Optimized Air Asset Scheduling Within a Joint Aerospace Operations Center (JAOC)

by

Kevin Joseph Rossillon

B.S. Operations Research, B.S. Mathematics
United States Air Force Academy (2013)

Submitted to the Sloan School of Management
in partial fulfillment of the requirements for the degree of

Masters of Science in Operations Research

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2015

Author . . . .
Sloan School of Management
May 14, 2015

Certified by . . . .
Sung-Hyun Son
Assistant Group Leader, Lincoln Laboratory Group 36
Thesis Supervisor

Certified by . . . .
Dimitris Bertsimas
Boeing Professor of Operations Research
Co-Director, Operations Research Center
Thesis Supervisor

Accepted by . . . .
Patrick Jaillet
Dugald C. Jackson Professor, Department of Electrical Engineering and Computer Science
Co-Director, Operations Research Center

Optimized Air Asset Scheduling Within a Joint Aerospace Operations Center (JAOC)

by

Kevin Joseph Rossillon

Submitted to the Sloan School of Management
on May 14, 2015, in partial fulfillment of the
requirements for the degree of
Masters of Science in Operations Research

Abstract

In this thesis, we introduce and analyze models for air asset scheduling within a military theater. Specifically, we seek to create models that generate aircraft-specific schedules for Air Tasking Orders (ATOs) within a Joint Aerospace Operations Center (JAOC).

A JAOC provides command and control of all air and space assets tasked to a particular region/area of responsibility (AOR) or strategic command. Scheduling these assets requires a high level of unified effort whereby centralized planning must be handled in a decentralized fashion and is known as the Air Tasking Cycle. Given the complexity of this process, subject matter experts from diverse backgrounds are required to design and plan missions for most operations. In addition, the difficulty of the process dictates that mission prioritization and aircraft/munitions allocation are separated in the cycle, sacrificing some global perspective for the sake of efficiency in the scheduling process.

We present a modeling framework that allows planners to simultaneously select missions and assign aircraft/munitions to the missions, allowing for the optimal air asset scheduling toward the pursuit of theater-level objectives. This flexible framework takes into account air refueling considerations as well as the need for certain missions to be completed by "packages" of particular aircraft types. We submit heuristic, mixed integer optimization (MIO), and hybrid models within this structure and analyze the value of their schedules and the corresponding trade-offs with computational solve time.

Thesis Supervisor: Sung-Hyun Son
Title: Assistant Group Leader, Lincoln Laboratory Group 36

Thesis Supervisor: Dimitris Bertsimas
Title: Boeing Professor of Operations Research
Co-Director, Operations Research Center

Acknowledgments

I would like to sincerely thank everyone who has helped me both write this thesis and complete my degree at MIT. If I took the time to thank you all sufficiently, I would have to add another chapter to my thesis.

I would like to thank Steve Relyea for working with me week in and week out to guide my research. Whether it was scribbling the fiftieth formulation of my model on a whiteboard to make modifications, editing my thesis to make it readable for someone who hadn't been studying JAOC operations for over a year, or simply giving me advice on my classwork, your guidance and mentorship were essential. My thesis would not be what it is without your help.

I would like to thank my advisers, Sung-Hyun Son and Dimitris Bertsimas. Sung, thank you for supporting me and giving me the flexibility to pursue a project I was interested in. I would not have learned so much about Air Force operations and my technical interests without your support. Dimitris, thank you for meeting with me weekly and constantly pushing me to make a better model. Your modeling expertise was instrumental in driving this research.

I would like to thank Steve Rothstein, "Warbucks" Dillon, "RayBob" Barker, Chuck Charlton, Orton Huang, and Jerry Rossillon for talking to me about Air Force mission planning, the workings of the JAOC, and the Air Tasking Cycle. Without your help (and the supporting documentation you provided me), I would not have been able to even approach this problem. While I know I likely made unrealistic assumptions or simply misrepresented JAOC operations, I can say with confidence that I would have made even more errors without your help. Any realism in this problem is to your credit; any errors are solely my own.

I would like to thank all of my friends and colleagues at the ORC who helped me with my research, my classwork, and my sanity. Dan, Jack, Zeb, Will, Mallory, Mapi, Virgile, Zach, Krish, Joey, and the rest of my friends at the ORC, your PSET help, research edits, and conversation in the hallway helped me get through every day. I certainly had a wonderful time in Boston, largely due to our friendships.

Finally, I would like to thank my friends, and especially my family, for motivating me to work hard and get my research done. You know better than anyone that I like to complain, and your patience and advice drove me to succeed anyway, For that, I can't thank you enough.

Contents

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

Chapter 1

Introduction

Every day, the US military flies hundreds of air missions around the world in support of war and peacetime operations. The goal of these missions varies widely, including but not limited to strike operations, defensive counterair, reconnaissance, and airlift. Given the global presence of US air assets, the military establishes regional Joint Aerospace Operations Centers (JAOCs) wherever there is "a standing or potential force" [27]. These JAOCs are responsible for the command and control of all air and space assets in the region, including all aircraft and ballistic missiles. In addition, JAOCs task all aircraft with missions designed to accomplish force objectives in that region, and consequently, the efficient and intelligent use of air assets drives the effectiveness of operations within the theater.

Planning missions for hundreds of aircraft of various types and functions is a difficult task in and of itself. This process is complicated by mission requirements dictating that aircraft of different (or similar) types must fly missions in "packages," necessitating coordination between aircraft that may begin missions from different bases. Additionally, many aircraft will require air refueling. This requirement spawns further complexity as fuel considerations must be managed for both aerial tankers and fuel-receiving aircraft, and the air refueling sites must be selected.

In this thesis, we present a modeling framework that allows planners to create aircraft-by-aircraft schedules that optimize the pursuit of theater-level objectives. This framework captures the complexities involved with required air "packages" for mission completion, while also accounting for air refueling coordination. We examine the effectiveness of heuristic, mixed integer optimization (MIO), and hybrid models, analyzing their usefulness as scheduling aids for planners within a JAOC. While these models have some limitations when considering larger theater-level problems, we believe this modeling framework provides a proof of concept of simultaneous mission planning and aircraft/munitions allocation as a technique for increasing the value of air asset schedules.

1.1 Thesis Structure

This thesis is structured first to introduce the JAOC asset scheduling problem and the advantages of utilizing an automated model to aid in the planning process. Consequently, we discuss the purpose of the JAOC, highlighting areas for improvement in the planning and scheduling process. Then, we introduce models that address these weaknesses and analyze the effectiveness of the models as aids in the planning cycle.

In Chapter 2, we describe in depth the Air Tasking Cycle within a JAOC. We also discuss some of the tenants of air refueling, in particular as they apply to intratheater refueling. The chapter also contains a literature review, including an analysis of discrete-time versus continuous-time modeling and a survey of literature on mission planning.

In Chapter 3, we utilize our understanding of the Air Tasking Cycle to develop a modeling framework conducive to aiding planners in a JAOC. We specifically develop a MIO methodology and a greedy algorithm within this framework. We also provide optional constraints for the MIO model that give mission planners more flexibility in choosing which combinations of missions will be scheduled.

In Chapter 4, we run trials of our models and compare the results of the greedy algorithm against the MIO model. We examine the schedules produced on mock scenarios and investigate the performance of the MIO model when fewer missions are provided. Given these insights, we create a hybrid model combining the greedy heuristic and MIO models and explore the computational results of this model.

Finally in Chapter 5, we summarize our findings and assess the viability of our models as planning aids both now and in the future. In addition, we make suggestions for future work, both as an extension to the proposed modeling framework in this thesis and in the domain of JAOC planning, in general.

1.2 Contributions

We make the following contributions in this thesis:

- We propose a modeling framework that manages both the mission selection and aircraft/munitions allocation stages of the Air Tasking Cycle (see Chapter 2). Specifically, our framework creates an aircraft-specific schedule for every aircraft over the entire time period of the model. That is, we do not simply assign a squadron of aircraft to particular missions for completion, nor do we assign aircraft to missions without specifying the mission completion time. Rather, our models specify exactly where every aircraft will be operating for the duration of the time horizon. In addition, our structure honors the constraint that missions be completed by a "package" of aircraft of various types. It also manages air refueling by routing all aircraft through anchor areas (see "Air Refueling Within a JAOC" in Chapter 2 for a description) and coordinating fuel transfer between tankers and other aircraft in these areas.

- We apply this framework to create both a greedy heuristic algorithm, as well as a MIO formulation for creating air asset schedules. We also introduce certain optional constraints that can be added to the MIO formulation to manage munitions or require synchronization between certain missions, for example.

- We compare the greedy algorithm and various subsets of the MIO formulation, considering both "value" of missions and the number of missions scheduled, and contrasting these measures of success with computational run time. Given these results, we introduce a hybrid model of the greedy and MIO models that balances computation time with value added.

- We demonstrate the viability of the hybrid model. Specifically, for a problem of 20 aircraft, 4 tankers, and 50 potential missions, after just one hour of run time, we obtain an average solution that is within 8.1 percent of the provable optimal solution. This solution represents a 6.39 percent increase in mission "value" as compared to the greedy algorithm's solution, which approximates current scheduling methods. In addition, we show a 4.6 percent increase from the greedy algorithm's solution on a problem of 80 aircraft, 10 tankers, and 100 potential missions in one hour of run time. Given these results, we discuss the usefulness of this model to planners in a JAOC.

Chapter 2

Problem Background and Literature Review

2.1 Optimized Air Mission Planning

Every day, the US military flies hundreds of missions around the world in support of war and peacetime operations. The goal of these missions varies widely to include strike operations, defensive counterair, reconnaissance, airlift, and many others. Consequently, the branches of the US military operate over a hundred different types of aircraft, with the total number of aircraft nearing 10,000. Acquisition, maintenance, and operation of these aircraft is extremely expensive. For example, a recent RAND study found the average annual cost of operating a single fighter wing (about 100 aircraft) to be nearly $500 million [23].

Smarter mission planning would allow the US military to complete more missions (and/or more valuable missions) with the aircraft it currently has available, which would provide for quicker conflict resolution while limiting military expenditures. The goal of this thesis is to demonstrate a MIO framework that would allow the US military to plan more effectively and efficiently. Specifically, our model optimizes the the total "value" of all missions flown within a region given a finite set of aircraft of varying types (to include tankers), outputting an aircraft-by-aircraft schedule for a user-controlled time horizon. The model focuses on intratheater mission planning based out of a regional JAOC. While this framework is by no means all-encompassing, it is our hope that it provides proof of concept of model-based optimization for mission planning and will inspire further development in this area.

2.1.1 Mission Planning Within a JAOC

A JAOC provides command and control of all air and space assets tasked to a particular region/area of responsibility (AOR) or strategic command. Regional JAOCs exist "wherever there is a standing or potential joint force," such as the JAOC at Al Udeid in Qatar, which is responsible for the Middle East, ranging from Egypt to Syria to Kazakhstan to Pakistan and everything in between [27]. The AOR for a JAOC can vary greatly from the size of a country (such as the 607th JAOC in charge of Korea) to the size of a continent (such as the 603rd JAOC in charge of Europe and Africa) based on the political situation and operations tempo in the region. In most cases, the JAOC is commanded by the Joint Forces Air Component Commander (JFACC), whose role is to advise the Joint Forces Commander (JFC) on how he (or she) can best use his air and space assets. While the JFC is ultimately responsible for all operations within his AOR, he generally delegates authority for air and space operations to the JFACC. Hundreds of personnel to include subject matter experts from a variety of backgrounds (e.g., pilots, lawyers, etc.) support the JFACC in this mission.

The process of centralized control by which the JFACC controls air forces within a joint air operations environment is known as an Air Tasking Order (ATO) [20]. These orders, which generally cover a 24-hour period, list all sorties, both offensive and defensive, to be flown during that ATO period along with call signs, mission types, and aircraft types [20]. The creation of ATOs are (generally) planned out by hand by military members during a 72-hour cycle in advance of a particular ATO day. Consequently, five ATOs are simultaneously being planned, executed, or assessed at any given time. The ATO process involves selecting targets and assigning the appropriate aircraft and payload to those targets in accordance with the objectives set forth by the JFC. The timely completion of an ATO is currently a six part process that involves the successive completion of many documents, requiring a high level of unified effort whereby centralized planning must be handled in a decentralized fashion [7].

The ATO Cycle, shown in Figure 2-1, is composed of six stages: Objectives, Effects, and Guidance; Target Development; Weaponeering and Allocation; ATO Production and Dissemination; Execution Planning and Force Execution; and Assessment. The process is cyclical; strategy development leads to mission selection, which is limited by available resources. Selected missions are executed in accordance with campaign goals, although modifications may occur on the spot to account for last minute changes. Once executed, assessment of whether mission objectives are being met drives the evolution of strategy, which restarts the entire ATO development process [20].

To support this cycle, the JAOC is separated into five divisions: Strategy; Combat Plans; Combat Operations; Intelligence, Surveillance, and Reconnaissance (ISR); and Air Mobility. The Strategy Division is in charge of big picture perspective. Personnel in this division define campaign objectives and determine whether current operations are meeting objectives. The Combat Plans Division is in charge of day-to-day mission planning, to include target development and resource allocation. As our model is designed to aid this division, its sub-processes are covered in more detail below. The Combat Operations Division implements the ATO, sometimes also executing missions not on the schedule if high-valued Time Sensitive Targets (TSTs) appear. The ISR Division and Air Mobility Division operate in support capacities to the other three divisions, providing critical intelligence and logistics information that aids in both planning and execution of air operations [20, 27].

The Combat Plans Division is responsible for executing three steps within the Air Tasking Cycle: Target Development, Weaponeering and Allocation, and ATO Production and Dissemination. In the Target Development stage, the commander's guidance is taken and used to develop a Joint Air Operations Plan (JAOP), which lays out how available forces and capabilities can be employed. Specifically, each air wing component as well as the ISR Division develops a Target Nomination List (TNL),

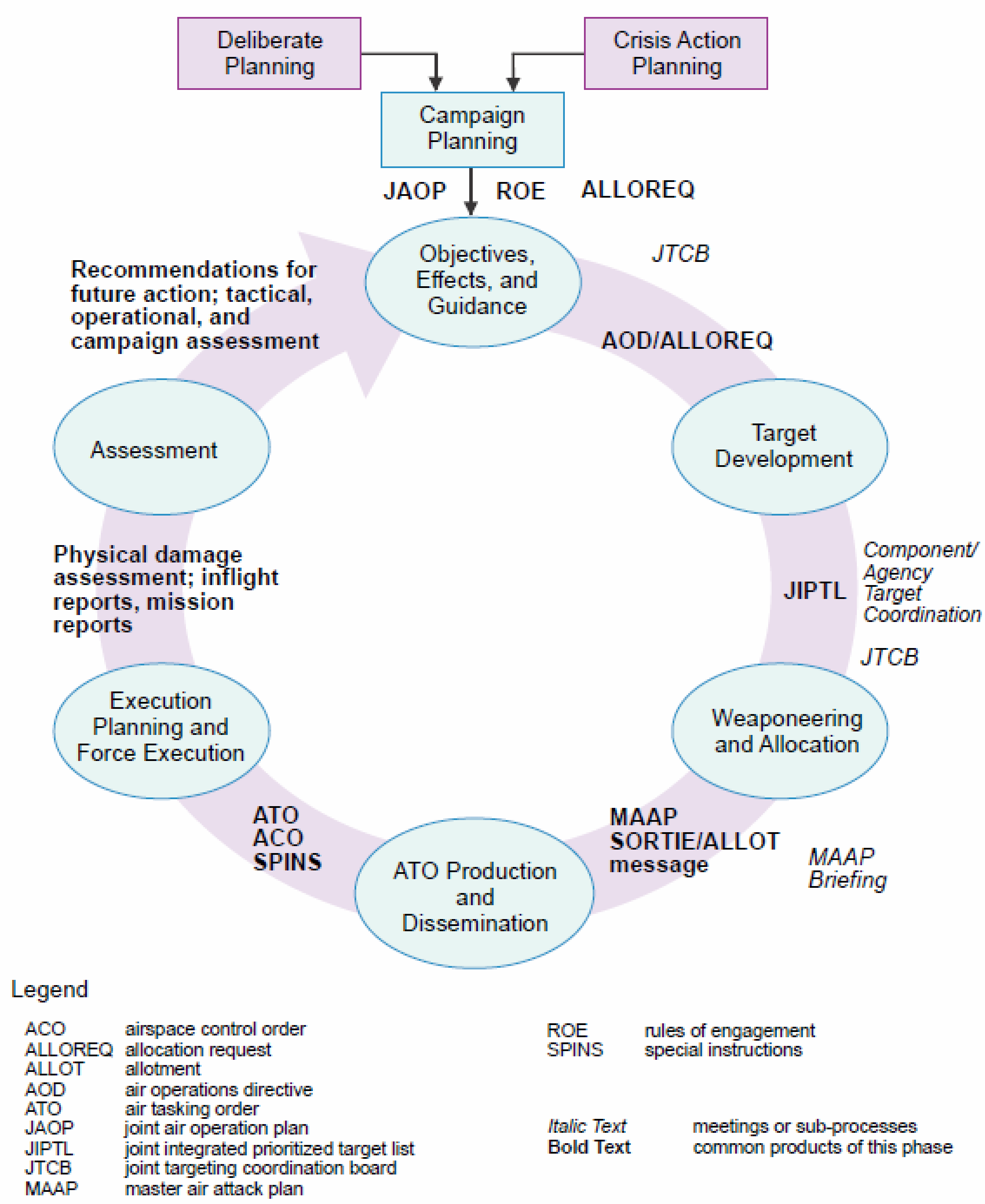

Figure 2-1: Air Tasking Cycle at a JAOC [20]

which is then merged to create the Joint Integrated Prioritized Target List (JIPTL) [7]. Once approved, the JIPTL becomes the groundwork for the ATO, containing the listing of targets in order of priority, normally with a potential "cut line" of which targets can be completed, all of which meet the commander's intent [5].

In the Weaponeering and Allocation stage, the Master Air Attack Plan (MAAP) Team within the Combat Plans Division creates the MAAP, a document that quantifies the predicted results of both lethal and non-lethal weapons and other capabilities used to ensure target objectives outlined in the JIPTL are met [20]. This plan details the estimated effects of the weapons employment and provides a foundation for the joint ATO [5]. Furthermore, target nominations may change slightly from the JIPTL to reflect capability limitations, changes in the commander's intent, or conflict in the allocation of aircraft or other forces [20]. Once developed, the MAAP is reviewed, and the JFC staff allocates which sorties will be flown by what aircraft on what mission, excess sorties not required, and/or requests for additional air support [20].

In the ATO Production and Dissemination stage, the MAAP is approved by the JFACC, all apportionment decisions are reviewed, and the specifics of the order are finalized [7]. During this phase, the backbone of ATO planning has already been completed; specifics are merely being rigorously defined to ensure coordinated mission success. Once the ATO is finalized, it is forwarded to all wings throughout the AOR, so they can prepare their aircraft for execution of the ATO.

The entire ATO process governs the development of "strike" missions [air interdiction, offensive counterair (OCA) surface attacks, etc.] as well as some suppression of enemy air defenses (SEAD) missions for manned aircraft. However, this is only a part of the mission planning that occurs. Unmanned aircraft are also tasked, and the combined schedule for both manned and unmanned aircraft is found in the Integrated Tasking Order (ITO). In addition, reconnaissance missions must also be scheduled. The Joint Integrated Prioritized Collection List (JICPL) is developed in similar fashion to the JIPTL to accomplish this task. Furthermore, the Combat Plans Division must also ensure US and allied assets are adequately defended and that air supremacy (or air superiority) is established. Consequently, the Airspace Control Order (ACO) is developed in accordance with the Air Defense Plan (ADP) to outline how defensive counterair missions will be planned to meet this crucial objective [19]. Finally, command and control (C2) aircraft are often required to monitor mission completion and provide C2 from the battlefield [20]. Many of these mission types draw from the same pool of aircraft or compete for other resources. Allocation decisions in these cases are generally made early on when the JFACC makes his air apportionment recommendations during the Objectives, Effects, and Guidance stage of the ATO cycle [20].

