NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

GENERATING EFFICIENT AND ROBUST SCHEDULES TO DELIVER BULK FUEL VIA AMPHIBIOUS CONNECTORS

by

Coleman Strickland

June 2018

Thesis Advisor:	Michael Atkinson
Second Reader:	Kyle Lin

Approved for public release. Distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704–0188

1. AGENCY USE ONLY *(Leave Blank)*	2. REPORT DATE	3. REPORT TYPE AND DATES COVERED
	June 2018	Master's Thesis 06-24-2016 to 06-15-2018

4. TITLE AND SUBTITLE

GENERATING EFFICIENT AND ROBUST SCHEDULES TO DELIVER BULK FUEL VIA AMPHIBIOUS CONNECTORS

5. FUNDING NUMBERS

6. AUTHOR(S)

Coleman Strickland

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Naval Postgraduate School
Monterey, CA 93943

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

N/A

10. SPONSORING / MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES

The views expressed in this document are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government. IRB Protocol Number: N/A.

12a. DISTRIBUTION / AVAILABILITY STATEMENT

Approved for public release. Distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT *(maximum 200 words)*

During an amphibious operation, transporting Marine Expeditionary Unit (MEU) resources from ship to shore is a burdensome endeavor full of risks and complexity. Current scheduling techniques are arduous and time consuming, resulting in inefficiencies in the transition of resources from sea to land. Planners need an effective tool to limit these inefficiencies in an effort to expedite and prioritize the movement of supplies. In 2017, Major Robert Christafore developed the MEU Amphibious Connector Scheduler (MACS) tool to optimize the delivery of bulk fuel to forces ashore. This thesis further improves the MACS model to develop models and algorithms capable of handling bulk fuel delivery across a variety of MEU scenarios. The improved MACS model takes into account multiple objectives, weather implications, and reliability issues which greatly enhance the MACS tool's ability to deal with real world situations. The updated model incorporates cost penalties, fuel efficiency factors, fuel consumption rates, and risk assessments to better facilitate planning.

14. SUBJECT TERMS

15. NUMBER OF PAGES 149

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	UU

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2–89)
Prescribed by ANSI Std. 239–18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited

GENERATING EFFICIENT AND ROBUST SCHEDULES TO DELIVER BULK FUEL VIA AMPHIBIOUS CONNECTORS

Coleman Strickland
Capt, USMC
B.S., United States Naval Academy, 2012

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS ANALYSIS

from the

NAVAL POSTGRADUATE SCHOOL
June 2018

Approved by: Michael Atkinson
Thesis Advisor

Kyle Lin
Second Reader

Distinguished Professor Patricia Jacobs
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

During an amphibious operation, transporting Marine Expeditionary Unit (MEU) resources from ship to shore is a burdensome endeavor full of risks and complexity. Current scheduling techniques are arduous and time consuming, resulting in inefficiencies in the transition of resources from sea to land. Planners need an effective tool to limit these inefficiencies in an effort to expedite and prioritize the movement of supplies. In 2017, Major Robert Christafore developed the MEU Amphibious Connector Scheduler (MACS) tool to optimize the delivery of bulk fuel to forces ashore. This thesis further improves the MACS model to develop models and algorithms capable of handling bulk fuel delivery across a variety of MEU scenarios. The improved MACS model takes into account multiple objectives, weather implications, and reliability issues which greatly enhance the MACS tool's ability to deal with real world situations. The updated model incorporates cost penalties, fuel efficiency factors, fuel consumption rates, and risk assessments to better facilitate planning.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

ACE	Air Combat Element
AOR	Area of Operations
B	Beaches
CE	Combat Element
CSV	Comma Separated Value
DNF	Dynamic Network Flow
E2O	Expeditionary Energy Office
FOB	Forward Operating Base
GCE	Ground Combat Element
LCAC	Landing Craft Air Cushion
LCAVAT	Landing Craft and Amphibious Vehicle Assignment Table
LCE	Logistics Combat Element
LCU	Landing Craft Utility
LHD	Landing Helicopter Dock
LPD	Landing Platform Dock
LSD	Landing Ship Dock
LST	Landing Sequence Table
LZ	Landing Zones
MACS	MEU Amphibious Connector Scheduler

MAGTF	Marine Air-Ground Task Force
MEB	Marine Expeditionary Brigade
MEU	Marine Expeditionary Unit
MIO	Maritime Interception Operations
MOC	Marine Corps Operating Concept
MOP	Measures of Performance
MTVR	Medium Tactical Vehicle Recovery
nm	nautical miles
OMFTS	Operational Maneuver from the Sea
SAM	Surface-to-Air Missile
SAT	Serial Assignment Table
SBLOM	Sea-Based Logistic Optimization Model
SIXCON	SIXCON Fuel Storage Modules
TBFDS	Tactical Bulk Fuel Delivery System
TRAP	Tactical Recovery of Aircraft and Personnel
UAV	Unmanned Aerial Vehicles
VBSS	Visit, Board, Search, and Seizure

