and planning tool for force mix decisions.

15. SUBJECT TERMS
Optimization, Military Planning, Defense Planning, Air Force Planning, Logistics Planning, Logistics Management, Linear Programming, Integer Programming, Force Mix

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON**
					Maj Stephen M. Swartz
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(Include area code)*
U	U	U	UU	150	(937) 255-6565, ext 4284 Stephen.Swartz@afit.af.edu