2.1.2 Air Refueling Within a JAOC

"Air refueling (AR) is the in-flight transfer of fuel between tanker and receiver aircraft" [12]. It is crucial as a force multiplier for the US military, allowing aircraft to greatly extend their range and project power globally. Taking on more fuel allows aircraft performing counterair or other air operations to increase their payload, loiter over an area longer, or service a mission from further away. For intratheater refueling, AR permits aircraft to be based beyond the range of enemy threats, yet still fly daily missions in enemy territory [19].

The US military primarily performs air refueling in one of two ways: in an anchor area, or along an AR track. For the anchor method, a tanker is assigned to fly in a racetrack pattern at a specified location. In this case, the tanker refuels receiver aircraft along the racetrack and loiters on location between refueling separate missions. For the AR track method, the tanker meets an aircraft or multiple aircraft along their flight path to their destination (or along a designated track in the area) and refuels the aircraft while continuing along that track. For intratheater AR, the anchor method is normally preferred, because of airspace limitations and the capability for tankers to operate centrally [12].

Air refueling can often be the limiting factor when planning missions. The Combat Plans Division and Air Mobility Division must iteratively match aerial refueling assets to receiver requirements to ensure all missions can be completed. "It is imperative that air refueling planners provide the best match between tanker capabilities and receiver mission requirements in order to maximize overall mission accomplishment" [12]. When establishing how tankers should be apportioned, the most important objective is how the tankers contribute to total campaign success, and generally should be apportioned in a fashion reflecting the relative apportionment of combat aircraft put forth by the JFACC. However, given the sheer variety of missions flown, the various interactions between those missions (such as a C2 aircraft supporting a strike mission) and geographical considerations, scheduling tankers can hardly be considered a straightforward task.

2.1.3 Current Technology in JAOCs

Due to the significant costs associated with operating and maintaining its air assets, the US military has made a considerable investment creating and updating the Air Operations Center Weapons System (AOC WS) to provide technical support to airmen who are planning and executing operational missions worldwide. The AOC WS integrates the command and control elements of each JAOC with the entire organizational infrastructure, providing data synthesis and integrating software and hardware over the entire network. For example, the Master Air Attack Planning Toolkit (MAAPTK) (one component of AOC WS) accesses and updates data from the entire AOC WS network so that combat planners can visualize and generate battle plans with near real-time battle information [25]. This application significantly reduces the amount of time and personnel required to create the appropriate missions. AOC WS provides a very strong framework for these data synthesis exercises, allowing the planner to focus on more sensitive areas of mission planning.

However, one of the weaknesses of AOC WS is its lack of automated tools for strategy development and course of action (COA) planning, particularly in the area of resource allocation. In large part, this automation aversion is well founded, as the complexity involved in scheduling aircraft is staggering. In fact, this complexity dictates that a group of subject matter experts from many backgrounds are required to design and plan missions for most operations. However with large operations, this method for mission planning is very time consuming. Furthermore, it necessitates breaking up mission prioritization and aircraft/munitions allocation, sacrificing some global perspective for the sake of efficiency in the scheduling process. In contrast, the evolution of operations research, particularly in the area of mixed integer optimization, suggests that automated models for scheduling are now feasible, even for very complex problems such as mission planning in a JAOC.

2.1.4 Optimization of Intratheater Mission Planning

As described above, the current process for mission planning within a JAOC is quite decentralized. Planning for many mission types is handled by separate entities working on different documents, despite competing for the same tanker and C2 aircraft. Furthermore, target selection and capability allocation (of aircraft and munitions) occur in a sequential manner. While important for efficiency in the planning process, all of these delegations prevent resources from being used in a manner most conducive to campaign success.

Our proposed optimization model improves force distribution by considering target selection in accordance with capability limitations, and by scheduling the most valuable missions, regardless of type. Furthermore, tankers are scheduled in a manner that best supports the completion of mission objectives and can refuel different mission types on the same flight. The model's framework is flexible enough to account for most mission profiles and can capture the dependency of one mission on another.

However, the model is not designed to be a one-step scheduler. It does not, for example, handle deconfliction of assets traveling to the battlefield. Rather, it is designed as a tool to aid the Combat Plans Division. A planner should run the model based on desired inputs, evaluate its output, and add constraints to implement desired changes, or use the model's output as a starting point for the scheduling process. Using the model iteratively in this way allows a planner to take advantage of its capability to efficiently schedule assets in such a way as to maximize their benefit to the campaign. However, expert knowledge is still necessary for ensuring an ATO can realistically be carried out effectively.

The model in this thesis is designed specifically to provide a schedule for intratheater mission planning with a known quantity of aircraft and tankers. Furthermore, the particular formulation described in this thesis has been primarily tested and evaluated with an eye toward scheduling strike missions from the JIPTL. While the model's framework is quite flexible to the type of mission inputs, the model will perform best if tailored to meet a particular JAOC's needs. With more model complexity, there is greater potential for improvement compared to a schedule created without aid of the model, but the model will require more time to find good solutions. On the other hand, the model can handle narrowly defined problems more quickly, but the benefits will not be as substantial. Balancing this trade-off when determining the best use for the model is crucial and should be handled differently at different JAOCs depending on their particular situations.

2.2 Literature Review

2.2.1 Continuous-Time vs. Discrete-Time Scheduling

When creating MIO formulations which involve scheduling events or processes over a time horizon, one can model the passage of time in one of two ways: discrete-time modeling or continuous-time modeling. In general, discrete-time formulations are much simpler. However, in order to get good approximations, time must often be discretized into small intervals, which can lead to large combinatorial problems which become computationally intractable. Continuous-time formulations are often much more complex, and extra variables are required to keep track of important events. However, these formulations give exactly optimal solutions (not optimal approximations) and do not grow combinatorially depending on process lengths [10].

Discrete-time formulations require breaking apart the time horizon into preselected intervals of uniform length. Variables are easily defined to indicate whether a process begins during a given time period (or is ongoing during that time period), or specify units available (or in use, etc.) during that time period. Consequently, constraints can be written in a very straightforward manner, which leads to relatively simple formulations [11]. Modeling unit availability and accounting for intermediate changes such as deliveries requires no new additional variables or increases in computation time. Furthermore, because of the simplistic nature of the formulations, discrete-time models often solve faster than continuous-time models when sequences are well-defined and fewer time periods are required [18].

On the other hand, when many time periods are required, the size of a discrete-time model grows combinatorially. In order to ensure the optimal solution for the discrete-time model matches that of the continuous-time model, time intervals must be selected to be as small as the greatest common factor of the processing times [10]. Unless the greatest common factor is quite large, one faces a trade-off between the accuracy of the solution and the computation time. In problems where processing times are not consistent, are very small relative to the time horizon, or do not have large common factors, this trade-off can be quite problematic [11].

In continuous-time modeling, variables are introduced to capture event times, either defined globally or for each unit. A number of variables must be created a priori to retain the time slots for when events occur. Also, variables may be required to account for both when an event begins and ends [11]. However, since these variables link additively as opposed to the multiplicative increases of discrete-time variables, continuous-time formulations generally have far fewer variables and constraints in comparison. Notably, they will almost always have fewer integer variables.

Continuous-time formulations do not rely on approximations of event time occurrences, so the optimal solutions found by these formulations will always be greater than or equal to the discrete-time solutions. Computationally the problems are smaller, but given the complicated structure, continuous-time models may perform slower than discrete-time models [11]. Yet in certain scenarios, where many time intervals would be needed or when sequence-dependence can create large changes in process changeover times, continuous-time models will often solve faster than discrete-time models, in addition to giving better solutions [18].

The ATO scheduling problem has many of the characteristics which suggest the efficacy of continuous-time scheduling in comparison to discrete-time scheduling. Mission completion times and travel times vary greatly, necessitating small time intervals to get good approximations for a discrete-time formulation. In addition, missions can be performed in different sequences, and these sequences have very different changeover times between when an aircraft starts one mission and is available to start another. Consequently, we hypothesize that continuous-time modeling suits the ATO scheduling problem much better than discrete-time modeling.

2.2.2 Allocation of Military Assets and Mission Planning

Mission planning in the military is a large field covering a variety of topics. Areas as wide-ranging as airlift transport [16, 26], target selection [8, 6], route planning [24], sensor placement [22], and many more have been researched, using techniques such as mixed integer optimization, dynamic programming [21], approximation algorithms, and even more basic analyses such as theory of constraints [7]. We will focus on literature specific to two areas: mixed integer optimization with regards to allocating assets for "mission completion" and any research that attempts to optimize the production or completion of ATOs.

One area of military mission planning that has been extensively researched, including formulations utilizing mixed integer optimization, is that of UAV (unmanned aerial vehicle) planning, also known as RPA (remotely piloted aircraft) planning. In 2006, Bryant and Sakamoto both submitted theses as part of research for Draper Laboratories investigating the optimization of UAV scheduling [6, 24]. Bryant's research utilizes a modified knapsack formulation to optimize the assignment of missions to UAV squadrons over extended time horizons. He calculates an expected capacity of each squadron to complete missions and allocates missions to the squadrons based on these estimates. Sakamoto's research utilizes a modified vehicle routing problem formulation with time windows to route UAVs between assigned missions. In essence, the two theses complement one another. First, missions are assigned to squadrons. Then, individual UAVs are routed to assigned missions. In addition, both theses account for uncertainty in much of the data using robust optimization, allowing for the generation of schedules that are effective, even within the "fog of war."

However, these UAV planning formulations rely largely on characteristics of UAVs, making the extension of the formulations to manned aircraft planning difficult. First, UAVs normally operate autonomously and can largely be scheduled independently from other aircraft. However, manned aircraft often fly missions in "packages" where many aircraft must fly jointly to accomplish mission objectives. Bryant's and Sakamoto's models do not provide the framework for ensuring many aircraft perform a mission in collaboration. Along these lines, both authors only utilize two UAV types and write constraints based solely on the characteristics of these UAV types. Neither provides the flexibility for extending the formulation to a wide variety of different aircraft performing different missions. Next, UAVs generally have very long ranges and can fly long missions without the need for air refueling. However, manned aircraft often need to refuel while airborne in order to complete their mission objectives. Consequently, accounting for air refueling is a crucial factor when planning missions for manned aircraft. Finally, UAVs generally fly longer missions or multiple missions in succession, warranting a separation of the assignment and routing processes. As loiter times and range decrease (as they do for many manned aircraft), separating the assignment and routing processes can lead to substantial decreases in optimality as compared to a joint assignment/routing process. All of these factors suggest that an effective UAV planning formulation may not translate well to a planning formulation for a variety of manned aircraft.

In 2013, Bertsimas et al. (as an extension to a thesis written by Culver [8]) extended the research of Bryant and Sakamoto toward the planning of reconnaissance operations from both ground forces and UAVs [2]. Their research adds the concepts of "redundancy" and "mixing" whereby multiple types of assets can perform surveillance on a target. Furthermore, the formulation extends the time horizon, giving a more descriptive schedule that combines both mission assignment and routing, and the formulation is more flexible in general. However, the MIO model still does not provide a concept of air refueling, nor does it account for requirements of joint completion of missions by "packages" of different aircraft types.

The problem of optimizing the ATO planning process has been specifically addressed in a variety of settings for over twenty years. In 1994, Briggs developed a theater level combat planning model combining linear optimization approximations to a MIO model and decision trees. In this thesis, Briggs utilizes “air strike packages” consisting of both SEAD and escort aircraft [14]. In addition, his formulation takes into consideration the geography of enemy defenses and provides for some contingency planning based on the weather. However, Briggs’s model generally only considers the allocation of aircraft over a single period and does not account for air refueling. Thus, his formulation should be used as a big picture theater planning tool as opposed to a day-to-day scheduling tool.

In addition, significant research has been done at the Naval Postgraduate School with the goal of optimizing mission planning for the ATO. Castro developed two models that allocate strike assets to missions in packages [9]. His static model can handle packages with different aircraft, but only allocates aircraft to one mission over one time period, similar to Briggs’s model. His dynamic model can schedule multiple missions over a limited time period, but only homogeneous aircraft packages are considered. Both models also handle the assignment of weapons to aircraft. However, neither considers the addition of SEAD or escort aircraft to packages, nor do they introduce air refueling. Consequently, Castro’s models have similar limitations to Briggs’s model.

Also at the Naval Postgraduate School, Zacherl attempted to tackle the problem of re-allocating assets on the ATO given the appearance of high-valued TSTs [27]. His heuristic algorithm was designed to quickly reassign assets to targets when TSTs were realized. However, as a reactionary formulation, the algorithm cannot anticipate TST appearances. Consequently, it acts not as a planning tool, but as an “on-the-job” tool that fits a different niche as compared to the research considered in this thesis.

Given the literature listed above, our research seeks to attack the ATO planning process from another angle. Specifically, we wish to create an ATO “scheduler,” a formulation that describes where aircraft will be at all times throughout the day. In addition, our formulation can handle a large variety of aircraft types in any combination and accounts for air refueling. We do assume that missions will have specific configuration and weapons requirements for the aircraft flying them and that the packages required to complete the mission are preset. Given these limitations, however, our model can assign aircraft to multiple missions in the same day, even when competing for a limited number of tankers. Thus, our model can be a powerful tool for scheduling missions during the early periods of a campaign when the ability to add more missions to the ATO can have profound benefits.

Chapter 3

Modeling Intratheater Operations in a JAOC

3.1 Problem Framework

We define a set $\mathcal{I}$ of available aircraft to complete missions over a set time horizon within a certain theater. In addition, we define a set $\mathcal{T}$ of available tankers assigned specifically to intratheater refueling of those aircraft. All tankers and other aircraft have known fuel consumption rates, fuel capacities, and speeds. Both tankers and other aircraft are located at their home bases at the beginning of the time horizon.

We also have a list of potential "missions" $\mathcal{J}$ which can be flown. A "mission" can be of any type to include strike, air-to-air, SEAD, and reconnaissance. Furthermore, a mission can include multiple strike targets or a sequence of actions (e.g., SEAD followed by a strike). Consequently, missions require a package for completion, where a package is formed of a specific number of aircraft of specified types. Missions are assigned a "value" reflecting their importance toward achieving campaign success. In addition, missions have duration times, indicating the amount of time it takes to complete the mission once the package reaches the mission starting point. Missions also have time windows for completion within the time horizon; if completed, missions must begin within their respective time windows.

To complete a mission, the required package of aircraft fly independently to an anchor refueling area of the set $\mathcal{D}$, where the package meets. (For convenience, we will henceforth refer to anchor refueling areas as anchor tracks, not to be confused with point-to-point refueling tracks.) If one or more aircraft flying the mission require air refueling, a tanker will also meet the package and refuel those aircraft at the anchor point. Then, the package flies to the starting point of the mission and completes the mission. Thereafter, aircraft return to the same anchor point, refuel if necessary, and fly separately back to their home bases.

The goal of the optimization model is to maximize the total "value" of all missions flown. Final output of the model includes a schedule of which missions an aircraft will fly and at what times. It also specifies which missions tankers will refuel and on what anchor tracks the refueling will occur.

3.1.1 Assumptions

We break apart our assumptions into two sets: structural assumptions and realism-limiting assumptions.

Structural Assumptions

These assumptions are necessary in order for us to provide structure to our model. None of these assumptions significantly restrict the solutions of our model. They closely model realistic practices and provide a framework in which to schedule aircraft and tankers to complete their required tasks.

- Every mission can only be completed once.
  - One mission may include multiple passes or require a series of actions. Redundancy in striking a target should be included in the mission description.
- All aircraft and tankers are available for the entire time horizon.
  - Tankers are dedicated to theater support for the entire time horizon.

- Anchor tracks are preselected and not subject to a change in location.
- All air refueling takes place at anchor points.
  - Air refueling occurs before aircraft depart their anchor tracks to complete a mission.
- All tankers can refuel all aircraft.
  - Tankers may not be refueled; they may only refuel other aircraft.

We note that our assumption on air refueling occurring before mission completion is logical if our aircraft only need to be refueled once. We can alternatively assume that tankers must loiter for the duration of a mission they refuel, providing for refueling before and after mission completion. However, attempting to determine whether or not a tanker should loiter during mission completion (or whether multiple tankers should refuel the same aircraft before and after completing a mission) adds computational complexity and is outside the scope of this model.

We also note that it is trivial to write constraints limiting which tankers can refuel which aircraft, if this constraint is constant over the entire time horizon. However, it is outside the scope of this model to limit only combinations of certain aircraft from simultaneously being refueled by one tanker (e.g., limiting boom vs. drogue refueling on any given flight).

Realism-Limiting Assumptions

These assumptions limit the accuracy or optimality of our solutions. Their inclusion is not based on reality, but rather a need to limit model complexity so that computational solve time is manageable. In other words, these assumptions limit the flexibility of the scheduling process. A planner may not be able to schedule a mission within this framework, but could relax these assumptions and still execute a mission in practice. However, incorporating this flexibility in the model severely increases computation time, making the model impractical.

- An aircraft starts at and returns to its home base after completing a mission.
  - Every individual aircraft will only have one home base, although different aircraft may have different home bases.
  - Aircraft CANNOT fly multiple missions in a row without returning to base in between.
- All aircraft packages meet and depart from an anchor point and return to the same anchor point after completing the mission.
- Tankers may only refuel other aircraft at one anchor track per flight.
  - Tankers must return to base in between refueling aircraft at different anchor tracks.
- Air refueling takes a negligible or constant amount of time.
  - Aircraft-specific refueling time CANNOT be accounted for.
  - Fuel loss occurring while loitering at anchor areas CANNOT be accounted for, except for tankers.
- A mission CANNOT have the requirement that multiple tankers refuel the aircraft flying it.
  - One can pre-allocate that multiple specific tankers refuel a mission. However, this limits the flexibility of the model.

The first two sets of assumptions require that aircraft follow the same path flying to a mission start location and returning back to base. Most missions will be completed in this fashion. Should a planner want flexibility in scheduling an aircraft to start and finish at different locations, or follow a different flight path, these missions should be scheduled explicitly.

The tanker-receiver assumptions limit, in particular, the completion of missions requiring very large packages. As very large packages generally require multiple tankers to refuel them and for aircraft to loiter while waiting for other aircraft to receive fuel, these types of missions should also be scheduled explicitly.

3.2 Mixed Integer Optimization Model

3.2.1 Indices

- $\mathcal{I}$: Set of all aircraft i
- $\mathcal{K}$: Set of all aircraft types k
  - $\mathcal{A}_k$: Set of all aircraft i of type k
- $\mathcal{T}$: Set of all tankers t (for air refueling)
- $\mathcal{J}$: Set of all potential missions j that can be flown
- $\mathcal{D}$: Set of all potential anchor tracks d where air refueling can occur
- $\mathcal{Y}$: Set of all flights y taken by a tanker

3.2.2 Decision Variables

- x_{ijd}: Binary; aircraft i engages target j out of track d
- z_{jd}: Binary; target j is engaged with the necessary number of aircraft out of track d
- f_{jd}: Continuous; time at which refueling and/or package meet-up takes place for mission j at track d (if refueling takes place)
- $p_{j_1 j_2}$: Binary; takes on a value of 1 if j_1 is engaged before j_2 (if both are engaged)
- l_{tdy}: Continuous; amount of time tanker t spends at track d during flight y
- h variables: Binary; does tanker t refuel some mission on some track during some flight?