Executive Summary

The current planning process for constructing connector delivery schedules to transport critical supplies and resources to shore is a burdensome process full of inefficiencies and frustrations for the Marine Expeditionary Unit (MEU) staff. A planning tool to develop connector delivery schedules is critical to the success of the MEU's high-tempo operations. The MEU Amphibious Connector Scheduler (MACS), developed by Major Robert Christafore in 2017, is a model to build minute-by-minute connector schedules to deliver bulk fuel from ship-to-shore. The MACS model features three unique modules: Quickest Flow, Assignment Heuristic, and Scheduler. In this thesis, we improve the MACS model to incorporate a variety of new constraints providing increased capability and flexibility to the MEU staff. Furthermore, we produce an in-depth analysis to integrate a mixture of threat concerns, weather conditions, and connector failure rates to analyze how effectively the schedules constructed by the improved MACS tool perform during uncertainties. We then apply techniques to build more robust schedules by adding connector runs.

This thesis focuses on the Quickest Flow module only and we introduce an assortment of constraints to better the MACS tool for real-world use. Our Quickest Flow module is a temporal network flow model to measure the amount of resources delivered to shore across a given time horizon. We reformulate existing constraints inside of Quickest Flow to more accurately reflect the movement of supplies across the land network. We implement a fuel constraint to calculate the amount of fuel consumed by the connectors during the course of a schedule, which produces schedules that are more fuel efficient. Moreover, we implement a technique to incorporate land node priorities and introduce a constraint to integrate equity across the land nodes. The improved MACS tool referred to as MACS2 is capable of constructing five unique schedules, each with their own strength. Furthermore, each of these schedules can be layered to provide a schedule tailored specifically for the user.

We study the effects of uncertainty in our MACS2 model through the application of two different analysis: 1) Initial Connector Availability and 2) Intra-Day Connector Failures. Analysis 1 examines the effectiveness of bulk fuel delivery during a MEU deployment, and presents a greater understanding of what the MEU staff can expect to deliver to shore under a variety of unpredictable situations. For this analysis, we construct a set of scenarios and

sub-scenarios that we apply to our planning tool. Each scenario features a land network with realistic fuel demands and connector assets. These scenarios create the baseline inputs into our analysis. To incorporate uncertainty, we develop a set of sub-scenarios to represent a multitude of weather concerns, threat conditions, and connector failure probabilities. Each sub-scenario is then applied to our baseline scenarios to study the effects of uncertainty. Analysis 1 allows us to assess how a variety of uncertain conditions degrades timely delivery of resources

Analysis 2 focuses on the impact of connector failures throughout the duration of a connector's schedule. This analysis incorporates failure rates, degradation factors, and repair times to accurately represent maintenance actions during the execution of a connector's schedule. Additionally, we study how padding extra connector runs to the original schedule can increase the probability of timely delivery of resources, if connectors are subject to failure during operation. Our assignment policy for adding extra connector runs across the connectors produces a 5% increase in the amount of fuel successfully delivered to shore.

A critical assumption in the MACS and MACS2 models is that all connectors of the same type carry the same amount of fuel. In MACS2, we model the fuel containers explicitly, and develop an algorithm to assign a fixed number of fuel containers to connectors, so that different connectors of the same type do not always carry the same amount of fuel. Our algorithm assigns containers to connectors in the most efficient way without significant increase in computation time.

While this thesis focuses on the delivery of one commodity, bulk fuel; Captain Matthew Danielson (2018) constructs a similar tool to build connector schedules across multiple commodities simultaneously. Utilizing a similar model building approach, Danielson's tool formulates minute-by-minute schedules for each connector type across all classes of supply. The combination of his model with the improvements made in this thesis, provides an effective tool to alleviate the planning burden of the MEU staff.

References

Christafore R (2017) Generating ship-to-shore bulk fuel delivery schedules for the marine expeditionary unit. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA, https://calhoun.nps.edu/handle/10945/55582.

Danielson ME (2018) Scheduling amphibious connectors to deliver multiple commodities. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA, Forthcoming, https://calhoun.nps.edu/.

THIS PAGE INTENTIONALLY LEFT BLANK

Acknowledgments

First and foremost, to my loving and always supportive wife, Paige: I would not be where I am today without your encouragement. You are an incredible woman, amazing wife, and loving mother, and I thank God every day that He guided me to you.

To Professors Michael Atkinson and Kyle Lin, I sincerely thank you for your leadership, patience, and guidance. I could not have asked for a better set of thesis advisors. Specifically, Professor Atkinson, I thank you for your engagement and diligence over the past year. I would not be here without you.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1: Introduction

Originating from the Marine Corps' amphibious accomplishments during World War II, today's Marine Expeditionary Unit (MEU) is a multi-mission-capable, forward-deployed force small enough to respond quickly to emerging threats but large enough to provide substantial military strength when necessary. Consisting of roughly 2,000 Marines, three ships, and a multitude of tactical vehicles and aircraft, the MEU's ability to rapidly deploy is achieved by its relatively small footprint and seabase. One of the key capabilities of the MEU is its ability to deploy military forces quickly in a peaceful, chaotic, or uncertain environment in order to provide a force capable of an immediate response to any emerging crisis (Department of the Navy, Headquarters United States Marine Corps 2015). With speed and efficiency at the forefront of an effective MEU, any tool or program that expedites the planning processes involved with MEU operations adds critical value to the MEU's ability to achieve its mission. This thesis focuses on developing tools and algorithms to enhance and optimize the logistics planning process of MEU operations.