- h^1_{tjdy}: Binary; tanker t refuels at least one aircraft flying mission j on track d during flight y
- h^2_{tdy}: Binary; tanker t refuels some mission on track d during flight y
- h^2_{tj}: Binary; tanker t refuels at least one aircraft flying mission j

- m variables: Continuous; how much fuel does tanker t transfer?
  - m^1_{itj}: Continuous; amount of fuel transferred by tanker t to aircraft i flying mission j
  - m^2_{tjy}: Continuous; amount of fuel transferred by tanker t to all aircraft flying mission j during flight y
  - m^3_{tj}: Continuous; amount of fuel transferred by tanker t to all aircraft flying mission j

3.2.3 Data

Data on mission specifics is largely obtainable through the JIPTL. However, some of the variables, including values, time windows, and mission completion times, require planners to perform some extra work to format the data differently from the inherent data found on the JIPTL. All data on aircraft specifics is obtainable from the Friendly Order of Battle (FrOB) in coordination with MAAPTK. This includes the calculations of speeds, fuel consumption rates, and fuel capacities which depend on the configuration of the aircraft. As aircraft configuration is dependent on the mission flown, these variables can be calculated based on a given aircraft-mission-track triplet. MAAPTK is well-suited to handle these data synthesis exercises.

- V_j: Value of completing mission j
- A_{jk}: Number of aircraft of type k required by target j
- $Q(Q^1_{jd}, Q^2_{jd}, Q^s_j)$: Triplet specifying the window for when a package must leave track d to arrive at target j within the target window

  - Q^1_{jd}: The earliest time at which mission j can leave from track d
  - Q^2_{jd}: The latest time at which mission j can leave from track d
  - Q^s_j: The duration of mission j once on location

- $BASE_i$: Base at which aircraft i resides
- $BASE_t$: Base at which tanker t resides
- $DIST_{BASE,d}$: Distance from a particular base to a particular track
- SP_{ij}: Average speed of aircraft i traveling to and from mission j (varies depending on payload)
- SP_t: Average speed of tanker t
- TT_i: Turn time for aircraft i (amount of time it takes to get aircraft back in the air after landing at base)
- TT_t: Turn time for tanker t (amount of time it takes to get tanker back in the air after landing at base)
- MT_{ijd}: Amount of time it take aircraft i to complete mission j and return to base starting from track d
- R_t: Rate at which tanker t burns fuel
- FA_t: Fuel available for use and transfer by tanker t
- FN_{ijd}: Amount of fuel aircraft i must take on through air refueling when performing mission j out of track d

3.2.4 Formulation

Objective Function

$$max \sum_{j \in \mathcal{J}, d \in \mathcal{D}} V_j \cdot z_{jd} \tag{3.1}$$

The objective of the model is to maximize the net value of all missions flown over the time horizon.

Constraints on aircraft/time window requirements

$$SOS1(z_{jd} : \forall d \in \mathcal{D}) \qquad \forall j \in \mathcal{J} \tag{3.2}$$

$$\sum_{i \in \mathcal{A}_k} x_{ijd} = A_{jk} \cdot z_{jd} \qquad \forall k \in \mathcal{K}, j \in \mathcal{J}, d \in \mathcal{D} \tag{3.3}$$

$$f_{jd} \geq Q^1_{jd} \cdot z_{jd} \qquad \forall j \in \mathcal{J}, d \in \mathcal{D} \tag{3.4}$$

$$f_{jd} \leq Q^2_{jd} \cdot z_{jd} \qquad \forall j \in \mathcal{J}, d \in \mathcal{D} \tag{3.5}$$

These constraints ensure that missions are completed with the right aircraft in the proper time frame. Constraints (3.2) ensure that a mission is only completed out of one track. In other words, a mission can only be completed once. Constraints (3.3) call for the proper number of each type of aircraft to complete mission j, if mission j is flown. Constraints (3.4) and (3.5) dictate that a mission must be flown in the time window necessary for mission success, if flown.

We note that Constraints (3.2) (as well as Constraints (3.8) and (3.9) below) utilize type 1 special ordered sets (SOS) for specifying integrality conditions. These SOS1 constraints require that at most one variable of the set takes on a positive value (since we use these constraints with binary variables, we are ensuring at most one variable in the set can take the value 1). Constraining the variables using SOS1 constraints (as opposed to standard inequality constraints) speeds up the branch and bound process for these variables as branches can occur on sets of variables as opposed to each variable individually [1].

Constraints on the logistics of tanker variables and m/h variable interactions

$$z_{jd} \geq h^1_{tjdy} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J}, d \in \mathcal{D}, y \in \mathcal{Y} \tag{3.6}$$

$$h^2_{tdy} \geq h^1_{tjdy} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J}, d \in \mathcal{D}, y \in \mathcal{Y} \tag{3.7}$$

$$SOS1(h^2_{tdy} : \forall d \in \mathcal{D}) \qquad \forall t \in \mathcal{T}, y \in \mathcal{Y} \tag{3.8}$$

$$SOS1(h^1_{tjdy} : \forall d \in \mathcal{D}, y \in \mathcal{Y}) \qquad \forall t \in \mathcal{T}, j \in \mathcal{J} \tag{3.9}$$

$$h^3_{tj} \leq \sum_{d \in \mathcal{D}, y \in \mathcal{Y}} h^1_{tjdy} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J} \tag{3.10}$$

$$m^3_{tj} \geq \sum_{i \in \mathcal{I}} m^1_{itj} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J} \tag{3.11}$$

$$m^3_{tj} \leq \sum_{y \in \mathcal{Y}} m^2_{tjy} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J} \tag{3.12}$$

These constraints control the interaction between the variables governing tanker flights. Constraints (3.6) ensure tankers only refuel missions that are flown and on the proper track. Constraints (3.7) imply a mission can only be refueled on a track that is open. Constraints (3.8) prevent a tanker from refueling more than one track per flight and constraints (3.9) prevent a tanker from refueling the same mission on different flights or different anchor tracks. Constraints (3.10) impose that a mission must be refueled on at least one track and flight, if refueled. Finally, constraints (3.11) and (3.12) govern the interaction of fuel transfer variables.

Travel time constraints

$$x_{ij_1d_1} + x_{ij_2d_2} \leq 1 + p_{j_1j_2} + p_{j_2j_1} \qquad \forall i \in \mathcal{I}, j_1, j_2 \in \mathcal{J}, d_1, d_2 \in \mathcal{D} \tag{3.13}$$

$$p_{j_1j_2} + p_{j_2j_1} \leq 1 \qquad \forall j_1, j_2 \in \mathcal{J} \tag{3.14}$$

$$f_{j_2d_2} - f_{j_1d_1} \geq MT_{ij_1d_1} + TT_i + \frac{DIST_{BASE_i,d_2}}{SP_{ij_2}} - M \cdot (3 - x_{ij_1d_1} - x_{ij_2d_2} - p_{j_1j_2})$$
$$\forall j_1, j_2 \in \mathcal{J} : j_1 \neq j_2, d_1, d_2 \in \mathcal{D}, i \in \mathcal{I} \tag{3.15}$$

$$f_{j_2d_2} - f_{j_1d_1} \geq \frac{DIST_{BASE_t,d_1}}{SP_t} + TT_t + \frac{DIST_{BASE_t,d_2}}{SP_t} - M \cdot (2 - h^1_{tj_1d_1y_1} - h^1_{tj_2d_2y_2})$$
$$\forall j_1, j_2 \in \mathcal{J} : j_1 \neq j_2, d_1, d_2 \in \mathcal{D}, t \in \mathcal{T}, y_1, y_2 \in \mathcal{Y} : y_1 < y_2 \tag{3.16}$$

Constraints in this section ensure travel time between bases, anchor tracks, and mission locations are observed for aircraft flying multiple missions and for tankers performing multiple flights. Constraints (3.13) and (3.14) define "ordering precedence;" if an aircraft performs multiple missions, one must have "order precedence." Constraints (3.15) do not allow an aircraft to engage target j_2, the latter target, until it has finished traveling back to its home base and been refueled (and been re-outfitted with weapons, if necessary) following the completion of its former target j_1. Constraints (3.16) do not allow a tanker to refuel missions on different flights unless it has finished traveling to and from its home base and been refueled. Here the concept of tanker "flights" are realized. A tanker may refuel many different missions on the same flight without returning to base, so travel time restrictions only apply to missions refueled by the same tanker on different "flights."

Fuel constraints

$$M \cdot h^3_{tj} \geq m^3_{tj} \qquad \forall t \in \mathcal{T}, j \in \mathcal{J} \tag{3.17}$$

$$l_{tdy} \geq f_{j_1 d} - f_{j_2 d} - M \cdot (2 - h^1_{tj_1 dy} - h^1_{tj_2 dy}) \quad \forall t \in \mathcal{T}, j_1, j_2 \in \mathcal{J} : j_1 \neq j_2, d \in \mathcal{D}, y \in \mathcal{Y} \tag{3.18}$$

$$R_t \cdot l_{tdy} + \sum_{j \in \mathcal{J}} m^2_{tjy} + 2R_t \cdot \frac{DIST_{BASE_{t,d}}}{SP_t} \cdot h^2_{tdy} \leq FA_t \quad \forall t \in \mathcal{T}, d \in \mathcal{D}, y \in \mathcal{Y} \tag{3.19}$$

$$\sum_{t \in \mathcal{T}} m^1_{itj} \geq FN_{ijd} \cdot x_{ijd} \qquad \forall i \in \mathcal{I}, j \in \mathcal{J}, d \in \mathcal{D} \tag{3.20}$$

These constraints ensure all aircraft and tankers have enough fuel to complete their missions. Constraints (3.17) only permit a tanker to offload fuel to a mission to which it is assigned. Constraints (3.18) calculate the amount of time a tanker spends at a particular track during a particular flight. Constraints (3.19) provide that the sum of fuel used by a tanker loitering and transferred to other aircraft does not exceed its capacity after accounting for travel from its home base. Constraints (3.20) ensure an aircraft receives enough fuel through air refueling to complete its mission and return to its home base safely.

3.2.5 Optional Constraints

In addition to the formulation above, certain optional constraints can easily be added to reflect the desires of mission planners. This section highlights just some of the ways extra constraints can be added to the model. For ease in describing these constraints, we use the following simplifications of variables:

$$z_j = \sum_{d \in \mathcal{D}} z_{jd} \qquad \forall j \in \mathcal{J} \tag{3.21}$$

$$f_j = \sum_{f \in \mathcal{D}} f_{jd} \qquad \forall j \in \mathcal{J} \tag{3.22}$$

$$x_{ij} = \sum_{d \in \mathcal{D}} x_{ijd} \qquad \forall i \in \mathcal{I}, j \in \mathcal{J} \tag{3.23}$$

If optional constraints are implemented, these variables can be defined, or the summation can be substituted for the variables on the left-hand side.

Constraints on relative time gaps

$$z_{j_1} \geq z_{j_2} \tag{3.24}$$

$$f_{j_1} \leq f_{j_2} - \alpha \cdot z_{j_1} + M \cdot (1 - z_{j_2}) \tag{3.25}$$

$$z_{j_1} = z_{j_2} \tag{3.26}$$

$$f_{j_1} \leq f_{j_2} - \alpha \cdot z_{j_1} \tag{3.27}$$

$$f_{j_1} \geq f_{j_2} - \beta \cdot z_{j_1} \tag{3.28}$$

These constraints allow for the creation of relative time windows for multiple missions that must be coordinated. If two missions must be synchronized, these constraints provide the framework for that requirement. Used together, constraints (3.24) and (3.25) can be combined to create an order precedence of mission j_1 over mission j_2. Specifically, mission j_2 can only be performed if mission j_1 is performed. Furthermore, mission j_2 must not begin sooner than α hours (or whatever time unit is being used) after the beginning of mission j_1. Similarly, constraints (3.26), (3.27), and (3.28) can be combined to create an joint dependency of missions j_1 and j_2 on one another. Specifically, if mission j_1 is performed, then mission j_2 must also be performed, and vice versa. Also, mission j_2 must begin between α and β hours (time units) after the beginning of mission j_2.

Constraints on aircraft performing certain missions

$$x_{ij_1} = x_{ij_2} = ... = x_{ij_n} \tag{3.29}$$

$$x_{i_1 j} = x_{i_2 j} = ... = x_{i_n j} = (z_j) \tag{3.30}$$

These constraints give mission planners the ability to require aircraft to fly certain combinations of missions or create specific types of packages. Constraints (3.29) ensure that if an aircraft performs mission j_1, it also executes mission j_2, j_3, ..., j_n. A planner could use these constraints to ensure one aircraft handles an entire subset of missions. Constraints (3.30) require a mission to be performed by a set of aircraft if aircraft i_1 performs the mission. By adding the last equality, these constraints would designate certain aircraft must perform mission j, if performed. Thus, a planner could ensure a specific squadron or set of aircraft work together to execute a particular mission.

Constraints on mission subsets

$$z_{j_1} = z_{j_2} = ... = z_{j_n} \tag{3.31}$$

$$z_{j_1} + z_{j_2} + ... + z_{j_n} \leq 1 \tag{3.32}$$

These constraints allow mission planners to put restrictions on certain subsets of missions. Constraints (3.31) require an entire subset of missions to be performed, or none of the subset. If certain missions only have value when performed on the same day as others, these constraints capture that effect. On the other hand, constraint (3.32) only allows one mission to be flown from a subset of missions. This constraint gives the planner more flexibility in the scheduling process, by allowing the model to select how targets should be combined when flying missions. For example, two missions could both include bombing an enemy command and control center, but have different secondary targets. Adding this constraint would let the model select the mission which provides the most “value” and fits the daily schedule without redundantly attacking the command and control center more than once.

Constraints on munitions

We let $\mathcal{U}$ be the set of all munitions u where there is a limited supply (and $\mathcal{B}$ be the set of all bases b). We introduce new data MUN_{ju} (or MUN_{iju}) and $CAPM_u$ (or $CAPM_{bu}$). MUN_{ju} represents the number of munitions of type u needed to complete mission j (and MUN_{iju} represents the number of munitions of type u needed by aircraft i to complete mission j). $CAPM_u$ represents the total number of munitions of type u that can be allocated for use over the time horizon (and $CAPM_{bu}$ represents the number of munitions of type u and located at base b that can be used over the time horizon). Then we can define the following constraints.

$$\sum_{j \in \mathcal{J}} MUN_{ju} \cdot z_j \leq CAPM_u \qquad \forall u \in \mathcal{U} \tag{3.33}$$

$$\sum_{i \in \mathcal{I}: BASE_i = b, j \in \mathcal{J}} MUN_{iju} \cdot x_{ij} \leq CAPM_{bu} \qquad \forall b \in \mathcal{B}, u \in \mathcal{U} \tag{3.34}$$

These constraints allow mission planners to take into account limited munitions. Constraints (3.33) ensure that the total number of munitions used of type u does not exceed the number of munitions of type u available. Constraints (3.34) also takes into account the location of munitions, ensuring that the total number of munitions used by aircraft with home base b does not exceed the number of munitions available at that base.

3.3 Greedy Algorithm

We also consider a greedy algorithm to approximate how an intelligent human planner might create an ATO schedule. Specifically with strike missions, we assume the planner schedules the most important mission off of the JIPTL first. Then, he works his way down the JIPTL, attempting to schedule every mission with the remaining aircraft. If a mission cannot be scheduled with any aircraft, the planner skips it. Otherwise, he attempts to schedule every mission on the list or until all aircraft are operating throughout the entire ATO day.

This algorithm attempts to replicate and automate the process above, while intelligently scheduling tankers to refuel multiple missions on the same flight if the missions can be scheduled on the same anchor track around the same time frame. We believe that this algorithm represents a good baseline for mission planning, as it benchmarks the approximate performance an expert human planner can obtain. The greedy algorithm is outlined in the steps below.

1. Order all potential missions from most to least important. Also calculate distance between aircraft/tanker bases and the potential mission locations, as well as between anchor points and the mission locations. These distances will be ordered for each mission to determine precedence of anchor points, aircraft, and tankers when scheduling that mission.

2. Loop FOR all missions, starting with the most important. If no more missions exist, EXIT ALGORITHM.

3. Loop FOR all potential times at which the mission can be scheduled, starting at the earliest. This includes the beginning of the mission time window, one or more random times in the time window, and the end of the time window. If no more times exist, the mission cannot be scheduled. Skip to the next mission.

4. Loop FOR all potential anchor tracks, starting with the anchor track closest to the mission starting location. If no more anchor tracks exist, the mission cannot be scheduled at this time. Skip to the next potential mission start time.

5. Loop FOR all aircraft types required to complete the current mission. IF no aircraft of that type are required, skip to the next type.

6. Loop FOR all aircraft of the selected type, starting with the closest aircraft to the mission start location. If no more aircraft of the proper type exist and

enough aircraft have not been assigned of that type to complete the mission, the mission cannot be scheduled out of this anchor track at this time. Remove the busy periods and fuel required of all aircraft previously assigned to the mission. Skip to the next anchor track.

7. IF an aircraft cannot hold enough fuel to complete the mission from the selected track, skip to the next aircraft. ELSE check to see if the aircraft is available to complete the mission at the selected time, accounting for travel time to and from the mission starting location, as well as mission completion time.

   (a) IF the aircraft is available, schedule the aircraft for the mission at that time. Record new busy times for the aircraft, as well as any increase in fuel that will be required to refuel the scheduled aircraft. If there are now the requisite number of aircraft scheduled for the mission of this type, skip to the next aircraft type. Otherwise move to the next aircraft on the list.

   (b) ELSE the aircraft is not available. Move to the next aircraft on the list.

8. Loop FOR all tankers, starting from the tanker closest to the currently selected track. Check to see if any tanker is already scheduled to fly a mission from the selected track within some time (e.g., 2.5 hours) of the currently proposed mission start time. If no more tankers exist, skip to next tanker loop.

   (a) IF such a tanker exists, check to see if it has enough fuel on board to loiter between missions and refuel the next mission. Also check to see if extending the flight time of the tanker will interfere with the tanker's other scheduled flights. If the tanker is available and has enough fuel on board, assign the tanker to the mission. Record the extended busy time for the tanker and subtract required fuel from the tanker's fuel allotment for that flight. MISSION SCHEDULED. Otherwise, proceed to next tanker.

9. Loop FOR all tankers, starting from the tanker closest to the currently scheduled track. If no more tankers exist and none have been assigned to the mission, the mission cannot be scheduled out of this anchor track at this time. Remove the

busy periods and fuel required of all aircraft previously assigned to the mission. Skip to the next anchor track.

10. Check to see if the tanker is available to complete the mission at the selected time on a new flight, accounting for travel time to and from the currently selected anchor track and the anchor tracks where the tanker refuels aircraft on other flights.

    (a) IF the tanker is available, schedule a new flight for the tanker to refuel the mission at that time. Record new busy times for the tanker and subtract required fuel from tanker's fuel allotment for that flight. MISSION SCHEDULED.

    (b) ELSE the tanker is unavailable. Move to the next tanker on the list.

In comparing the greedy algorithm to the MIO model above, we note that the greedy algorithm always produces a feasible solution to the MIO model, if no optional constraints are added. In fact, the greedy algorithm often produces very good solutions to the MIO model. Since the greedy algorithm solves in seconds, we can use it as a warm start to the MIO model. A comparison of the solutions produced by the greedy algorithm and by the MIO model can be found in the results section of this thesis.

We note that the greedy algorithm may not provide a feasible solution if optional constraints are added. While the greedy algorithm can be modified to account for some of the optional constraints, these modifications are not trivial. In particular, modifying the algorithm to account for equality constraints, such as the requirement that a single aircraft fly multiple missions, can be quite difficult in the context of the greedy algorithm.