1.1 Motivation for Research

The Marine Corps Operating Concept (MOC) emphasizes the need to provide a flexible amphibious force, such as the MEU, that can rapidly deploy to meet ever-changing operating environments around the world (Department of the Navy, Headquarters United States Marine Corps 2016). The MEU achieves its status as a flexible amphibious force through its ability to seabase. Seabasing provides the MEU with the capability to not only deploy forces in an expedient manner, but also sustain those forces with organic assets. Seabasing, however, brings a specific set of logistical difficulties that are unique to the MEU. One of the primary concerns is the mode in which logistical supplies must travel to forces ashore. MEU planners are restricted to using smaller transport ships and aircraft, known as surface and air connectors, to carry supplies from ship-to-shore. This constraint results in the formulation of delivery schedules that must be de-conflicted and fit inside specific time constraints. Therefore, the planning, coordination, and scheduling of the delivery of supplies and equipment to forces ashore are critical in MEU sustainment operations.

The need for a planning tool to aid in the development of landing plans was identified by the National Research Council in 1999 as a recommendation in supporting the Operational Maneuver from the Sea (OMFTS) concept during fast-paced, complex support operations (National Research Council 1999). Almost 20 years later, the MEU staff continues to depend solely on experience and manpower to execute the transition of equipment and supplies from ship to shore. The staff is quickly overburdened with the sheer volume of decisions required to move thousands of tons of supplies via air and surface connectors from the amphibious assault ships to ground forces ashore. The immense amount of material coupled with an intense operating environment sets conditions for inefficiencies to develop in the landing plan, further hampering the movement of supplies. Without the aid of a planning tool, planners develop a schedule that is feasible but has little regard for efficiency of effectiveness. Moreover, logistical planners must repeat the planning process if the plan changes or encounters problems.

The development of a mechanism to alleviate the responsibility of creating an optimal landing plan is critical to saving valuable time and resources during an amphibious operation. This thesis focuses on optimizing delivery schedules for the movement of resources from ship-to-shore via air and surface connectors through a decision support tool. Focusing primarily on sustainment operations, the planning tool developed not only optimizes delivery schedules for amphibious connectors but also alleviates the planning burden for the MEU staff.

1.2 Overview of Amphibious Operations

Through the use of air, land, and sea, an amphibious operation consists of a military operation that is conducted from the sea to carry out an amphibious landing of military forces ashore (Joint Chiefs of Staff 2014). The four types of amphibious operations are Amphibious Assault, Amphibious Raid, Maritime Interception Operations (MIO) / Visit, Board, Search, and Seizure (VBSS), and Advance Force Operations (Department of the Navy, Headquarters United States Marine Corps 2015). In addition, the MEU is responsible for providing expeditionary support to other military operations to include Humanitarian Assistance, Tactical Recovery of Aircraft and Personnel (TRAP), and Airfield/Port Seizure (Department of the Navy, Headquarters United States Marine Corps 2015). The ability to accomplish all of these tasks in any location within a limited amount of time brands the

MEU as a highly valuable strategic asset. Additionally, all of these missions take place from the seabase, to include the sustainment operations needed to support military forces conducting operations ashore.

Figure 1.1 illustrates an overview of how an amphibious operation is logistically supported. The picture shows a MEU stationed offshore consisting of three ships. The MEU conducts an offload of ground forces in the area and establishes a set of Beaches (B) (B1 and B2) as well as Landing Zones (LZ) (LZ1 and LZ2) as points of interest ashore. These points of interest contain a light logistical footprint for ease of movement and speed; therefore, the beaches and landing zones' logistical needs are fully dependent on the MEU's ability to flow resources to these areas. To transport resources to the points of interest, the MEU contains specific connectors (surface and air) to support the logistical demands ashore. Surface connectors can access the beaches while the air connectors can access the landing zones. The air connectors can also access a beach if the beach is constructed to support air operations.

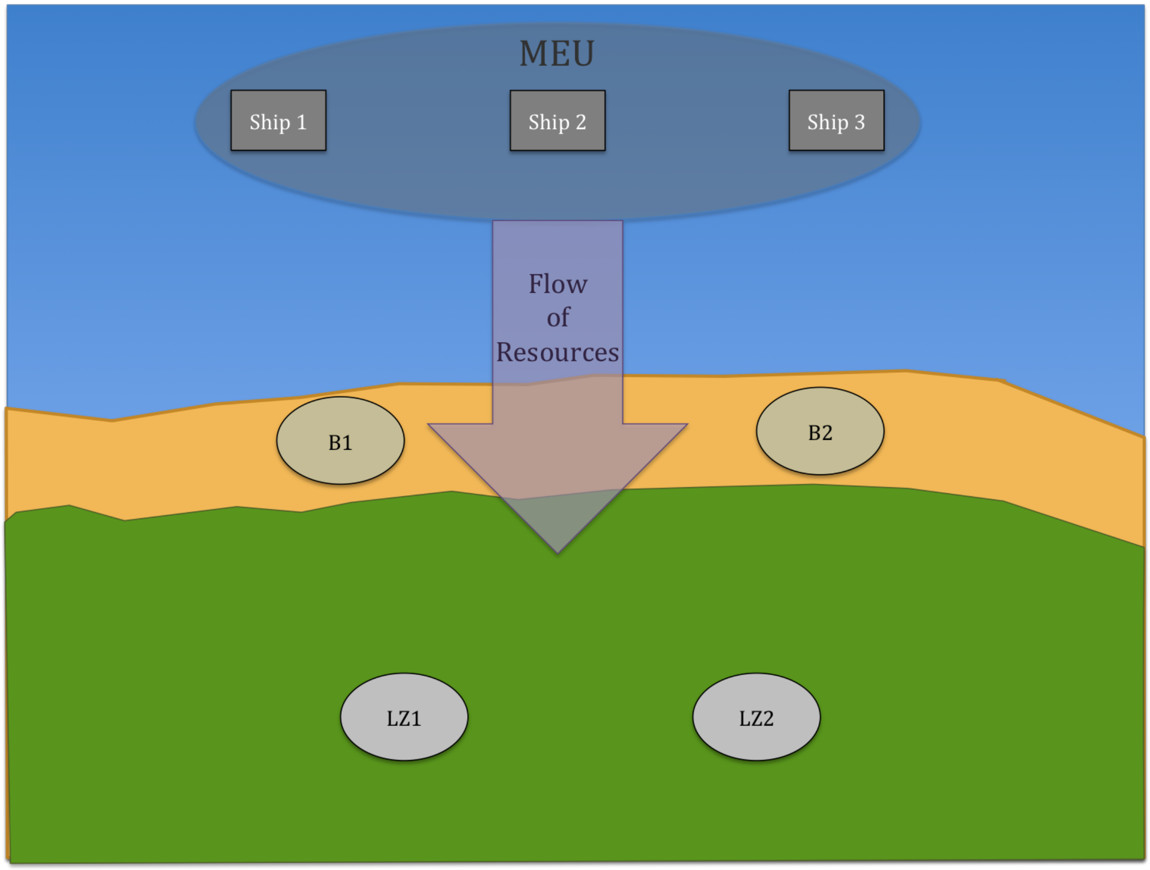


Figure 1.1. Flow of Resources

For the purposes of this thesis, our model will primarily focus on one resource, bulk fuel; however, the model can be generalized to any class of supply. Additionally, this thesis will work with a variety of MEU assets such as surface and air connectors. Specifically, the Landing Craft Air Cushion (LCAC), Landing Craft Utility (LCU), CH-53 Sea Stallion, and the MV-22 Osprey. A more thorough overview of the MEU equipment incorporated into this thesis can be found in Appendix A.

1.3 Original MACS Tool

In this thesis, we extend the work of Major Robert Christafore in which a logistical planning tool was developed to optimize the movement of resources from the MEU to points of interest ashore. The MEU Amphibious Connector Scheduler (MACS) optimizes connector schedules to move bulk fuel from ship-to-shore across a given time horizon. The MACS tool utilizes three different models: Quickest Flow linear program, Assignment Heuristic, and the Scheduler linear program. These sub-models work successively to produce an efficient and detailed schedule for the surface and air connectors. A more thorough description of the MACS tool is described in Chapters 2 and 3 of this thesis.

The output from the MACS tool is a minute-by-minute delivery schedule of each connector run. Figure 1.2 depicts an example output from the MACS tool from Christafore (2017). The example features the transportation schedule developed for one of the surface connectors, in this case the LCAC. In the scenario, the LCAC makes eight runs to B1 and four runs to B2 over a time horizon of eight hours. We designate a run to equal a single round trip from the ship to the destination node. One can see the minute-by-minute schedule is broken into specific time events (i.e., Start Load, Depart Ship, Arrive at Beach, Start Unload, Depart Beach, and Arrive at Ship). The schedule also shows the ship being departed from, the run number of that ship, and the final destination of the run. The LCAC schedule is feasible under the model assumptions and can be easily implemented by the MEU logistical planners.