Chapter 4

Computational Results

In this chapter, we wish to examine the effectiveness of our model in creating ATO schedules given a set of available aircraft (including tankers) and a potential mission list. First, we display an example scenario, showing the exact data provided to the model, as well as the corresponding schedules created by both the MIO model and the greedy algorithm. This comparison informs our analyses of the trade-offs between the two models. Then we present some computational run-time comparisons of the models. In addition, we consider a hybrid which combines the two algorithms and modifies them to achieve a balance between solution value (i.e., how close our solution is to optimal) and solve time.

4.1 Example Scenario with Results

4.1.1 Inputs

We consider a scenario where the military has designated a set of aircraft to perform "strike" missions (i.e., taking offensive actions in support of campaign objectives) and to provide air support to friendly forces on the ground in enemy territory. These aircraft may be tasked to perform missions ranging from offensive counterair (OCA) (such as attacking enemy aircraft/missiles at their source) to strategic attack and air interdiction (such as attacking an enemy's command and control center) to close air support (CAS) (such as providing supporting fire to an Army unit engaged in a firefight) [13]. In short, all missions performed in this scenario would fall under the core functions of obtaining/maintaining air superiority or global precision attack, but excluding defensive counterair (DCA) and airspace control missions. (For further descriptions of mission types, please reference [13]. This document discusses the basic doctrine of Air Force mission planning.)

We have 20 aircraft and 4 tankers assigned specifically to perform these missions over a 24-hour time horizon. To accomplish these missions, aircraft must fly in "packages" of different aircraft types. To specify aircraft functionality, the US military has developed designations which are used when naming air-frames. For example, "F" stands for fighter aircraft, "A" stands for ground attack aircraft, "B" stands for bombers, and "E" stands for aircraft with special electronic equipment. However, specifying aircraft types solely by designation is generally not sufficient, as many aircraft of the same designation perform vastly different functions.

Consequently we specify our aircraft types for this scenario based upon a mix of aircraft designations and aircraft functionality. Loosely following "Air Force Doctrine Document 1," we classify our 20 aircraft as one of five types: strike fighters (strike aircraft with an "F" designation), attack aircraft (ground attack aircraft with an "A" designation), air-to-air fighters (aircraft with an "F" designation which generally perform DCA or airspace control missions, to be used as escorts), bombers (aircraft with a "B" designation), and SEAD aircraft (aircraft with an "F" or "E" designation which have jamming equipment on board for the suppression of enemy air defenses) [13]. Air-to-air and SEAD aircraft are only being considered for OCA support roles in this scenario (such as OCA-Sweep and OCA-SEAD) (see [13] for explanation). Furthermore, all tankers are tasked specifically to refuel these OCA missions for the entire time horizon. All aircraft and tanker specifications can be found in Table 4.1 and Table 4.2, respectively. (Note all specifications are unclassified estimates and do not in any way represent the actual specifications of the aircraft or tankers.)

All aircraft and tankers begin the time horizon at their home bases outside enemy territory. In addition, all anchor track locations have been preselected just outside

Aircraft	Type	Base	Speed (knots)	Fuel Capacity (1000lbs)	Burn Rate (1000lbs/hr)	Turn time (hrs)
1	1	1	420	5.3	2.1	1.0
2	1	1	420	5.3	2.1	1.0
3	1	1	420	5.3	2.1	1.0
4	1	2	420	20.8	6.6	1.5
5	1	2	420	20.8	6.6	1.5
6	1	3	420	5.3	2.1	1.0
7	1	5	420	20.8	6.6	1.5
8	1	5	420	20.8	6.6	1.5
9	1	6	420	5.3	2.1	0.5
10	1	6	420	5.3	2.1	0.5
11	1	6	420	5.3	2.1	0.5
12	1	8	420	6.9	3.2	1.0
13	1	8	420	6.9	3.2	1.0
14	2	4	340	9.0	2.4	1.5
15	2	4	340	9.0	2.4	1.5
16	2	4	340	9.0	2.4	1.5
17	3	5	420	12.5	7.0	1.5
18	3	7	420	12.5	5.4	2.0
19	4	3	450	260.0	12.0	2.5
20	5	7	420	5.3	2.1	1.0

Table 4.1: Notional Aircraft Specifications

Tanker	Base	Speed (knots)	Fuel Capacity (1000lbs)	Burn Rate (1000lbs/hr)	Turn time (hrs)
1	3	480	340.0	22.3	2.0
2	3	480	340.0	22.3	2.0
3	6	400	180.0	8.1	2.0
4	6	400	180.0	8.1	2.0

Table 4.2: Notional Tanker Specifications

enemy territory. We also assume all aircraft, to include tankers, may travel outside of enemy territory without an escort (i.e., air supremacy has been established outside of enemy territory). All strike missions begin and end at specific locations within enemy territory. Locations for this scenario can be found in Figure 4-1 (with the rectangle indicating enemy territory).

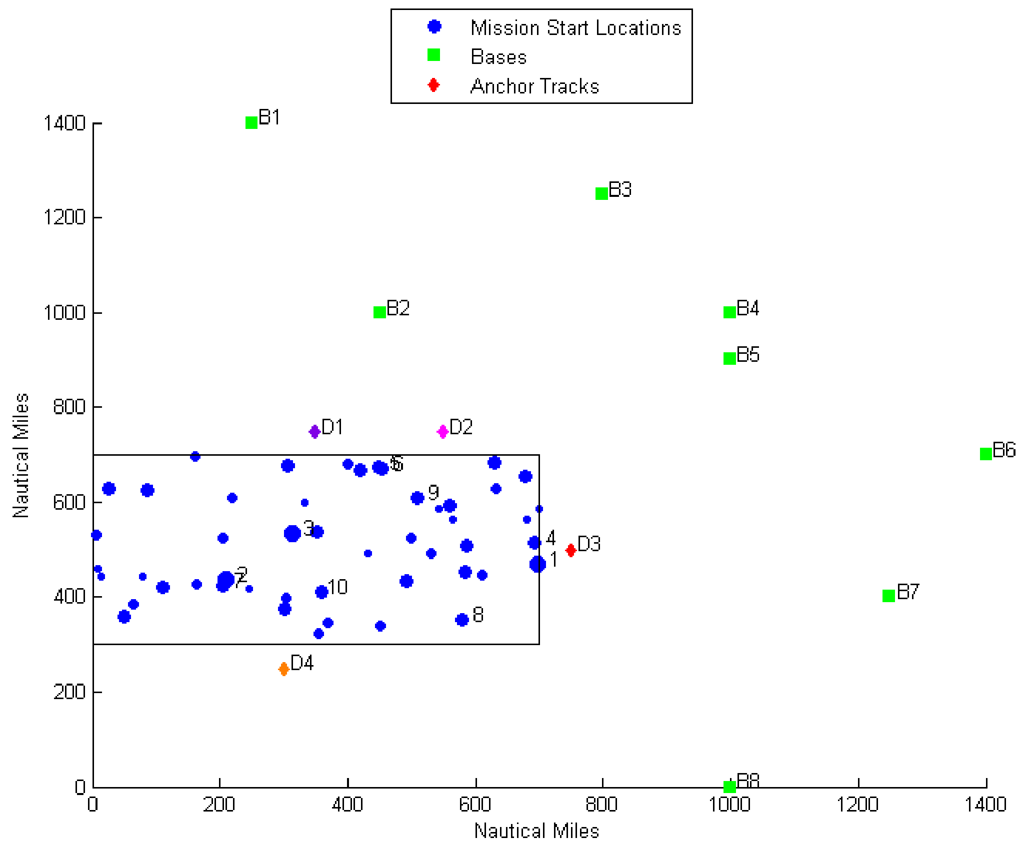

Figure 4-1: Scenario Locations

The missions that can potentially be completed over the 24-hour period have characteristics designed to mirror those found on the JIPTL. Specifically, missions may require a certain number of fighters, attack aircraft, and bombers, and may require SEAD support, and/or an air-to-air escort. In addition, missions have varying durations that could depend on a variety of factors. Some of these factors include the requirement for a dependent sequence of actions (e.g., SEAD followed by a strike) or a mission-specific number of bombing passes. Also, missions must be completed in particular time windows. These windows could reflect the need for a mission to be completed in the dark (or during the day), to match scheduled efforts of ground forces, or to honor Rules of Engagement, among other things. Finally, each mission has an associated value that accounts for the importance of the mission. All mission characteristics for this scenario are summarized in Table 4.3. Note that the drop-off in values of missions is strongly non-linear (in this case, piece-wise linear). This drop-off ensures that the most important missions are far more valuable than missions lower on the JIPTL. If a planner is more interested in how many missions can be scheduled (as opposed to the importance of those missions) assigning mission values that are very similar or decrease approximately linearly will allow the model to make more trade-offs and schedule more missions at the expense of a few higher valued missions.

In this scenario, we made many basic assumptions about distances required for travel, aircraft speeds, fuel consumption rates, and turn times. Specifically, we assumed all distances are strictly point-to-point (i.e., no detours are required for deconfliction or to avoid certain areas). Aircraft speeds are input as a constant estimate of medium altitude cruise speeds based on aircraft type. Fuel consumption rates are also input as constants reflecting estimated burn rates at a medium altitude cruise. Turn times (i.e., the amount of time it takes to get an aircraft back in the air after landing at base) are assumed to be constant for a given aircraft throughout the time horizon.

All of this information is passed on to the model that is coded in the programming language Julia using the domain-specific modeling language JuMP [4, 17]. The model is passed to the solver Gurobi, which solves the model and sends the output back through Julia [15]. All tests in this thesis are carried out on a Intel Xeon E5687W (3.1 GHz) using up to 8 cores and 64 GB of RAM.

Observations on the flexibility of inputs

It is worth noting that the data input for aircraft travel times and fuel consumption can be defined in any way, provided the data entries are constant for any given

		Mission Time Windows		Amount of Time	Number of Aircraft Required				
Mission	Value	Earliest Start Time	Latest Start Time	on Station Required	Strike-F	Attack-A	Air-to-Air	Bombers-B	SEAD
1	9713	11.96	15.53	0.19	4	0	1	0	0
2	9573	1.00	23.00	0.18	2	0	0	0	1
3	9525	1.00	23.00	0.63	2	0	0	0	0
4	9328	11.39	13.81	0.64	4	0	1	1	1
5	9274	18.25	18.62	0.96	0	1	0	0	0
6	8911	17.19	19.59	0.93	2	0	0	0	1
7	8719	1.00	23.00	0.21	2	0	0	0	1
8	8511	5.59	9.90	0.55	2	0	0	0	0
9	7736	13.75	17.15	0.52	2	0	1	0	0
10	7357	4.07	4.87	0.60	0	1	0	0	0
11	6893	4.83	5.04	1.39	0	1	1	0	0
12	6111	16.04	16.87	1.34	0	1	0	0	0
13	5989	4.76	5.46	0.67	0	1	0	0	0
14	5398	18.55	23.42	0.31	0	0	1	1	0
15	3672	1.00	23.00	0.59	2	0	0	0	0
16	3314	21.22	23.50	0.41	1	0	0	0	0
17	2836	19.49	23.50	0.50	2	0	0	0	0
18	2558	1.00	23.00	0.74	2	0	0	0	0
19	2424	1.00	23.00	0.87	2	0	0	0	0
20	2056	10.48	13.05	0.73	4	0	1	0	0
21	1929	1.00	23.00	0.51	2	0	0	0	0
22	1294	10.84	14.33	0.81	1	0	0	0	0
23	989	7.79	9.79	0.43	4	0	1	0	0
24	982	10.17	15.48	0.32	1	0	0	0	0
25	921	2.56	6.36	0.39	4	0	1	0	0
26	920	15.37	16.14	1.36	1	1	0	0	0
27	892	1.12	2.01	1.24	0	1	1	0	0
28	878	6.88	11.89	0.44	2	0	0	0	0
29	744	1.00	23.00	0.91	2	0	0	0	0
30	706	2.50	4.94	0.81	2	0	0	0	0
31	679	9.56	14.23	0.80	2	0	0	0	0
32	528	10.47	10.71	0.80	0	1	0	0	0
33	506	19.76	23.50	0.33	2	0	0	0	0
34	299	20.27	23.50	0.50	4	0	1	0	0
35	163	16.56	21.74	0.44	1	0	0	0	0
36	104	19.94	22.64	0.88	1	0	0	0	0
37	98	1.00	23.00	0.85	2	0	0	1	0
38	95	15.03	15.68	1.31	0	1	0	0	0
39	93	15.65	19.02	0.64	2	0	1	0	0
40	86	3.32	8.69	0.57	2	0	0	0	0
41	83	1.00	23.00	0.20	1	0	0	0	0
42	76	8.31	12.58	0.81	4	0	1	0	0
43	73	1.00	23.00	0.73	1	0	0	0	0
44	51	1.00	23.00	0.15	2	0	0	0	0
45	41	14.97	17.37	0.38	2	0	0	1	1
46	39	21.31	21.44	0.75	0	1	0	0	0
47	33	8.61	9.51	1.12	0	2	0	0	0
48	32	4.52	8.23	0.28	4	0	1	0	0
49	10	13.52	14.20	1.48	1	1	0	0	0
50	10	3.21	8.89	0.27	2	0	0	0	0

Table 4.3: Mission Characteristics

mission-aircraft-anchor track triplet. For example, one can change the fuel consumption rate for an aircraft depending on what mission it will be flying to account for the munitions it will have on board. Cruise altitudes can affect fuel consumption rates and average speeds; these differences can be accounted for on a mission-specific basis. One could also account for increased travel times resulting from the need to avoid certain airspace when flying to an anchor area or mission start location. However, all fuel consumption rates and travel times must be constant regardless of which combination of missions are flown or at what time the missions are flown. Given these limitations, modifying these parameters will have little to no effect on the speed of the models. As MAAPTK already organizes these data parameters, syncing its capabilities with this thesis's optimization model would allow a mission planner to capture all the intricacies involved in the planning process while still utilizing computer-based optimization.

Another area where the user has flexibility is in defining aircraft types. Aircraft "types" in this thesis merely reference the categorization of a particular air-frame. These categorizations can include designation-based categorizations or a breakdown depending on aircraft functionality. In this scenario, we gave each aircraft only one "type." However, it is possible to give aircraft many possible types. For example, one could define the types of "F-15E" and "strike fighters" and assign all F-15Es to both types. Then, if a mission called for an F-15E in particular, one could require the former aircraft type to complete the mission. If a mission had more flexible requirements for the aircraft type, one could require the latter type. This option gives the planner more insight into the trade-off of using an aircraft for a particular mission versus saving it for other missions. Giving an aircraft more than one type does increase computational complexity, but only marginally so. Consequently, as long as the number of aircraft types are "reasonably managed," classifying aircraft into multiple types can be a worthwhile planning option.

Observations on tanker refueling requirements

For this scenario, tankers are required to be present only when aircraft meet at an anchor track to begin flying a mission. In addition, we do not allocate any time to refueling, but assume the refueling takes place in negligible time. Clearly these assumptions are inaccurate. We utilize these assumptions only to manage problem complexity, so as to provide consistency when measuring other model characteristics. As this model is not designed to be all-encompassing, but merely a proof of concept for ATO mission scheduling, we believe these simplifications are reasonable within the model scope. Instead, we discuss the ways tanker complexity can be handled in the remainder of this section.

With regards to air refueling time, it is possible to allocate a constant amount of time to air refueling for every mission. For example, one can allocate thirty minutes for refueling a mission that requires two strike fighters and two bombers and ten minutes for a mission that requires just two strike fighters. Then, it is straightforward to account for this extra time under the "mission completion" section for the mission-oriented aircraft. Also, one can simply require that tankers remain at the anchor track for that time period, and the tankers can be prevented from refueling other missions at that time. However, air refueling times cannot be altered to account for on-load/off-load rates of individual aircraft or tankers. In addition, it is very difficult to account for fuel loss experienced by aircraft loitering at the anchor track while waiting for other aircraft to be refueled. Allowing multiple tankers to refuel the same mission to decrease refueling time is often used in practice, but such a solution falls outside the scope of this thesis (and is incompatible with the greedy algorithm in its current form) [12]. Solving this fuel-loss loitering problem (or the individual on-load/off-load rate problem) requires a significant increase in computational complexity within the modeling process and is probably better handled at the individual mission-planning level.

With regards to managing when air refueling should occur, we have some flexibility. Within our anchor track framework, we want to restrict air refueling locations to our anchor tracks. Thus, we can refuel aircraft before a mission begins, after a mission ends, both, or neither. Given the variability involved with fuel consumption, we certainly want to refuel an aircraft before a mission if it will require air refueling. However, certain aircraft may require refueling both before and after performing a mission. It is straightforward to require any tanker refueling a mission to loiter at an anchor track for the duration of the mission. It is also possible to require tankers to loiter at an anchor track only if there is an aircraft flying the mission that requires air refueling twice. However, the latter option significantly increases the computational complexity of the MIO model (although it is compatible with the greedy algorithm). One might also wish to consider the possibility of having different tankers refuel aircraft before and after the completion of a mission. However, this concept is much more difficult to capture and would require a significant increase in computational complexity.

4.1.2 Outputs

At the conclusion of the algorithm, the model outputs which missions are performed by which aircraft at what times on what tracks. This information is also output for tankers, as well as "flights" of the particular tankers. This output, in addition to previously defined travel and turn times, provides all the necessary data to create a schedule of where aircraft will be for the entire time horizon. Schedules for both the greedy algorithm and MIO model are shown in Figures 4-2 through 4-5.

In comparing the two models, we can analyze certain characteristics that the models display. For example, we notice that the greedy algorithm tends to schedule missions at three times: the beginning of the time period, the end of the time period, and at the very middle of the time period, with a few exceptions. This behavior is a consequence of the algorithm's structure. The algorithm tries to schedule missions first at the beginning of the mission's time window, then some random time in the middle, and finally at the end of the time window. However, this structure does spawn a few beneficial attributes. First, we note that tankers are able to refuel many missions simultaneously, while only having to loiter for short amounts of time.