LCAC Schedule

Ship	Run	Destination	Start Load	Depart Ship	Arrive at Beach	Start Unload	Depart Beach	Arrive at Ship
LHD	1	B1	0.0	30.0	100.0	100.0	125.0	195.0
LHD	2	B1	30.0	60.0	130.0	130.0	155.0	225.0
LHD	3	B1	60.0	90.0	160.0	160.0	185.0	255.0
LHD	4	B1	195.0	225.0	295.0	295.0	320.0	390.0
LHD	5	B2	225.0	255.0	345.0	345.0	370.0	460.0
LHD	6	B2	255.0	285.0	375.0	375.0	400.0	490.0
LPD	1	B1	0.0	30.0	90.0	90.0	115.0	175.0
LPD	2	B1	30.0	60.0	120.0	120.0	145.0	205.0
LPD	3	B1	175.0	205.0	265.0	265.0	290.0	350.0
LPD	4	B1	205.0	235.0	295.0	295.0	320.0	380.0
LPD	5	B2	350.0	380.0	460.0	460.0	485.0	565.0
LPD	6	B2	380.0	410.0	490.0	490.0	515.0	595.0

Figure 1.2. Example LCAC Schedule from the MACS Tool. Source: Christafore (2017)

In Christafore (2017), the scenario considered is a MEU consisting of three ships (Landing Helicopter Dock (LHD), Landing Ship Dock (LSD), and Landing Platform Dock (LPD)) conducting sustainment operations across two beaches and two landing zones. While the MACS tool effectively produces a connector schedule to deliver bulk fuel in the given time horizon, there are limitations in its capabilities and room for improvement. For example, fuel efficiencies of connectors, prioritization of fuel demands across the land nodes, and a more realistic model for the movement of fuel across the land nodes are all subjects the MACS tool does not consider. We extend the MACS tool to account for these real world concerns. We incorporate cost penalties, fuel efficiency factors, fuel consumption rates, and equity considerations across all land nodes to better facilitate planning considerations under real world conditions. Furthermore, we consider the reallocation of bulk fuel assets across the connector types to find a more efficient use of MEU assets when developing the connector schedules.

This thesis also focuses on expanding the scenario provided in Christafore (2017). For example, the given scenario assumes all connectors are operational over the entire duration of the time horizon, and the weather facilitates the usage of all connector types. We test the improved model through building a set of additional scenarios that incorporate real world fuel demands, weather considerations, and connector failure rates to determine the constraints and limitations of the improved tool. Through testing the updated model on a multitude of scenarios, we make improvements to the model to produce a tool more applicable for real-world use.

1.4 Research Objectives

The following questions will guide the research and model development of improving the MACS tool. These questions focus on specific limitations of the current tool with an emphasis on building unique schedules specific to the user's needs. Additionally, we investigate the effects of real-world uncertainties and how these situations can be mitigated through additions made to the MACS tool.

1. **Given limited resources are available such as fuel and time, how can we improve the MACS model to better facilitate the movement of resources to shore?**
   The capability to produce various schedules to account for fuel efficiencies, equity,

and priorities throughout the land network provides the MEU staff with greater flexibility for sustaining forces ashore. A tool allowing the user to construct a situation specific schedule is highly valuable. For example, at the beginning of an amphibious offload, the goal of MEU planners is to satisfy the fuel requirement ashore as quickly as possible in order to sustain operations inland. Minimum delivery times are critical in this scenario and are extremely important in order to not limit the military activities ashore. Conversely, if the amphibious operation has carried on for several days or weeks and fuel supply is limited, planners potentially need a delivery schedule with the focus of satisfying the fuel demand ashore based on fuel efficiencies of the connectors. This scenario comes into play when fuel resources are running low and a fuel resupply is unavailable for the near future.

2. **How can the tool consider a realistic flow of resources through the land network in which the fuel capacities are restricted based on land connectors ashore?**
The movement of bulk fuel from ship-to-shore is only the first piece in the transition of the resource. The ability for a land node to receive more fuel is based on the consumption rate of fuel at that node and the ability for that node to push fuel to where it is needed most. Pushing fuel further inland is primarily dependent on connectors ashore such as land vehicles. A more realistic approach to modeling the flow of resources across the land network is critical in understanding how supplies is delivered to the end user.

3. **How can the MACS model account for unforeseen issues such as connector failures and weather considerations when constructing connector schedules?**
Focusing on incorporating maintenance activities, either scheduled or unexpected, is critical in building a scheduler that is robust and flexible. Air and surface connectors are highly susceptible to failure the more they are utilized and a planning tool not capable of handling these considerations is potentially useless to a MEU planning staff. Additionally, the connectors cannot operate in specific environments in which the sea state is too high to use surface connectors or visibility requirements keep aircraft grounded and unable to fly. A tool that incorporates these considerations and finds an efficient schedule based on constrained connector usage is imperative.

The first two questions consider the development and formulation of the model. These questions drive the additions made to the MACS tool in Chapter 3 and Chapter 6. The third question correlates to the application of the model under realistic conditions, and how effective our updated tool is at handling uncertainties during real-world situations. We provide an in-depth analysis of this study in Chapter 5.

1.5 Related Research

The development of a tool to optimize the movement of assets to shore from a sea base has been covered in a multitude of different research areas. Ranging from minimizing fuel consumption rates to developing schedules to optimize the transport of people during humanitarian assistance operations, logistics staffs are focused on finding innovative ways of easing the problem of scheduling movements.

Ward (2008) constructs a mixed integer program to optimize the movement of personnel to/from a hospital ship during humanitarian assistance operations. Using a multi-objective optimization problem, Ward minimizes the transportation costs associated with moving medical personnel and patients from ship-to-shore and vice versa. Utilizing a "desired arrival time" and calculating the operating costs associated with moving the required personnel to their destinations, Ward's model seeks to minimize both of these factors to determine an optimal schedule (Ward 2008). Using the idea of cost-minimization, we implement a similar approach to maximize the delivery of fuel to shore in the quickest time window possible.