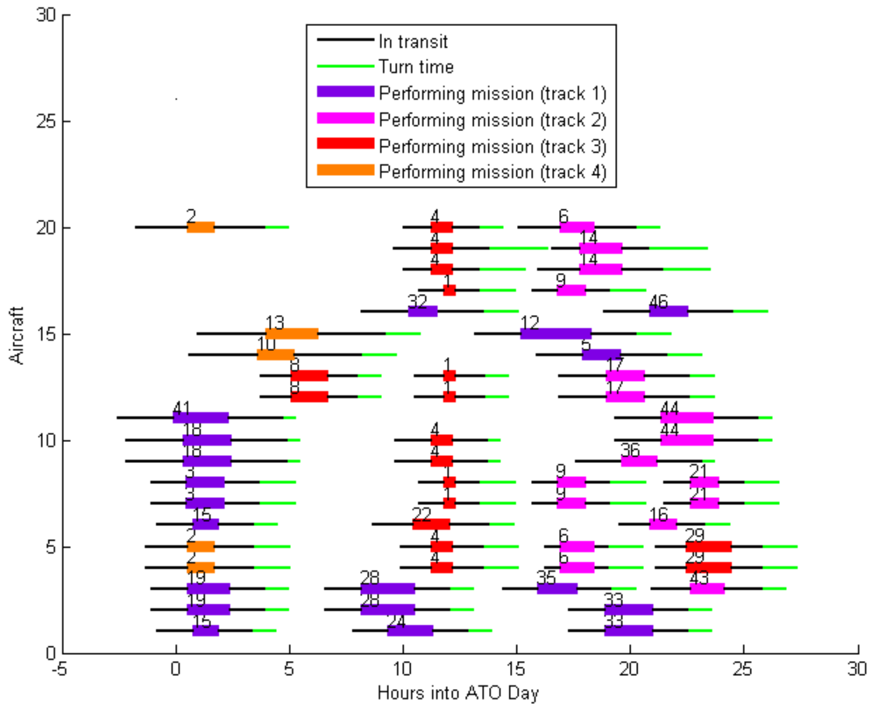

Figure 4-2: Greedy Algorithm's Aircraft Schedule

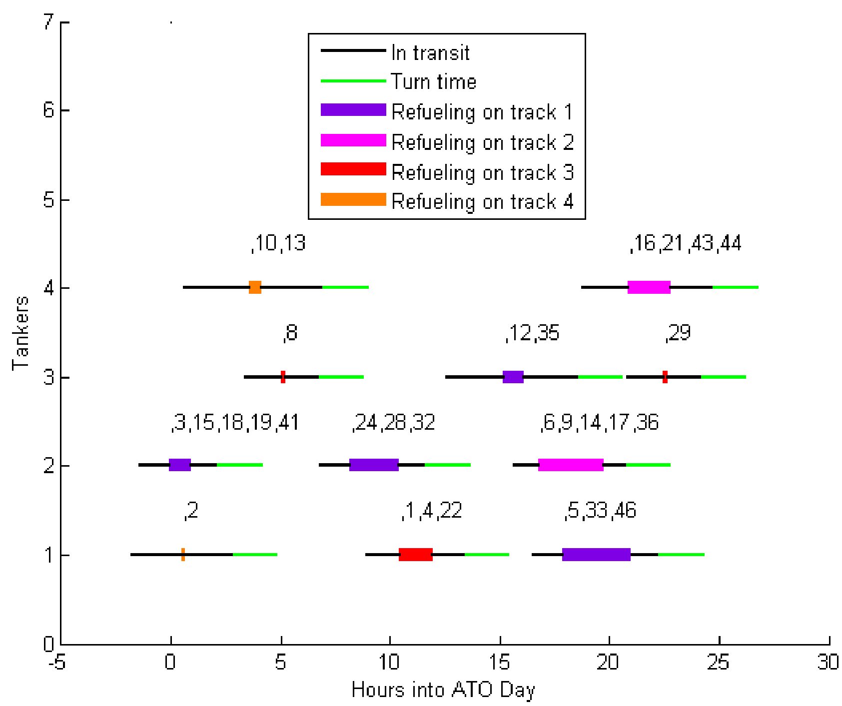

Figure 4-3: Greedy Algorithm's Tanker Schedule

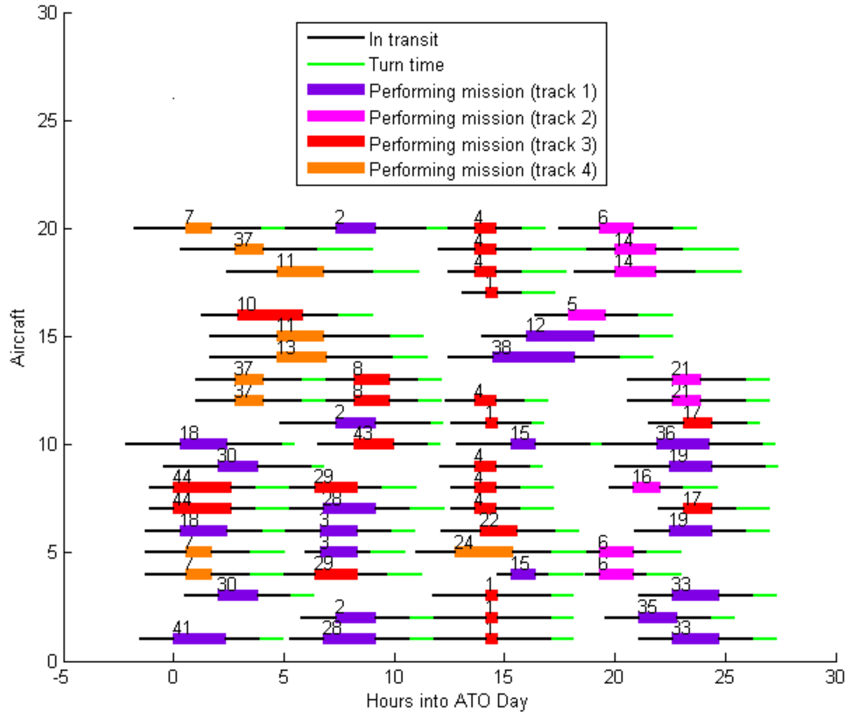

Figure 4-4: MIO Model's Aircraft Schedule

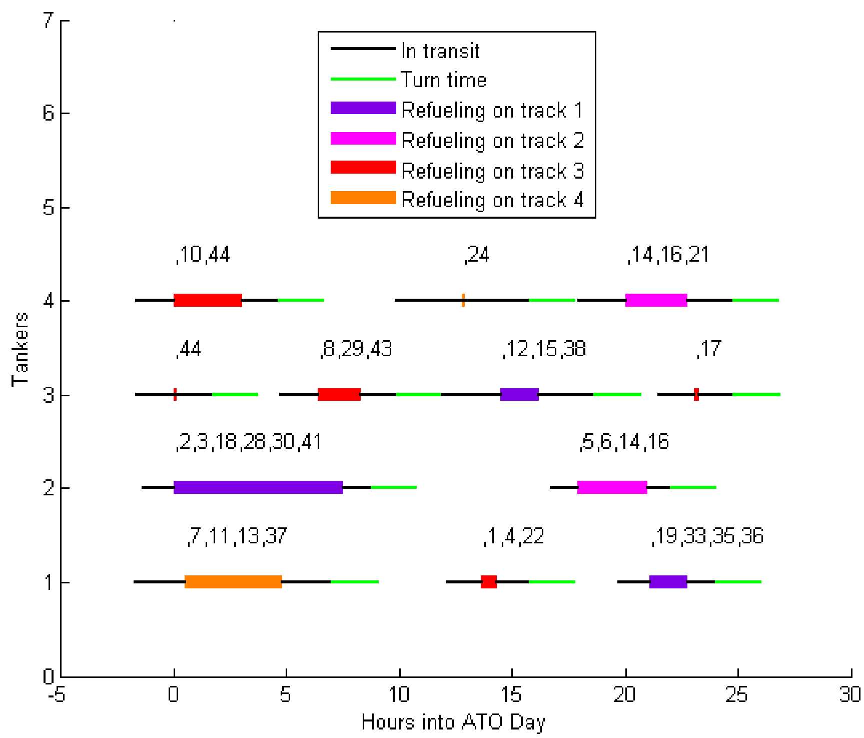

Figure 4-5: MIO Model's Tanker Schedule

Also, most of the important missions are scheduled at the beginning of their time windows (if scheduled). In this scenario, missions one through five and missions eight through ten are all scheduled at the very beginning of their time windows. This feature provides some protection against the unforeseeable. Even if unanticipated events cause some delay to the schedule during execution, there should be enough slack to ensure the most important missions can still be accomplished.

The MIO model, on the other hand, schedules missions more uniformly throughout the entire ATO day. Aircraft and tankers are often scheduled to perform multiple missions immediately in succession following their turnovers at their home bases. While this does provide a lesser margin for error, more missions can be scheduled, and more valuable missions can be substituted for less valuable ones. In addition, the uniform distribution of missions throughout the day spreads out the competition for resources required for aircraft and their pilots, such as runway use, airspace around the base, and even cafeteria busy periods. A comparison of missions flown depending on the model chosen is shown in Table 4.4.

The MIO model yields a total value of 127812 in comparison to the value of 119604 found by the greedy algorithm. This difference is primarily realized in the fact that the MIO schedule substitutes the completion of both missions 7 and 11 for the greedy algorithm's completion of mission 9. In addition, the MIO schedule has a total of 32 missions as opposed to the 30 missions generated by the greedy schedule. Thus, we can see the MIO model creates a schedule that performs both more valuable missions and more missions in general.

A solution of 127812 represents a solution which is 8.64 percent removed from the provable solution bound of 139856. In order to obtain this solution, we warm-started our MIO model with the greedy algorithm's best solution (from five trials) and allowed the model to run for ten hours. Figure 4-6 shows the solution values obtained using the MIO model over time, both with and without warm-starting the MIO model with the greedy algorithm. As shown, utilizing a warm-start allows the MIO model to find better solutions much more quickly. Consequently, we will use a greedy warm-start in the remainder of the models considered in this thesis.

	Greedy	MIO	Value
Missions Completed	1	1	9713
	2	2	9573
	3	3	9525
	4	4	9328
	5	5	9274
	6	6	8911
		7	8719
	8	8	8511
	9		7736
	10	10	7357
		11	6893
	12	12	6111
	13	13	5989
	14	14	5398
	15	15	3672
	16	16	3314
	17	17	2836
	18	18	2558
	19	19	2424
	21	21	1929
	22	22	1294
	24	24	982
	28	28	878
	29	29	744
		30	706
	32		528
	33	33	506
	35	35	163
	36	36	104
		37	98
		38	95
	41	41	83
	43	43	73
	44	44	51
	46		39
Total Value	119604	127812	
Total # of Missions	30	32	

Table 4.4: Mission Completion Comparison

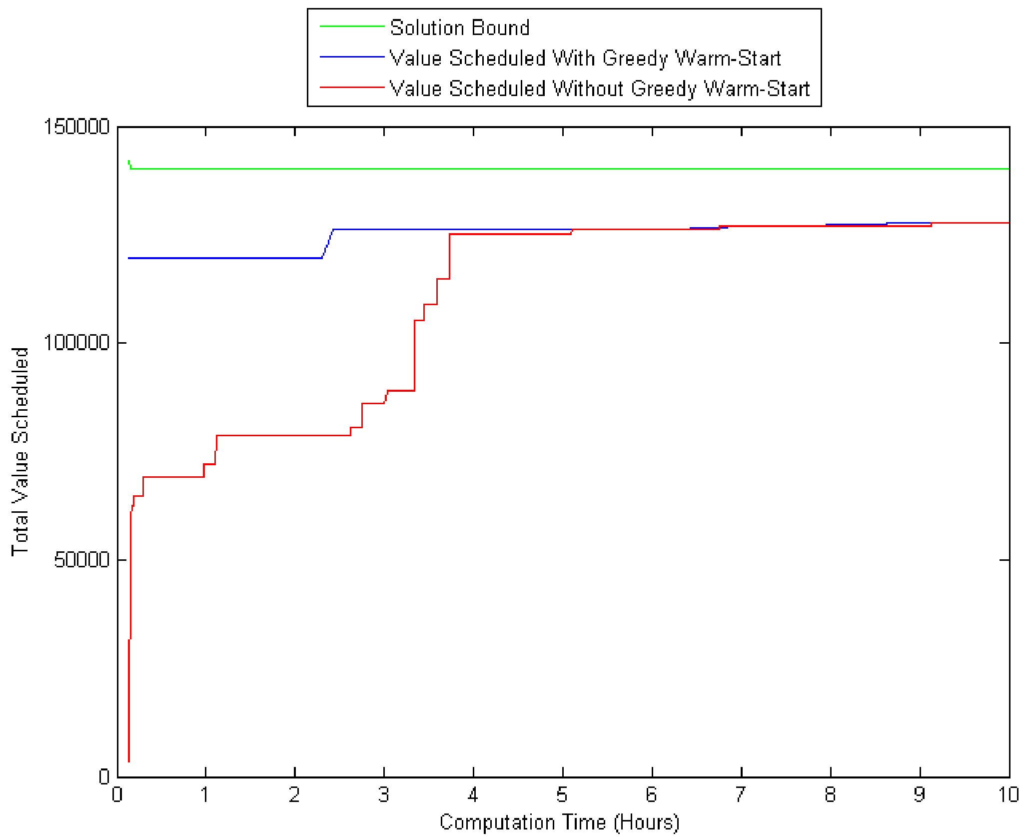

Figure 4-6: Solution Value Obtained Over Time Depending on the Use of A Greedy Warm-Start

However, even with a warm-start, the MIO model still does not find solutions much better than the greedy algorithm in multiple hours of run time. This behavior is not conducive to a daily scheduling tool, as planners will need to make modifications to the model. We recognize that an effective model should use no more than about an hour of run time. Consequently, the next section of the chapter analyzes the effectiveness of the MIO model over time in comparison to the greedy algorithm, depending on the number of potential missions provided to the MIO model.

4.2 Comparing Greedy vs. MIO Solutions

In this section, we examine the process in which MIO solutions are improved upon in comparison to their greedy counterparts. Specifically, we explore how quickly MIO solutions improve and quantify how substantial those improvements are. We also compare the suitability of the MIO model when it is provided with many potential missions or far fewer.

We analyze ten separate scenarios, similar to the example scenario previously presented. Specifically, we have 20 aircraft, 4 tankers, 4 anchor tracks, and 8 bases oriented in the same way as the example scenario presented at the start of this chapter. (Note that the example scenario presented is Trial 4 in this section and the remainder of the thesis.) Each of the scenarios also has 50 potential missions. However, the missions are drawn uniformly over the enemy region and have varying values drawn from a piece-wise linear distribution. Also, each of the missions may require a different strike package, have varying time windows for completion, and require varying amounts of time on station. We warm-started every scenario with the best of five runs of the greedy algorithm (as the solutions change slightly depending on the random selection of some possible completion times). Then, we allowed the MIO model to run for up to ten hours, at which point the run was ended with the current best solution and bounds recorded. Again all tests were programmed in Julia, interfaced with Gurobi, and were run on an Intel Xeon E5687W (3.1 GHz) using up to 8 cores and 64 GB of RAM.

4.2.1 Basic Run Comparison

When providing the MIO model with all variables, we note that its solution time is very slow. This behavior is understandable, as a model of this size has approximately 15,000 variables and 3 million constraints. Even after presolve, the model must generally account for over 8,000 variables and 1 million constraints. As a result, we find that the MIO model generally takes a very long time to improve upon the greedy algorithm's solution. In Figure 4-7, we show the improvement of the MIO solution as a percentage increase from the greedy algorithm's solution.

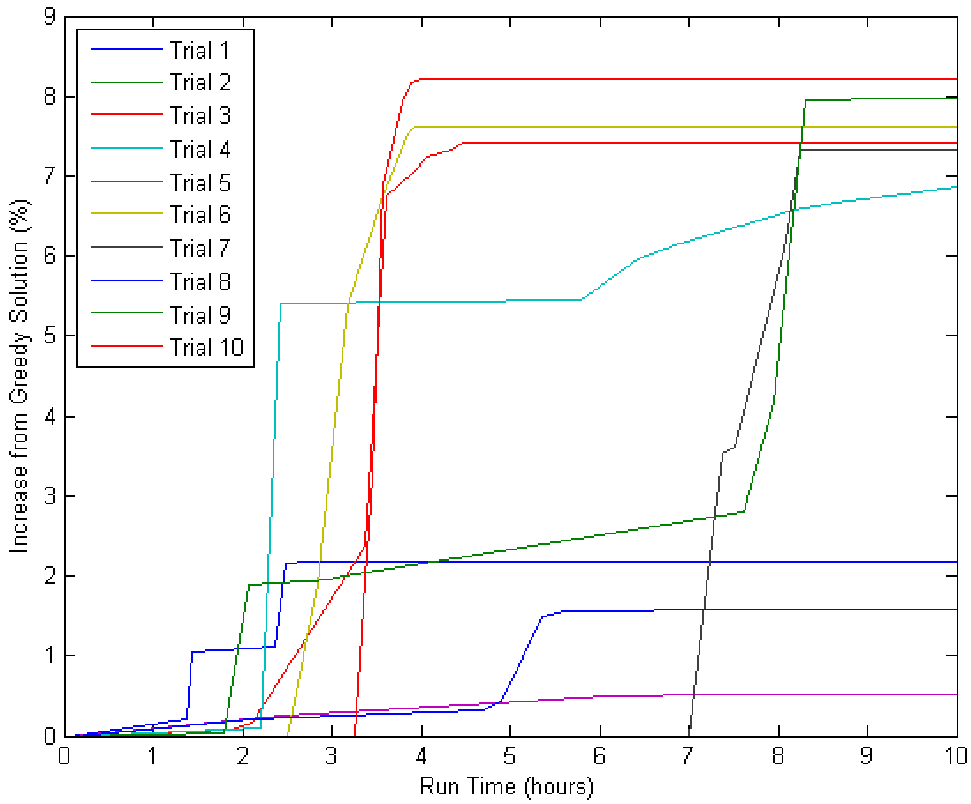

Figure 4-7: Increase in Solution Value of MIO Model Over Greedy Algorithm

We can observe in Figure 4-7 that the MIO model does not improve upon the greedy solution for at least two to three hours. Thereafter, modest improvements are made, capping out at an improvement of approximately 8 percent from the greedy solution. However, some runs yield no improvement or only marginal improvement from the greedy solution, inviting the question of how much the MIO model really helps a scheduler.

In some ways, the non-linearity of our mission values hides the actual benefits proffered by the MIO model. Because of the relative importance of the most important missions listed, completion (or non-completion) of these missions drives the objective function. However, as the values are largely arbitrary, chosen only to force the model to schedule the most important missions first, a more thorough analysis provides better insight into the benefits of the MIO model. In Tables 4.5 and 4.6, we break down the missions scheduled using each model, grouping the missions in tens.

Trials	Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
1	9	8	4	3	5	29
2	9	8	7	1	2	27
3	8	6	5	4	2	25
4	9	8	5	4	4	30
5	8	8	4	4	1	25
6	7	10	5	2	3	27
7	7	5	5	6	1	24
8	7	7	7	3	5	29
9	9	5	4	3	4	25
10	8	5	5	6	4	28
Avg	8.1	7	5.1	3.6	3.1	26.9

Table 4.5: Greedy Algorithm Mission Completion Breakdown

Trials	Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
1	9	8	4	2	6	29
2	9	8	7	1	2	27
3	9	6	4	5	4	28
4	9	9	6	5	3	32
5	8	8	5	5	2	28
6	8	10	7	4	2	31
7	7	7	3	6	1	24
8	7	7	8	4	5	31
9	10	6	4	2	3	25
10	9	5	5	5	4	28
Avg	8.5	7.4	5.3	3.9	3.2	28.3

Table 4.6: MIO Model Mission Completion Breakdown

We notice that the MIO model adds about 1.5 missions on average to the schedule. In addition, the most important missions receive the greatest average increases of about 0.4 more missions scheduled per 10 mission category. In other words, utilization of the MIO model allows planners to formulate a schedule with 1.5 extra missions per day, without compromising anything in the way of scheduling the most important missions. Over the course of a month, this would allow one to schedule over 40 missions more with the same resources. Given the costs associated with operating military aircraft, the savings could be substantial.

However, it is not practical to allow the MIO model to run for 10 hours. The ATO scheduling process relies upon each stage of the planning process taking no more than a few hours, and planners must be able to make adjustments to any schedule proposed by the model, as the model simply cannot account for all specifics involved. In order for the proposed model to be effective, its solution time should not exceed an hour or so. Unfortunately, Figure 4-7 shows that stopping the model (in its current form) before allowing it to run for at least 3-4 hours severely limits its impact.

4.2.2 Varying Number of Missions Provided to the Model

In our attempt to reduce solve time, we investigate the limiting variables and constraints on the solve time of our model. We note the biggest (non-dominated) variables are the x, p, and h variables that grow to the order $\mathcal{O}(ijd)$, $\mathcal{O}(j^2)$, and $\mathcal{O}(tjdy)$, respectively. The biggest constraints are those defining the ordering of the p variables, the travel times of the aircraft, and the travel and loiter times of the tankers [constraints (3.13), (3.15), (3.16), and (3.18)] which grow to the order $\mathcal{O}(ij^2d^2)$, $\mathcal{O}(ij^2d^2)$, $\mathcal{O}(tj^2d^2y^2)$, and $\mathcal{O}(tj^2dy)$, respectively.