In his thesis, Viado (1999) develops the Sea-Based Logistic Optimization Model (SBLOM). The model is an integer program producing optimal schedules for the movement of bulk fuel from ship-to-shore. The model's objective function minimizes the initial gallons of fuel at the land nodes ashore and incorporates a "shipping cost" to build a "just-in-time" fuel delivery approach (Viado 1999). Viado's model builds a list of sorties for each of the connectors in order to develop an hourly schedule to transport fuel to shore. While effective, Viado's model lacks the ability to build a detailed minute-by-minute schedule that can be implemented by a planning staff. More importantly, the model is significantly hampered by its inability to scale to larger amphibious operations. For example, a single scenario can take up to 68 minutes to solve (Viado 1999). This time constraint has negative implications when producing multiple schedules due to changing conditions in the operating environment. Our planning tool will avoid integer programming in order to construct delivery schedules in a reasonable time period.

In December of 2013, Cohort 311-122O from the Naval Postgraduate School analyzed the reduction of fuel consumption for ship-to-shore connectors for a Marine Expeditionary Brigade (MEB) (Skahen et al. 2013). Examining a variety of scenarios including sea state conditions, the cohort builds a stochastic simulation to study the outcomes of fuel usage during these scenarios. The most profound outcome of this study is the use of the LCAC should be limited when planning staffs want to reduce fuel consumption during amphibious operations (Skahen et al. 2013). Moreover, the dominant factor for fuel consumption when the amphibious ship is further than 11 nautical miles from shore is sea state (Skahen et al. 2013). We use the outcome from this thesis to implement specific fuel consumption strategies in order to produce more fuel friendly schedules, and consider both distance and sea state conditions simultaneously.

Herendeen (2015) considers the movement of fuel from ship-to-shore to replenish the fuel requirement for the MEU's Ground Combat Element (GCE) (Herendeen 2015). His thesis uses the Marine Air-Ground Task Force (MAGTF) Power and Energy Model to construct fuel consumption data for further analysis. Herendeen concludes in order to support the GCE during its highest demand of fuel, 2.5 percent of the MEU's connector capacity must be solely dedicated to moving fuel ashore (Herendeen 2015). Again, this outcome considers the GCE only and does not incorporate fuel demand from the Air Combat Element (ACE), Logistics Combat Element (LCE), or Combat Element (CE). We use the outputs from Herendeen's results as inputs into the MACS tool to further develop our model to maximize connector efficiency.

In 1955, Harold Kuhn introduced the Hungarian Algorithm which optimizes the assignment of n persons to n jobs using the summation of scores for each assignment (Kuhn 1955). Similar to the traveling salesman problem, assignment problems contain a variety of feasible solutions but only a few or one optimal solution. Kuhn utilizes a cost matrix to distribute the assignment of each person to each job (Kuhn 1955). Noting the score of each assignment, the algorithm reassigns individuals to jobs until the maximum score is achieved. We utilize a similar approach when allocating bulk fuel containers to connectors and connectors to land nodes. We build a cost matrix associated with each connector-land node pair, then seek to minimize the cost (number of runs) in order to find an efficient solution.

The development of the MACS tool provides the ability to integrate with the current planning documents created and utilized by the MEU planners. Currently the MEU uses a multitude of documents to track the flow of personnel and resources from ship-to-shore. These documents include the Landing Craft and Amphibious Vehicle Assignment Table (LCAVAT), the Serial Assignment Table (SAT), and Landing Sequence Table (LST) (Joint Chiefs of Staff 2014). All of these documents are further discussed in Joint Publication 3-02.1 (Joint Chiefs of Staff 2014). Each of these documents play a critical role in the formulation of an adequate and organized schedule for the MEU connectors. We use the inputs to these documents to guide the outputs in the MACS tool. Through the use of these documents we are able to tailor the MACS tool's outputs to be implemented easily into the MEU staff's planning documents.

1.6 Thesis Organization

This thesis is organized into seven chapters. Chapter 2 is an evaluation of the current MACS tool using a variety of scenarios to test its capabilities and limitations. The goal of Chapter 2 is to identify critical shortfalls with the MACS model in order to develop a more robust and capable tool. Chapter 3 is centered on the changes made to the MACS tool and the mathematical formulation of those changes. Chapter 4 produces a comparison of the MACS tool against the changes made from Chapter 3, illustrating the MACS2 model produces more capable schedules applicable to the MEU staff. Chapter 5 offers two different uncertainty analyses of the MACS2 model and implements new approaches to further build robustness in the connectors' schedule. Chapter 6 features an addition to the MACS2 tool to allow MEU planners the ability to reallocate fuel container assets across different quantities of connectors in an efficient manner. Finally, Chapter 7 reviews the work completed and provides suggestions for future research.