While many different indices drive the growth of the model, it is impractical to limit most of them. Limiting the number of aircraft or tankers runs counter to the model's goal of coordinating mission completion on a large scale. Limiting flights or tracks reduces the flexibility of the refueling process, and these indices are relatively small anyway. However, the number of potential missions is present as an index in each of the problematic variables and constraints (and often as a squared factor). Furthermore, the structure of the JIPTL is very conducive to limitation. As the most important missions are already identified and hold much more relative value, limiting the number of potential missions provided to the model can speed up run time without compromising total objective function value.

In this section, we seek to quantify the impact of reducing the number of missions provided to the model for scheduling. Utilizing the same trials as in the last section, we limited the number of potential missions provided to the model to 40, 30, and 20. Then, we ran the model for up to 10 hours and recorded the increase in value from the 50 mission greedy algorithm for each (at a variety of time cutoffs). The average results are shown in Table 4.7 (results for each of the trials can be found in the appendix Tables A.1 through A.20).

	Increase from 50 Mission Greedy Algorithm				
Missions Provided	10 min	30 min	1 hr	3 hr	10 hr
50	0.00	0.00	0.00	1.16	4.97
40	0.48	0.50	1.45	6.08	6.53
30	0.91	4.46	5.49	6.30	6.30
20	1.71	1.71	1.71	1.71	1.71

Table 4.7: Percentage Increase of MIO Model from Greedy Algorithm Over Time

The results of Table 4.7 suggest that limiting the number of potential missions provided to the model can greatly impact model performance. Not only do the models with fewer potential missions find better solutions faster, but the best solutions obtained by the limited models often exceed in value the best solutions found by the basic model after the 10 hour limit. By limiting the model scope, the solver can better search the solution space for schedules that best combine the aircraft toward the completion of the most important missions.

However, we would also hypothesize that limiting the number of potential missions available to the model would impact the total number of missions scheduled. To investigate this trade-off, we again split the potential missions into tens and investigate the breakdown of missions scheduled. The results are summarized in Table 4.8.

	Average Number of Missions Scheduled					
Missions Provided	1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50	8.5	7.4	5.3	3.9	3.2	28.3
40	8.5	8.0	6.1	5.0	0.0	27.6
30	8.6	8.0	6.4	0.0	0.0	23.0
20	8.6	8.0	0.0	0.0	0.0	16.6

Table 4.8: MIO Model Mission Completion Breakdown With Varying Missions Provided

As shown in Table 4.8, decreasing the number of missions provided to the model allows the solver to schedule the most important missions more frequently. While the improvements may seem modest, even intra-category changes can yield significant benefits when they involve the most important missions. On the other hand, as hypothesized, a decrease in the number of missions provided also decreases the total number of missions scheduled. When 40 missions are provided, the decrease in missions scheduled is quite modest, suggesting a level of saturation within the schedule. However, when 20 or 30 missions are provided, far fewer missions are scheduled, suggesting that more missions could be scheduled provided the right type of aircraft were available.

We recall from Table 4.7 that the best solutions were found when the MIO model was provided with 30 or 40 missions. As a result, we can postulate that finding the best solutions quickly involves balancing the ability to search the solution space quickly with having enough choice to saturate the schedule over the entire period. This conclusion suggests that a model that cleverly combines the flexibility of the MIO model for the most important missions with the speed of the greedy algorithm for saturating the schedule thereafter will yield the best results. We investigate such a hybrid in the next section.

4.3 Hybrid Modeling

As shown in previous sections, utilizing exclusively the MIO model to create an ATO schedule is not practical from a computational time perspective (even when warm-started with the greedy algorithm). Furthermore, limiting the number of potential missions provided to the MIO model reduces the number of total missions that can be completed. However, it is possible to create hybrid models which employ the MIO model to look for good solutions on a subset of the potential missions and then use other algorithms to populate vacant times in the schedule with missions that fit into those vacancies. Specifically, we consider a model that utilizes the greedy algorithm to populate these openings.

4.3.1 Greedy Completed MIO Model

The structure of the Greedy Completed MIO Model (GCMIO) can be summarized as follows:

1. Split apart the potential missions into subsets of the $\mathcal{J}_1$ most important missions and the $\mathcal{J}_2 = \mathcal{J} - \mathcal{J}_1$ remaining missions.

2. Apply the greedy algorithm to $\mathcal{J}_1$.

3. Using the warm-start from the previous step, run the MIO model on $\mathcal{J}_1$.

4. Lock in the solutions of the MIO model, noting aircraft/tanker busy times, fuel consumption, etc.

5. Run the greedy algorithm on $\mathcal{J}_2$, only allowing missions to be completed if they do not conflict with previously scheduled missions from earlier steps.

6. The final schedule is the combination of all missions completed from both $\mathcal{J}_1$ and $\mathcal{J}_2$.

In short, we run the MIO model on the most important missions, and use the greedy algorithm to fill in any empty spaces with the most important missions that fit into the vacancies. As the greedy algorithm solves in seconds, our computation time is simply reduced to that of the MIO model on the subset of potential missions $\mathcal{J}_1$. Consequently, we can get solutions that approximate the MIO-only solutions in a shorter amount of time. Average results are shown in Table 4.9 below.

The results of Table 4.9 are quite encouraging. Utilization of the GCMIO model provides us with solutions comparable to the MIO-only model, but in a fraction of the time (1 hour as opposed to 10 hours). We do note that the improvements are not universal. For example, we notice that the MIO-only model had a better average performance than the GCMIO model when solution time was less than or equal to 30 minutes. Looking at the individual trials (found in Tables A.1 through A.20 in Appendix A) we notice that this behavior is likely attributable to randomness in the selection of the mission completion time, particularly when the GCMIO model only

Missions Provided	Increase from 50 Mission Greedy Algorithm				
	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	0.00	0.00	0.00	1.16	4.97
40- MIO-only	0.48	0.50	1.45	6.08	6.53
30- MIO-only	0.91	4.46	5.49	6.30	6.30
20- MIO-only	1.71	1.71	1.71	1.71	1.71
40- GCMIO	0.73	0.99	2.10	n/a	n/a
30- GCMIO	0.59	4.10	6.39	n/a	n/a
20- GCMIO	5.61	5.61	5.61	n/a	n/a

Table 4.9: Percentage Increase of MIO and GCMIO Models from Greedy Algorithm

makes very small improvements from the greedy warm start. However, when the MIO portion of the model improves upon the greedy warm start substantially, the GCMIO clearly outperforms the MIO-only model. This improvement is particularly noticeable with the 20-missions provided GCMIO model, where just 10 minutes of run time beats the greedy algorithm by an average of over 5.6 percent, giving up less than 1 percent of an improvement from the best model.

To further examine the impact of the GCMIO on our solutions, we plot the solutions obtained over time for the example scenario introduced at the start of the chapter. This plot can be found in Figure 4-8. In this plot, upward and downward facing triangles indicate the amount of value gained from the greedy completion portion of the optimization. The 20-missions provided GCMIO model provides the greatest insight for this scenario. We see that the MIO portion of the model solves to optimality in just a few minutes, and the greedy completion portion of the model adds nearly 5000 in value, bringing the final solution very close to the best solution obtained (see Table A.7 in Appendix A for details). Obtaining such a good solution in just a few minutes suggests that we can find good solutions to bigger problems within our self-imposed one hour time limit. We will investigate this idea in the next section.

In addition to computation time, the other area we hoped to investigate with the GCMIO model was that of total missions scheduled. We explore this matter of concern in Table 4.10.

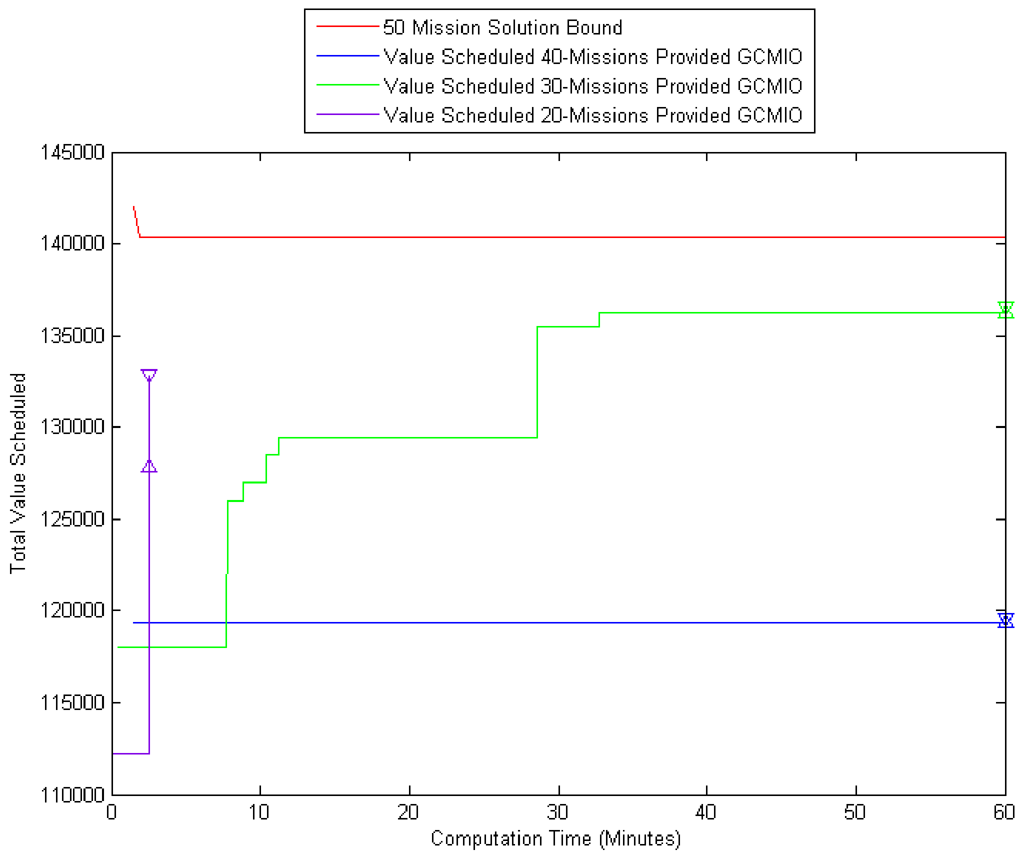

Figure 4-8: Trial 4 GCMIO Solution Value Over Time

Table 4.10 shows the true value of the GCMIO model. Given only 1 hour of computation time as opposed to 10 hours, the GCMIO model schedules missions to a level of saturation only a couple of missions short of the level found by the standard 50-missions provided MIO model. As a trade-off, the GCMIO model sacrifices only modestly in the number of the most important missions scheduled. Given the quick turnarounds required at every step of the Air Tasking Cycle, sacrificing some "value" for computational speed is most certainly a good trade-off.

	Average Number of Missions Scheduled					
Missions Provided	1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	8.5	7.4	5.3	3.9	3.2	28.3
40- MIO-only	8.5	8.0	6.1	5.0	0.0	27.6
30- MIO-only	8.6	8.0	6.4	0.0	0.0	23
20- MIO-only	8.6	8.0	0.0	0.0	0.0	16.6
40- GCMIO	8.3	7.1	5.4	4.3	2.6	27.7
30- GCMIO	8.6	7.8	6.3	1.9	1.3	25.9
20- GCMIO	8.6	7.9	4.0	3.0	2.8	26.3

Table 4.10: 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

4.4 Utilizing the GCMIO on a Larger Scenario

Ultimately, the goal of this thesis is to demonstrate the viability of our model on theater-sized problems. Most JAOCs will maintain hundreds of aircraft, but not all aircraft will be available to fly on any given day. Certain aircraft will have maintenance issues, and the JFC may want to hold back certain aircraft in case unforeseen circumstances arise. Consequently, we estimate that about 200 aircraft will be flown on a busy day (although this estimate can vary greatly from JAOC to JAOC). While our model cannot handle problems of this size, we wish to demonstrate its ability to handle problems of about 80 aircraft at a time. Then, planners can break down larger problems into chunks of 80 or so aircraft by region. The combined schedule will generally approximate the optimal solution which can be obtained if the problem is solved globally.

With that goal in mind, we define a problem with 80 aircraft, 10 tankers, and 100 potential missions. The map of the scenario is shown in Figure 4-9. To reduce computational complexity, we take advantage of the fact that many aircraft types (fighters, in particular) will not be assigned to fly missions as solo aircraft. They will always fly missions at least in pairs. Consequently, we can model two aircraft as one "aircraft variable," thereby reducing our total number of aircraft variables to 50. Aircraft and tanker specifications for this scenario are shown in Table 4.11 and 4.12, respectively.

We make the same assumptions for this larger scenario as we made for our initial

Aircraft Variable	Number of Aircraft	Type(s)	Base	Speed (knots)	Fuel Capacity (1000lbs)	Burn Rate (1000lbs/hr)	Turn time (hrs)
1	2	1, 7	10	420	5.3	2.1	0.5
2	2	1, 7	10	420	5.3	2.1	0.5
3	2	1, 7	2	420	5.3	2.1	1.0
4	2	1, 7	2	420	5.3	2.1	1.0
5	2	1, 7	4	420	5.3	2.1	1.0
6	2	1, 7	4	420	5.3	2.1	1.0
7	2	1, 7	6	420	5.3	2.1	1.0
8	2	1, 7	6	420	5.3	2.1	1.0
9	2	1, 7	6	420	5.3	2.1	1.0
10	2	1, 7	6	420	5.3	2.1	1.0
11	2	1, 7	7	420	6.9	3.2	1.0
12	2	1, 7	7	420	6.9	3.2	1.0
13	2	1, 7	7	420	6.9	3.2	1.0
14	2	1, 7	7	420	6.9	3.2	1.0
15	2	1, 7	8	420	10.5	4.0	1.5
16	2	1, 7	8	420	10.5	4.0	1.5
17	2	1, 7	8	420	10.5	4.0	1.5
18	2	1, 7	8	420	10.5	4.0	1.5
19	2	1, 7	8	420	10.5	4.0	1.5
20	2	1, 7	8	420	10.5	4.0	1.5
21	2	1, 6	3	420	20.8	6.6	1.5
22	2	1, 6	3	420	20.8	6.6	1.5
23	2	1, 6	10	420	20.8	6.6	1.5
24	2	1, 6	10	420	20.8	6.6	1.5
25	1	2	1	340	9.0	2.4	1.5
26	1	2	1	340	9.0	2.4	1.5
27	1	2	4	340	9.0	2.4	1.5
28	1	2	4	340	9.0	2.4	1.5
29	1	2	4	340	9.0	2.4	1.5
30	1	2	9	340	9.0	2.4	1.5
31	1	2	9	340	9.0	2.4	1.5
32	1	2	7	420	14.5	3.3	0.5
33	1	2	7	420	14.5	3.3	0.5
34	1	2	7	420	14.5	3.3	0.5
35	1	2	7	420	14.5	3.3	0.5
36	1	2	7	420	14.5	3.3	0.5
37	2	3	1	420	12.5	7.0	1.5
38	2	3	1	420	12.5	7.0	1.5
39	2	3	6	420	12.5	5.4	2.0
40	2	3	6	420	12.5	5.4	2.0
41	1	4, 10	3	450	200.0	16.6	3.0
42	1	4	5	390	260.0	12.0	2.5
43	1	4	5	390	260.0	12.0	2.5
44	1	4	11	450	185.0	23.1	3.0
45	2	5, 8	4	420	5.3	2.1	1.0
46	2	5, 8	4	420	5.3	2.1	1.0
47	1	5, 9	1	280	53.0	5.2	1.5
48	1	5, 9	1	280	53.0	5.2	1.5
49	1	5, 9	8	420	9.8	4.4	1.5
50	1	5, 9	8	420	9.8	4.4	1.5

Table 4.11: Notional Aircraft Specifications

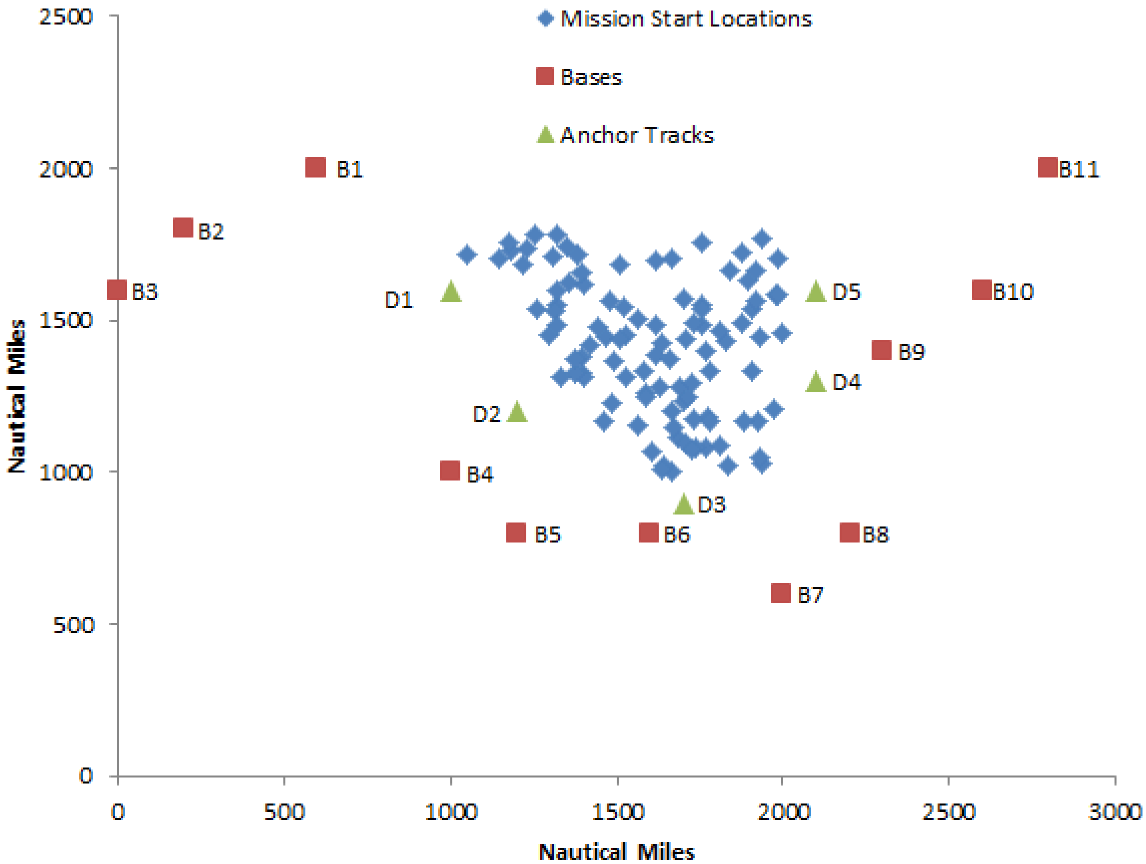

Figure 4-9: Scenario Locations

scenario with two notable exceptions. First, we do not assume that travel from bases to anchor tracks is necessarily point-to-point. Rather, we ensure travel does not cross over enemy territory. This addition simply adds realism to the problem. Second, we assign multiple "types" to certain aircraft. In other words, we assume certain missions can only be performed by a specific air frame and create aircraft types for those aircraft to dictate this special need (aircraft types are found in Table 4.11). Mission characteristics are drawn from the same distributions as for the initial scenario, although package requirements are somewhat modified. Characteristics for all of the missions can be found in Tables A.21 through A.23 in Appendix A.