CHAPTER 2:
Evaluation of the MACS Tool

The current MACS tool produces a minute-by-minute feasible schedule for pushing bulk fuel from the MEU seabase to land nodes ashore. While the current model is effective, the tool has limited capacity to effectively handle a variety of situations. In this chapter, we provide an initial evaluation of the model through a few examples to highlight some of the model's shortcomings. Furthermore, we present a brief overview of the components of the baseline MACS tool including the model's inputs and outputs. We defer a more thorough description of the models until Chapter 3.

2.1 Description of the MACS Tool

The MACS tool contains three models that work consecutively to output the connector schedules. Each model performs a specific role that feeds into the next model for further processing and schedule development. Below is a discussion of each program and its particular role in the MACS tool. Additionally, each model is explained in further detail in Chapter 3 of Christafore (2017) and we provide more technical details in the next chapter.

2.1.1 MACS Inputs

The inputs to the MACS tools are given in six Comma Separated Value (CSV) files and contain all the necessary information to formulate a connector schedule. Utilizing a set of Python scripts for bookkeeping, the files are used to build the network considered and to associate connector capacities and quantities, distances, and node locations inside of the given network. Each of these files is further outlined in Section 4.1 of Christafore (2017). These inputs feed directly into the MACS tool in order to output the desired schedule for each connector from all ships to all land nodes in the network.

The primary inputs to the MACS tool are outlined below. These inputs are used to maximize the throughput of fuel across all the connectors in order to satisfy the fuel demand at the locations ashore.

1. **Quantity/Type of Connector**
   The quantity and type of connector available for each ship type is inputted along with the connector's bulk fuel carrying capacity and average movement speed.

2. **Ship to Land Network**
   The ship to land network consists of multiple inputs to enable the tool to properly define the network considered. Inputs include the number of land nodes, the distances between each land node and all other land nodes, and the distances between a land node and each ship in the seabase. Also inputted are the numbers of specific landing spots available for each connector at the land nodes. For example, B1 might have two LCAC landing spots and one LCU landing spot while LZ1 could have two CH-53 landing spots and two MV-22 landing spots.

3. **Fuel Demands and Capacities Ashore**
   The demand of fuel required by each of the land nodes ashore is inputted in gallons along with the land nodes' fuel capacity.

4. **Load/Unload Times of Connectors**
   To properly calculate an accurate schedule, the amount of time it takes to load/unload the bulk fuel containers on each connector is inputted. These times are dependent on the bulk fuel carrying capacity of each connector. For example, an LCU that can carry 1,800 gallons of bulk fuel will take longer to load than an LCU carrying 900 gallons of bulk fuel.

5. **Time Horizon**
   The time horizon input consists of the number of hours in which the schedule must encompass. This value is dependent on what the user defines as operating hours and is used to restrict the schedule to a specified time horizon.

2.1.2 Quickest Flow Model

The Quickest Flow model maximizes the throughput of fuel across all connectors inside a given time horizon and is discussed in further detail in Chapter 3. Output from the Quickest Flow model is the total amount of fuel the model can transport from ship-to-shore inside of the given time window. The total amount of fuel pushed ashore is broken down by connector type through which the total number of runs for each connector is calculated. The final product from the Quickest Flow model is the total number of runs ashore by each connector for the given network. For example, three CH-53 runs to LZ1, one LCU run to B1, and three LCAC runs to B2.

2.1.3 Assignment Heuristic

With the number of runs calculated from the Quickest Flow model, the Assignment Heuristic orders these runs to make sense in a real world setting (Christafore 2017). The Assignment Heuristic removes the need to create an integer linear program with numerous integer variables, providing us the means to create a schedule in a reasonable amount of time. The heuristic seeks to meet the constraints of landing locations ashore as well as the limited space on the ship while simultaneously tracking the beach runs, ship departure runs, and ship return runs for each connector type. The heuristic assigns a connector run from the seabase to each run outputted from the Quickest Flow model. In its simplest terms, the heuristic assigns the connectors to their land nodes when the connector is loaded and ready to depart the ship. Output from the heuristic is a "rough" schedule that feeds into the third model of the MACS tool. A more detailed description of the Assignment Heuristic can be found in Section 3.2 of Christafore (2017).

2.1.4 Scheduler

Scheduler utilizes the output from the Assignment Heuristic to "smooth out" a feasible minute-by-minute schedule for each connector type. The Scheduler model uses six time events as a set of rules for ensuring specific actions stay in order. The six time steps include:

1. Start Loading
2. Depart Ship
3. Arrive at Land Node
4. Unload Cargo
5. Depart Land Node
6. Return to Ship

The Scheduler's objective function minimizes the mean completion time of each connector type, and the constraints guarantee all runs stay in a specific order. For example, a LCAC cannot arrive at a beach node until a beach spot is available or an aircraft cannot take off from the ship until it is loaded with cargo. The output from Scheduler is a succinct and feasible minute-by-minute schedule for each connector type consisting of the run number for each specific ship in the MEU, an example of this output appears in Figure 1.2 in Chapter 1. Appendix B provides the full formulation of the Scheduler model as depicted in Christafore (2017). For a more detailed discussion, see Section 3.3 of Christafore (2017).