Given the increased number of constraints and variables required, we opt for a 50-mission provided GCMIO model which greedily completes missions 51 through 100. Using the same Intel Xeon E5687W utilizing up to 8 cores and 64 GB of RAM,

Tanker	Base	Speed (knots)	Fuel Capacity (1000lbs)	Burn Rate (1000lbs/hr)	Turn time (hrs)
1	5	400	160.0	8.1	2.0
2	5	400	160.0	8.1	2.0
3	5	400	160.0	8.1	2.0
4	11	400	160.0	8.1	2.0
5	11	400	160.0	8.1	2.0
6	2	480	331.0	22.3	2.0
7	2	480	331.0	22.3	2.0
8	2	480	331.0	22.3	2.0
9	5	480	331.0	22.3	2.0
10	5	480	331.0	22.3	2.0

Table 4.12: Notional Tanker Specifications

Missions Completed						
Both Models					Greedy Only	GCMIO
1	10	25	46	72	55	34
2	11	26	52	79	60	43
3	14	27	53	81	65	44
5	15	30	54	83	71	48
6	17	38	56	84	78	49
7	20	39	57	91		50
8	21	41	63			82
9	24	42	69			94

Table 4.13: Big Scenario Mission Completion Comparison

we allow the model to run for 1 hour. The obtained schedule is shown in Figures 4-10 and 4-11.

We observe that the final solution takes on a value of 166187, a 4.6 percent increase from the basic greedy solution of 158831. A more thorough analysis shows that the GCMIO model schedules a total of 46 missions, as opposed to 43 missions scheduled by the greedy. The detailed breakdown of which missions are completed can be found in Table 4.13. This comparison shows that the GCMIO model schedules three additional missions, while only upgrading to more valuable missions in the process. Over the course of a month, the GCMIO model would allow planners to schedule nearly 100 missions more than current methods. In a high operations environment, the benefits could be significant.

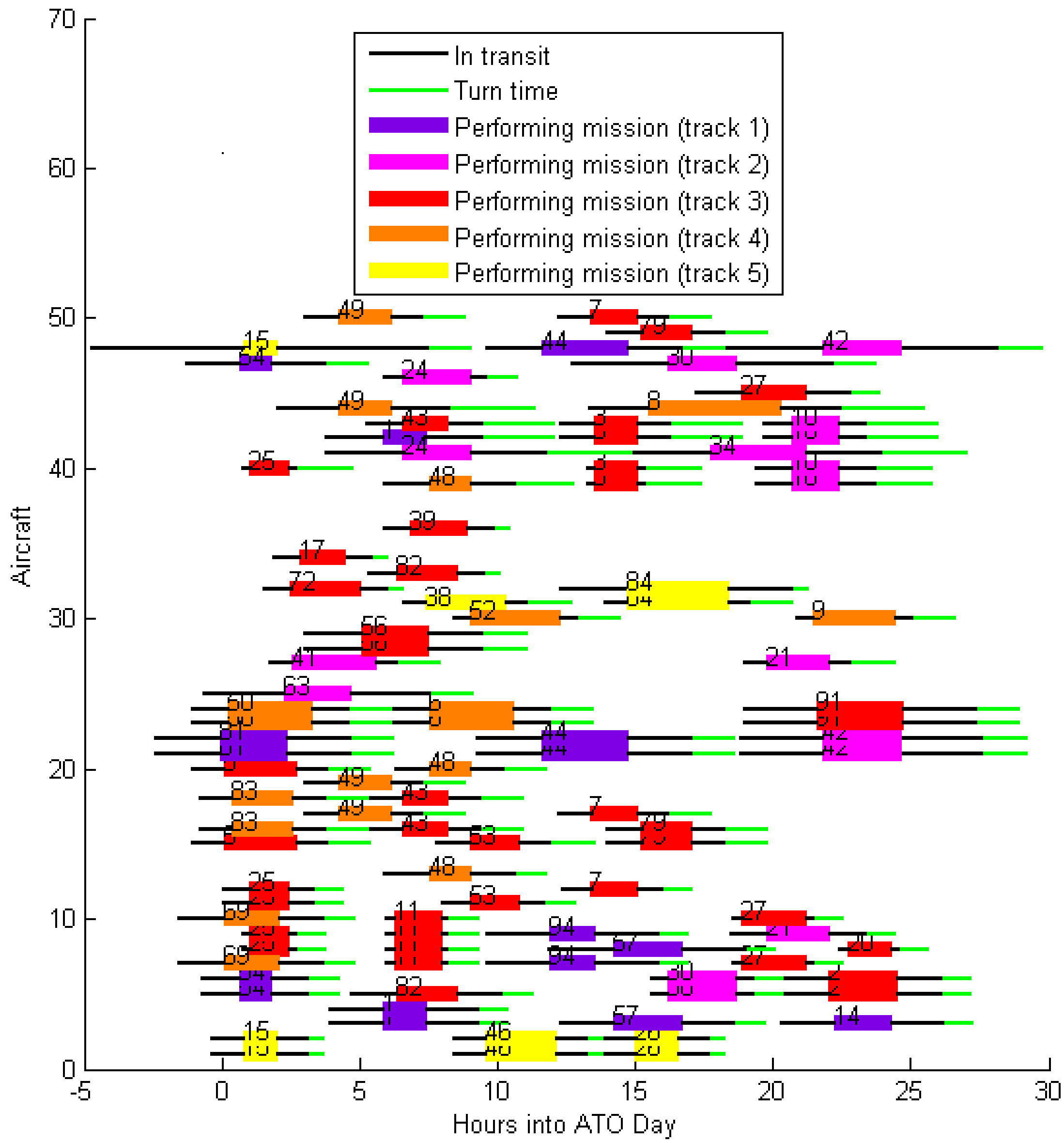

Figure 4-10: GCMIO Model's Aircraft Schedule

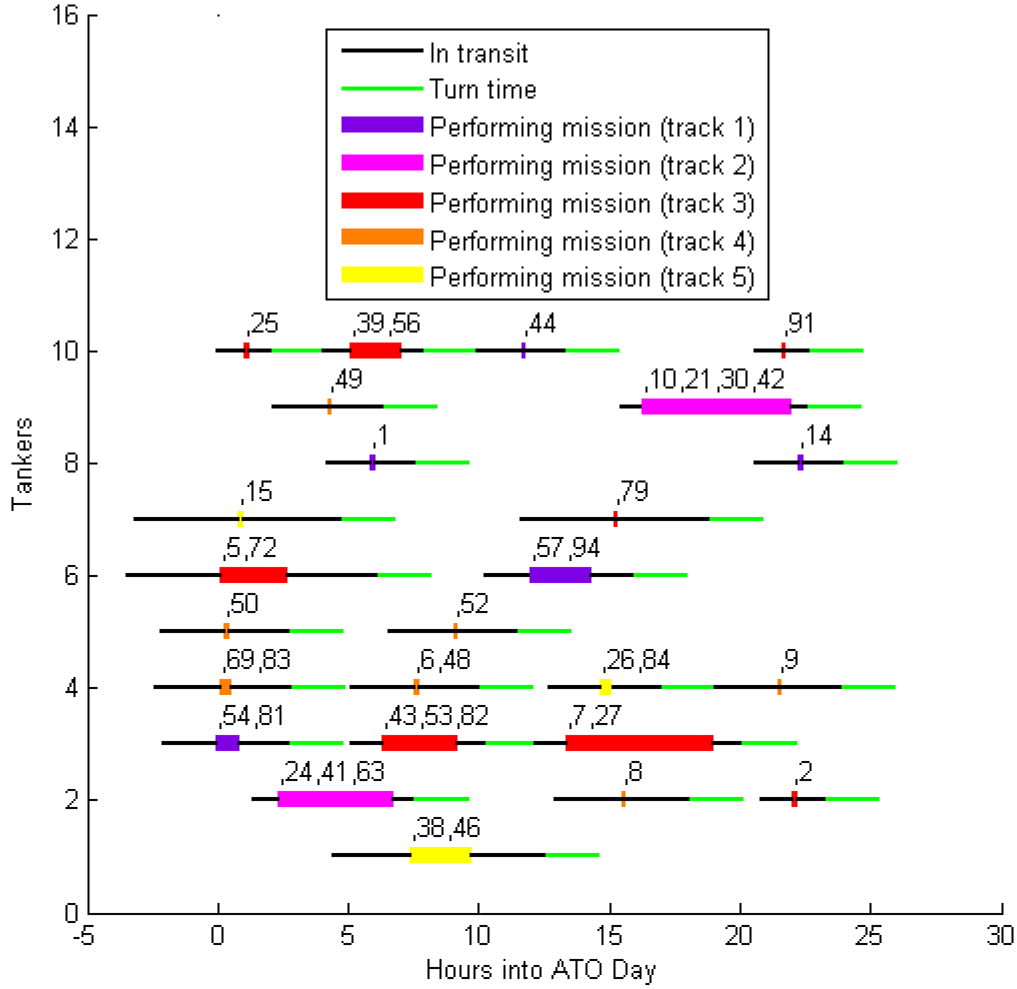

Figure 4-11: GCMIO Model's Tanker Schedule

THIS PAGE INTENTIONALLY LEFT BLANK

Chapter 5

Conclusions and Future Work

Results of the modeling framework proposed in this thesis suggest that automated mission scheduling will be a realistic tool for increasing planning proficiency in the near future. The hybrid model presented in this thesis shows great potential, as it is capable of finding nearly optimal solutions to moderate-sized problems in less than an hour. Given even modest improvements in solver and computer processing power, this model could solve theater-sized scheduling problems in the same time in just a few years. Even without these processing improvements, cleverly breaking down the AOR of a JAOC into smaller regions would allow a planner to utilize this model as it currently stands with very modest compromises to solution values. In addition, the flexible framework, which accounts for both aircraft "package" requirements and air refueling considerations, can be modified and refined to manage a variety of campaign objectives or fleet compositions. Accordingly, we believe the modeling framework in this thesis provides a proof of concept for automated MIO-based models in the area of coordinated mission scheduling.

5.1 Summary of Results and Contributions

- We summarized the Air Tasking Cycle within a JAOC. We identified an area for improvement in the scheduling process; specifically, we wished to combine the mission selection and aircraft/munitions allocation stages of the Air Tasking

Cycle for optimal scheduling efficiency.

- We presented a modeling framework that incorporates both unified completion of missions by aircraft packages and air refueling considerations. We utilized this framework to create aircraft-by-aircraft schedules over a specified time horizon.

- We developed greedy heuristic, mixed integer optimization, and hybrid models within this framework. We also offered optional constraints for the MIO model that give extra flexibility to planners in the scheduling process.

- We compared the performance of the models, focusing on the total value of missions completed and the total number of missions completed, and contrasting these performance measures with computational run time.

- We demonstrated the capability of the GCMIO model to solve problems of 20 aircraft, 50 missions, and 4 tankers to within 8.1 percent of optimality in just one hour of computation time, an increase of 6.39 percent from current greedy methods. We also demonstrated its capability to solve a problem of 80 aircraft, 100 potential missions, and 10 tankers in one hour, yielding an increase of 4.6 percent from the greedy algorithm's solution.

5.2 Future Work

Mission planning within JAOCs is a broad area where many improvements can be made using operations research techniques. We discuss a few areas for future research in this section.

5.2.1 Future Work Within This Modeling Framework

Within the modeling framework proposed in this thesis, there are a number of areas where improvements could be made, both in computation time and realism of the structure. For one, various hybrid models could be considered that may outperform the hybrid model covered in this thesis. Second, adding robustness to many of the model parameters would make the model robust to uncertainty.

As discussed in earlier sections, computation time for this modeling framework is currently acceptable for moderate-sized problems, but somewhat excessive for theater-sized problems. One way of curbing this computation time is by continuing to examine hybrid models that combine heuristics with standard mixed integer optimization techniques, or use more complicated heuristics exclusively. For example, given the relatively strong performance of the greedy algorithm, one could experiment with utilizing a modified greedy algorithm as a heuristic at certain steps of the branch and bound process of the MIO (see [3] for an explanation of the branch and bound process). One could also run algorithms that explore different combinations of models, to guess which hybrid model will be most effective. In the category of exclusively heuristic algorithms, one could explore simulated annealing or genetic algorithms with the mission scheduling problem as other options for finding good solutions fast.

A key complexity that must be dealt with by mission planners is the uncertainty in certain model parameters. For example, weather could cause travel times for aircraft to increase/decrease or fuel consumption rates to increase/decrease. If a planner creates a schedule that is too full so that some of these normal variations cause the schedule to fall apart, then the schedule is not particularly useful. Understandably, creating a schedule that protects against such "normal" variation is crucial. Robust optimization is designed to protect against this exact situation. Consequently, adding robustness to uncertain parameters could yield significant benefits when applying the model in practice. Uncertain parameters that could be controlled include value, travel times, fuel consumption rates, and mission completion times. Creating a robust optimization model that accounts for uncertainty in these parameters would significantly aid mission planners in creating schedules that hold firm, even through the "fog of war."

5.2.2 Future Work for Mission Planning in JAOCs

The mission planning area for JAOCs is ripe for development. There are many areas within this context where improvements can be made. We will mention just a few of these areas.

One area where improvement in mission planning could yield significant benefits concerns itself with Time Sensitive Targets (TSTs). TSTs are high value targets "requiring immediate response because they pose (or will soon pose) danger to friendly forces or are highly lucrative, fleeting targets of opportunity" [27]. Given the nature of TSTs, they cannot be dealt with through the standard ATO process. Rather, commanders must decide whether fleeting targets should be attacked, and if so, what forces should be diverted or otherwise tasked to deal with the targets. Methods have been proposed to modify an ATO to maximize benefits once TSTs do appear (see [27]). However, no models have been developed to create the initial ATO based on the expectation that some TSTs will appear. Commanders have the ability to allocate aircraft to "standby" missions, either in the air or at their home base, or can choose to divert aircraft from assigned missions to attack TSTs. Creating a model that aids mission planners in determining which aircraft should be tasked to known missions and which should be placed on standby would allow for the best possible allocation of aircraft on the battlefield.

Mission "creation" is another area for improvement within mission planning. In this thesis, we assumed missions were already developed, but could include multiple targets or require completing a sequence of actions. Consequently, we could assume a finite, preselected mission completion time, fuel consumption rate, and number and type of munitions required. However, the process of developing these missions is not trivial. Targets can only be combined if an aircraft has enough fuel to travel between locations, enough munitions to adequately attack (or otherwise engage) all targets, and the capability to engage all targets, among other things. Jointly "creating" missions to accomplish campaign objectives and scheduling them provides the framework for significant improvement from "by-hand" scheduling.

Another area for improvement in the mission planning area centers on the air refueling process. Finding ways to model how many tankers should refuel a given package of aircraft and coordinating these refuelings to account for individual fuel upload rates of aircraft could substantially improve the process of assigning tankers for air refueling.

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix A

Tables

Tables A.1 through A.20 contain the individual results of separate trials run on varying example scenarios (with 20 aircraft, 50 potential missions, and 4 tankers). These tables show the total "value" obtained by missions scheduled over time, as well as the total number of missions scheduled, broken down into groups of ten. Note that for the tables displaying the number of missions scheduled, the MIO-only model results are based on a computation time of 10 hours whereas the GCMIO model results are based on a computation time of 1 hour.

	Value					
Missions Provided	Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	73304	73304	73304	73304	73304	74903
40- MIO-only	73286	73286	73286	75386	76824	77148
30- MIO-only	72867	72867	76379	76782	76782	76782
20- MIO-only	70841	74353	74353	74353	74353	74353
40- GCMIO	73304	73304	73304	76510	n/a	n/a
30- GCMIO	73304	73304	77040	77040	n/a	n/a
20- GCMIO	73304	75821	75821	75821	n/a	n/a

Table A.1: Trial 1 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	9	8	4	2	6	29
40- MIO-only	9	10	5	3	0	27
30- MIO-only	9	10	5	0	0	24
20- MIO-only	9	10	0	0	0	19
40- GCMIO	9	9	6	5	2	31
30- GCMIO	9	10	5	2	0	26
20- GCMIO	9	10	2	3	5	29

Table A.2: Trial 1 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	76440	76440	76440	76440	76440	76440
40- MIO-only	76767	76767	76767	77074	80511	80511
30- MIO-only	76711	76711	80148	80148	80148	80148
20- MIO-only	73964	77933	77933	77933	77933	77933
40- GCMIO	76440	76799	76919	79517	n/a	n/a
30- GCMIO	76440	76832	80252	80252	n/a	n/a
20- GCMIO	76440	79215	79215	79215	n/a	n/a

Table A.3: Trial 2 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	9	8	7	1	2	27
40- MIO-only	9	10	7	5	0	31
30- MIO-only	9	10	7	0	0	26
20- MIO-only	9	10	0	0	0	19
40- GCMIO	9	9	7	4	2	31
30- GCMIO	9	10	7	1	2	29
20- GCMIO	9	10	4	2	2	27

Table A.4: Trial 2 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	64088	64088	64088	64088	64188	68837
40- MIO-only	67897	67897	67897	67897	68638	68960
30- MIO-only	63908	64011	68348	68348	68455	68455
20- MIO-only	66273	67205	67205	67205	67205	67205
40- GCMIO	64088	67990	67990	68104	n/a	n/a
30- GCMIO	64088	67963	68560	68560	n/a	n/a
20- GCMIO	64088	68492	68492	68492	n/a	n/a

Table A.5: Trial 3 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	9	6	4	5	4	28
40- MIO-only	9	7	5	6	0	27
30- MIO-only	9	7	5	0	0	21
20- MIO-only	9	7	0	0	0	16
40- GCMIO	9	6	4	5	3	27
30- GCMIO	9	6	4	2	1	22
20- GCMIO	9	6	3	2	4	24

Table A.6: Trial 3 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	119604	119604	119604	119604	126077	127812
40- MIO-only	119358	119358	119358	119358	128736	128736
30- MIO-only	118057	126956	135510	136216	136216	136216
20- MIO-only	112230	127842	127842	127842	127842	127842
40- GCMIO	119604	119604	119604	119604	n/a	n/a
30- GCMIO	119604	119604	130152	136527	n/a	n/a
20- GCMIO	119604	132884	132884	132884	n/a	n/a

Table A.7: Trial 4 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	9	9	6	5	3	32
40- MIO-only	9	9	9	4	0	31
30- MIO-only	10	9	8	0	0	27
20- MIO-only	10	9	0	0	0	19
40- GCMIO	9	8	5	4	4	30
30- GCMIO	10	9	8	1	3	31
20- GCMIO	10	9	3	4	3	29

Table A.8: Trial 4 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	112092	112092	112092	112092	112377	112666
40- MIO-only	111996	111996	111996	111996	115674	115785
30- MIO-only	111731	112465	115788	115788	115788	115788
20- MIO-only	107081	110686	110686	110686	110686	110686
40- GCMIO	112092	112092	112092	112092	n/a	n/a
30- GCMIO	112092	112092	112762	115999	n/a	n/a
20- GCMIO	112092	115551	115551	115551	n/a	n/a

Table A.9: Trial 5 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	8	8	5	5	2	28
40- MIO-only	8	9	4	5	0	26
30- MIO-only	8	9	6	0	0	23
20- MIO-only	8	9	0	0	0	17
40- GCMIO	8	8	4	4	1	25
30- GCMIO	8	9	6	2	1	26
20- GCMIO	8	9	4	2	1	24

Table A.10: Trial 5 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	76533	76533	76533	76533	79286	82357
40- MIO-only	76060	76060	76199	80620	82315	83014
30- MIO-only	75021	79456	81908	81908	81908	81908
20- MIO-only	72472	76893	76893	76893	76893	76893
40- GCMIO	76533	76533	78440	80095	n/a	n/a
30- GCMIO	76533	76533	82146	82146	n/a	n/a
20- GCMIO	76533	80974	80974	80974	n/a	n/a

Table A.11: Trial 6 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	8	10	7	4	2	31
40- MIO-only	8	10	8	5	0	31
30- MIO-only	8	10	8	0	0	26
20- MIO-only	8	10	0	0	0	18
40- GCMIO	8	9	5	3	3	28
30- GCMIO	8	10	8	2	2	30
20- GCMIO	8	10	6	1	2	27

Table A.12: Trial 6 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	73131	73131	73131	73131	73131	78500
40- MIO-only	73002	73002	73002	73002	78990	79026
30- MIO-only	72545	72545	72545	78444	78444	78444
20- MIO-only	71426	76632	76632	76632	76632	76632
40- GCMIO	73131	73131	73131	73131	n/a	n/a
30- GCMIO	73131	73131	73131	78726	n/a	n/a
20- GCMIO	73131	77744	77744	77744	n/a	n/a

Table A.13: Trial 7 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled					
	1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	7	7	3	6	1	24
40- MIO-only	7	7	5	8	0	27
30- MIO-only	7	7	6	0	0	20
20- MIO-only	7	7	0	0	0	14
40- GCMIO	7	5	5	6	1	24
30- GCMIO	7	7	6	3	0	23
20- GCMIO	7	7	3	4	1	22

Table A.14: Trial 7 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value					
	Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	69127	69127	69127	69127	69274	70226
40- MIO-only	68925	68925	68925	69232	69232	70540
30- MIO-only	68174	68174	68174	68956	68956	68956
20- MIO-only	63136	63136	63136	63136	63136	63136
40- GCMIO	69127	69127	69127	69782	n/a	n/a
30- GCMIO	69127	69127	69127	69311	n/a	n/a
20- GCMIO	69127	69127	69127	69127	n/a	n/a

Table A.15: Trial 8 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled					
	1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	7	7	8	4	5	31
40- MIO-only	7	7	8	5	0	27
30- MIO-only	7	7	8	0	0	22
20- MIO-only	7	7	0	0	0	14
40- GCMIO	7	7	8	3	3	28
30- GCMIO	7	7	8	1	0	23
20- GCMIO	7	7	7	3	5	29

Table A.16: Trial 8 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value					
	Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	84297	84297	84297	84297	85950	91018
40- MIO-only	84440	84440	84440	84440	91153	91153
30- MIO-only	84068	84068	84068	84068	90781	90781
20- MIO-only	80236	87634	87634	87634	87634	87634
40- GCMIO	84297	84922	84922	84922	n/a	n/a
30- GCMIO	84297	84297	84297	88847	n/a	n/a
20- GCMIO	84297	90247	90247	90247	n/a	n/a

Table A.17: Trial 9 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	10	6	4	2	3	25
40- MIO-only	10	6	4	2	0	22
30- MIO-only	10	6	4	0	0	20
20- MIO-only	10	6	0	0	0	16
40- GCMIO	9	5	5	3	3	25
30- GCMIO	10	5	4	2	2	23
20- GCMIO	10	6	3	2	2	23

Table A.18: Trial 9 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Missions Provided	Value Greedy	10 min	30 min	1 hr	3 hr	10 hr
50- MIO-only	73056	73056	73056	73056	73056	79053
40- MIO-only	72838	72838	72838	72838	79551	79968
30- MIO-only	71509	74208	78696	78696	78696	78696
20- MIO-only	68740	74979	74979	74979	74979	74979
40- GCMIO	73056	73056	73056	73056	n/a	n/a
30- GCMIO	73056	72586	78321	79334	n/a	n/a
20- GCMIO	73056	79487	79487	79487	n/a	n/a

Table A.19: Trial 10 Value Obtained Over Time

Missions Provided	Number of Missions Scheduled 1st Ten	2nd Ten	3rd Ten	4th Ten	5th Ten	Total
50- MIO-only	9	5	5	5	4	28
40- MIO-only	9	5	6	7	0	27
30- MIO-only	9	5	7	0	0	21
20- MIO-only	9	5	0	0	0	14
40- GCMIO	8	5	5	6	4	28
30- GCMIO	9	5	7	3	2	26
20- GCMIO	9	5	5	7	3	29

Table A.20: Trial 10 10 hr MIO vs. 1 hr GCMIO Mission Completion Comparison

Tables A.21 through A.23 show the mission characteristics for the 100 potential missions to be flown in the larger scenario in Section 4.4. Note that certain packages require a specific air frame (aircraft types 6 through 10) while others can be completed using aircraft falling into broader categories (aircraft types 1 through 5). The GCMIO model balances aircraft assignments according to these trade-offs.

Mission	Value	Mission Time Window		Amount of Time on Station Required	Number of Aircraft Required									
		Earliest Start Time	Latest Start Time		Strike-F	Attack-A	Air-to-Air	Bomber-B	SEAD	Strike Fighter Type 1	Strike Fighter Type 2	Bomber Type 1	SEAD Fighter	Other SEAD
1	9570	1.00	23.00	0.397	2	0	0	1	0	0	0	0	0	0
2	9353	1.00	23.00	0.492	2	0	0	0	0	0	0	0	0	0
3	8689	10.97	14.05	0.499	0	0	2	2	0	0	0	0	0	0
4	8520	7.91	12.29	0.383	0	0	2	2	0	0	0	0	0	0
5	8514	1.00	23.00	0.751	2	0	0	0	0	0	0	0	0	0
6	8260	6.51	12.24	0.993	2	0	0	0	0	0	0	0	0	0
7	8245	8.11	13.75	0.907	2	0	0	0	0	0	0	0	1	0
8	8237	16.01	19.74	0.334	0	0	0	1	0	0	0	0	0	0
9	7745	21.39	22.32	1.246	0	1	0	0	0	0	0	0	0	0
10	7673	16.76	21.36	0.368	0	0	2	2	0	0	0	0	0	0
11	7530	5.33	10.66	0.266	4	0	0	0	0	0	0	0	0	0
12	7255	4.60	9.35	0.276	4	0	1	1	1	0	0	0	0	0
13	7144	21.27	21.51	1.393	0	1	1	0	0	0	0	0	0	0
14	6953	1.00	23.00	0.551	1	0	0	0	0	0	0	0	0	0
15	6844	1.06	6.07	0.325	2	0	0	0	0	0	0	0	1	0
16	6669	15.73	20.30	0.792	4	0	1	1	1	0	0	0	0	0
17	6604	2.31	3.23	0.870	0	1	0	0	0	0	0	0	0	0
18	6221	4.74	5.40	0.501	0	1	1	0	0	0	0	0	0	0
19	6028	19.54	23.50	0.500	0	0	2	2	0	0	0	0	0	0
20	5719	1.00	23.00	0.997	1	0	0	0	0	0	0	0	0	0
21	5684	20.42	20.43	0.964	1	1	0	0	0	0	0	0	0	0
22	5423	1.58	2.31	1.118	0	1	1	0	0	0	0	0	0	0
23	5344	1.00	23.00	0.506	4	0	1	1	1	0	0	0	0	0
24	5339	7.46	11.96	0.642	0	0	0	1	1	0	0	0	0	0
25	5286	1.47	7.16	0.495	4	0	1	0	0	0	0	0	0	0
26	5258	13.04	15.64	0.452	2	0	0	0	0	0	0	0	0	0
27	5126	1.00	23.00	0.552	2	0	0	0	1	0	0	0	0	0
28	5098	11.92	13.44	1.492	1	1	0	0	0	0	0	0	0	0
29	5047	13.14	13.24	0.613	0	1	1	0	0	0	0	0	0	0
30	4615	1.00	23.00	0.231	2	0	0	0	0	0	0	0	1	0
31	4582	9.89	15.55	0.269	0	0	2	2	0	0	0	0	0	0
32	4396	5.85	6.68	1.193	1	1	0	0	0	0	0	0	0	0
33	4331	19.42	19.93	1.125	1	1	0	0	0	0	0	0	0	0
34	3551	13.07	19.01	0.894	0	0	0	1	0	0	0	0	0	0
35	3532	20.07	23.50	0.264	0	0	2	2	0	0	0	0	0	0

Table A.21: Big Scenario Mission Characteristics: Missions 1-35

Mission	Value	Mission Time Window		Amount of Time on Station Required	Number of Aircraft Required									
		Earliest Start Time	Latest Start Time		Strike-F	Attack-A	Air-to-Air	Bomber-B	SEAD	Strike Fighter Type 1	Strike Fighter Type 2	Bomber Type 1	SEAD Fighter	Other SEAD
36	3510	8.95	12.34	0.893	4	0	1	1	1	0	0	0	0	0
37	3363	1.03	5.58	0.822	4	0	1	1	1	0	0	0	0	0
38	3242	8.19	8.30	1.343	0	1	0	0	0	0	0	0	0	0
39	3197	7.42	8.73	0.924	0	1	0	0	0	0	0	0	0	0
40	3092	21.33	23.50	0.500	4	0	1	0	0	0	0	0	0	0
41	2292	2.78	3.69	0.714	0	1	0	0	0	0	0	0	0	0
42	1789	18.42	23.50	0.498	2	0	0	0	0	0	0	0	1	0
43	1608	7.01	11.69	0.737	2	0	0	1	0	0	0	0	0	0
44	1284	11.39	17.09	0.788	2	0	0	0	1	0	0	0	0	0
45	980	12.15	16.10	0.863	4	0	1	0	0	0	0	0	0	0
46	935	7.02	12.16	0.951	2	0	0	0	0	0	0	0	0	0
47	919	1.00	23.00	0.323	4	0	1	0	0	0	0	0	0	0
48	866	7.13	10.14	0.457	2	0	1	0	0	0	0	0	0	0
49	859	4.99	7.42	0.354	2	0	0	1	1	0	0	0	0	0
50	802	1.32	6.08	0.884	2	0	0	0	0	0	0	0	0	0
51	749	1.00	23.00	0.697	2	0	0	0	1	0	0	0	0	0
52	748	9.99	10.24	1.332	0	1	0	0	0	0	0	0	0	0
53	704	9.80	15.02	0.178	2	0	0	0	0	0	0	0	0	0
54	654	1.00	23.00	0.474	2	0	0	0	1	0	0	0	0	0
55	632	21.15	23.50	0.500	2	0	0	1	0	0	0	0	0	0
56	621	5.94	7.19	0.657	0	2	0	0	0	0	0	0	0	0
57	597	14.98	17.26	0.942	2	0	0	0	0	0	0	0	0	0
58	593	18.54	21.60	0.831	4	0	1	0	0	0	0	0	0	0
59	550	13.48	18.01	0.516	2	0	0	1	0	0	0	0	0	0
60	521	1.00	23.00	0.491	0	0	0	1	1	0	0	0	0	0
61	428	17.31	21.84	0.411	4	0	1	0	0	0	0	0	0	0
62	414	19.51	23.50	0.414	2	0	0	0	0	0	0	0	1	0
63	382	3.21	5.14	0.532	0	1	0	0	0	0	0	0	0	0
64	381	6.63	9.20	0.548	0	0	0	1	1	0	0	0	0	0
65	309	1.00	23.00	0.301	0	0	0	1	1	0	0	0	0	0
66	306	15.88	20.31	0.342	2	0	0	0	0	0	0	0	1	0
67	304	10.62	14.03	0.761	2	0	1	0	0	0	0	0	0	0
68	257	1.00	23.00	0.625	2	0	0	0	0	0	0	0	1	0
69	234	1.00	23.00	0.164	2	0	0	0	0	0	0	0	0	0
70	233	13.30	16.59	0.795	4	0	1	0	0	0	0	0	0	0

Table A.22: Big Scenario Mission Characteristics: Missions 36-70

Mission	Value	Mission Time Window		Amount of Time on Station Required	Number of Aircraft Required									
		Earliest Start Time	Latest Start Time		Strike-F	Attack-A	Air-to-Air	Bomber-B	SEAD	Strike Fighter Type 1	Strike Fighter Type 2	Bomber Type 1	SEAD Fighter	Other SEAD
71	165	17.68	23.07	0.683	2	0	0	0	0	0	0	0	1	0
72	161	3.05	3.89	1.434	0	1	0	0	0	0	0	0	0	0
73	143	8.10	8.18	1.340	0	1	1	0	0	0	0	0	0	0
74	96	1.00	23.00	0.427	2	0	1	0	0	0	0	0	0	0
75	89	3.65	5.97	0.786	2	0	0	0	0	0	0	0	1	0
76	88	11.39	15.60	0.425	2	0	0	0	0	0	0	0	1	0
77	88	2.98	5.15	0.511	4	0	1	0	0	0	0	0	0	0
78	84	4.11	7.20	0.608	4	0	0	0	0	0	0	0	0	0
79	78	15.83	18.65	0.593	2	0	0	0	1	0	0	0	0	0
80	78	10.13	15.64	0.558	2	0	0	0	0	0	0	0	1	0
81	72	1.00	23.00	0.303	2	0	0	0	0	0	0	0	0	0
82	70	5.46	7.14	0.598	1	1	0	0	0	0	0	0	0	0
83	69	1.00	23.00	0.976	2	0	0	0	0	0	0	0	0	0
84	69	15.79	17.59	1.459	0	2	0	0	0	0	0	0	0	0
85	66	17.47	22.88	0.738	2	0	0	0	1	0	0	0	0	0
86	66	1.00	23.00	0.817	4	0	1	0	0	0	0	0	0	0
87	58	1.00	23.00	0.475	2	0	0	1	0	0	0	0	0	0
88	55	13.16	18.15	0.286	0	0	0	1	0	0	0	0	0	0
89	54	12.65	16.04	0.355	2	0	0	1	0	0	0	0	0	0
90	38	1.56	4.06	0.916	2	0	0	0	0	0	0	0	1	0
91	32	1.00	23.00	0.338	2	0	0	0	0	0	0	0	0	0
92	31	10.74	16.45	0.456	2	0	1	0	0	0	0	0	0	0
93	30	1.47	5.74	0.597	1	0	0	0	0	0	0	0	0	0
94	27	11.74	17.20	0.486	2	0	0	0	0	0	0	0	0	0
95	23	14.26	17.81	0.914	2	0	0	0	0	0	0	0	1	0
96	23	2.07	4.68	0.427	2	0	1	0	0	0	0	0	0	0
97	14	1.00	23.00	0.884	2	0	0	1	1	0	0	0	0	0
98	14	1.00	23.00	0.299	2	0	0	1	0	0	0	0	0	0
99	8	11.95	17.36	0.268	2	0	0	1	0	0	0	0	0	0
100	3	1.07	3.21	0.611	2	0	0	0	0	0	0	0	1	0

Table A.23: Big Scenario Mission Characteristics: Missions 71-100

THIS PAGE INTENTIONALLY LEFT BLANK

Appendix B

Military Acronyms

ACO	Airspace Control Order
ADP	Air Defense Plan
AOC WS	Air Operations Center Weapons System
AOR	Area Of Responsibility
AR	Air Refueling
ATO	Air Tasking Order
CAS	Close Air Support
COA	Course Of Action
C2	Command and Control
DCA	Defensive Counterair
FrOB	Friendly Order of Battle
ISR	Intelligence, Surveillance, and Reconnaissance
ITO	Integrated Tasking Order
JAOC	Joint Aerospace Operations Center
JAOP	Joint Air Operations Plan
JFACC	Joint Forces Air Component Commander
JFC	Joint Forces Commander
JIPCL	Joint Integrated Prioritized Collection List
JIPTL	Joint Integrated Prioritized Target List
MAAP	Master Air Attack Plan
MAAPTK	Master Air Attack Planning Toolkit
OCA	Offensive Counterair
RPA	Remotely Piloted Aircraft
SEAD	Suppression of Enemy Air Defenses
TNL	Target Nomination List
TST	Time Sensitive Target
UAV	Unmanned Aerial Vehicle

THIS PAGE INTENTIONALLY LEFT BLANK

Bibliography

[1] E.M.L. Beale and J.M. Tomlin. Special facilities in a general mathematical programming system for nonconvex problems using ordered sets of variables. *Operational Research*, 69:447–454, 1970.

[2] Dimitris Bertsimas, David Culver, and Stephen Kolitz. Addressing the fog of war in reconnaissance operatons: A robust optimization approach. *MORS (submitted)*, 2013.

[3] Dimitris Bertsimas and John N. Tsitsiklis. *Introduction to Linear Optimization*. The Art of Computer Programming. Athena Scientific and Dynamic Ideas LLC, Belmont, Massachusetts, 1997.

[4] Jeff Bezanzon, Stefan Karpinski, Viral Shah, and Alan Edelman. Julia: A fast dynamic language for technical computing. April 2012.

[5] David A. Brumbaugh. The parallel Air Tasking Order: Reducing the size of the Air Operations Center. Technical report, Science Applications International Corporation (SAIC), Chantilly, Virginia, June 2004.

[6] Corban H. Bryant. Robust planning for Effects Based Operations. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2006.

[7] Kevin Conner, Paul Lambertson, and Matthew Roberson. Analyzing the Air Operations Center (AOC) Air Tasking Order (ATO) using Theory Of Constraints (TOC). Master's thesis, Air Force Institute of Technology, Wright Patterson AFB, Ohio, June 2005.

[8] David M. Culver. Robust reconnaissance asset planning under uncertainty. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2013.

[9] Davi Rogerio da Silva Castro. Optimization models for allocation of air strike assets with persistence. Master's thesis, Naval Postgraduate School, Monterey, California, December 2002.

[10] Christodoulos A. Floudas and Xiaoxia Lin. Continuous-time versus discrete-time approaches for scheduling of chemical processes: a review. *Computers and Chemical Engineering*, 28:21092129, July 2004.

[11] Christodoulos A. Floudas and Xiaoxia Lin. Mixed Integer Linear Programming in process scheduling: Modeling, algorithms, and applications. *Annals of Operations Research*, 139:131162, 2005.

[12] United States Air Force. Air Force doctrine document 2-6.2, July 1999.

[13] United States Air Force. Air Force doctrine document 1: Air force basic doctrine, organization, and command, October 2011.

[14] Brian J. Griggs. An air mission planning algorithm for a theater level combat model. Master's thesis, Air Force Institute of Technology, Wright Patterson AFB, Ohio, March 1994.

[15] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.

[16] Nicholas R. Jernigan. Multi-modal, multi-period, multi-commodity transportation: Models and algorithms. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2014.

[17] Miles Lubin and Iain Dunning. Computing in operations research using julia. *INFORMS Journal on Computing*, 27(2):238–248, 2015.

[18] Andres F. Merchan, Sara Velez, and Christos T. Maravelias. Tightening methods for continuous-time mixed-integer programming models for chemical product scheduling. *AIChE Joural*, 59(12), October 2013.

[19] United States Joint Chiefs of Staff. Joint publication 3-01: Countering air and missile threats, March 2012.

[20] United States Joint Chiefs of Staff. Joint publication 3-30: Command and control of joint air operations, February 2014.

[21] Dennis C. Panos. Approximate dynamic programming and aerial refueling. Master's thesis, Princeton University, Princeton, New Jersey, June 2007.

[22] Jonathan L. Paynter. Optimized border interdiction. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2014.

[23] Albert A. Robbert. Costs of flying units in Air Force active and reserve components. Technical report, RAND Corporation, 2013.

[24] Philemon Sakamoto. UAV mission planning under uncertainty. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2006.

[25] Intelligent Software Solutions. Master Air Attack Planning Toolkit user guide, April 2012.

[26] Mark J. Williams. Column generation approaches to the military airlift scheduling problem. Master's thesis, Massachusetts Institute of Technology, Cambridge, Massachusetts, June 2014.

[27] Brian Zacherl. Weapon target pairing; revising an Air Tasking Order in real-time. Master's thesis, Naval Postgraduate School, Monterey, California, September 2006.