2.2 Scenario Development

The MACS tool was initially evaluated in Christafore (2017) on a small network consisting of two beaches and two LZs with limited demand needs at those nodes. To more thoroughly evaluate the tool, we consider a variety of scenarios to understand the model's constraints and limitations. Characteristics of these scenarios encompass real world fuel demands, distances, and equipment utilized during amphibious operations. To implement a multitude of scenarios, we develop a Scenario Builder program to construct various scenarios to run through the MACS tool.

We define a set of Measures of Performance (MOP) to accurately compare and contrast different connector schedules. Currently, the MACS tool uses demand satisfaction as its primary MOP. The tool performs an initial check of land node demands and supply capabilities to determine if all demand can be satisfied at the land nodes with the given MEU equipment inputted in the CSV files. If the demand cannot be 100% fulfilled, the tool will not construct a delivery schedule. In our model, we relax this constraint to analyze what demand satisfaction can be achieved with the current equipment set-up inputted. Using only a single MOP, the MACS tool is limited in its ability to build an efficient schedule. For example, once the MACS tool can build a schedule to satisfy 100% of the demand ashore, the tool will only build a schedule that is fast enough to fit inside the given time window; therefore, it has little regard for building the fastest schedule achievable.

Our focus is taken a step further to look at multiple MOPs. These MOPs include: total amount of bulk fuel transported throughout the time horizon, percentage of demand satisfied at each land node, total number of runs ashore, and the total amount of fuel consumed by connectors during the schedule. Each MOP is outlined in greater detail below:

1. **MOP1: Total Amount of Bulk Fuel Transported**
   The total amount of bulk fuel delivered is the primary measurement of performance for each schedule. The model maximizes this value within the given constraints of the program. The total amount of bulk fuel transported allows us to easily contrast the performance of each schedule in order to understand the effects and results of the various schedules constructed.

2. **MOP2: Percentage of Demand Fulfilled at Each Land Node**
   What is the percentage of demand achieved at each land node in the network? In practice, a planner may want each node to receive a minimum amount of fuel (a percentage) to ensure every node in the network receives at least some level of resources. Measuring the percentage of demand that the tool is able to satisfy enables us to understand how effective the tool is at pushing fuel to each land node individually. This measurement gives us the capability of determining if the distribution of flow throughout the network is equal across all of the land nodes or if the tool is pushing the majority of flow to only a few nodes based on how the given network is arranged.

3. **MOP3: Number of Runs to Shore**
   Calculating the total number of runs ashore enables us to measure the efficiency of the scheduler. Can we push the same amount of fuel and satisfy the same percentage of demand ashore while reducing the number of total runs of the connectors? Efficiency is key when resources and equipment are in short supply, and the ability to determine if a more efficient schedule is capable of producing the same results is vital in measuring the performance of each schedule constructed.

4. **MOP4: Total Amount of Fuel Consumed by Connectors**
   Similar to measuring the total number of runs ashore, calculating the total amount of fuel consumed by the connectors in a given schedule allows us an additional measure of efficiency. Building a fuel consumption constraint into the model also gives us the capability to directly measure and control the amount of fuel consumed by each connector type.

2.3 Initial Evaluation

In this section, we present three scenarios to highlight the shortcomings with the current MACS model. Our analysis focuses primarily on the formulation and outputs of the Quickest Flow model. We utilize the baseline tool developed in Christafore (2017), and assume all weather and threat conditions facilitate the movement of resources ashore. Additionally, we assume no connector failures in our initial evaluation. In Chapter 5, we relax these assumptions in order to test the robustness of the updated tool and determine its legitimacy at handling real world situations. The following subsections outline each of the three initial scenarios considered and the MACS outputs for each scenario.

Each scenario features a different network configuration to include varying distances, connector quantities and capacities, land node types, and fuel demands ashore. To better understand the network, each scenario contains a table outlining the input factors of the scenario along with a picture to visualize the network. Each land node contains a combination of surface, land, or air connector spots. If a road or highway connects two nodes, the arc is identified as a "Land Connection" with the two nodes outlined in the table containing "land" spots. This description is important to note as the "Land Connection" serves as a means to utilize a land connector to transport fuel between the connected nodes.

2.3.1 Scenario A

We begin with Scenario A which features four land nodes: B1, LZ1, LZ2, and a Forward Operating Base (FOB). Inputs for the scenario appear in Table 2.1 along with an illustration of the network in Figure 2.1. Note that we specifically label where the Medium Tactical Vehicle Recovery (MTVR) is located in the model, the importance of this designation is further explained in the next section. We use a time horizon of eight hours for Scenario A, which produces 32 time steps of 15-minute increments.

Scenario A Inputs			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	4,500
LCU	Surface	1	1,800
CH-53	Air	1	2,400
MV-22	Air	4	800
MTVR	Land (FOB1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	5,000	FOB1
LZ1	Air	25,000	None
LZ2	Air	30,000	None
FOB1	Air, Land	30,000	B1

Table 2.1. Scenario A Inputs

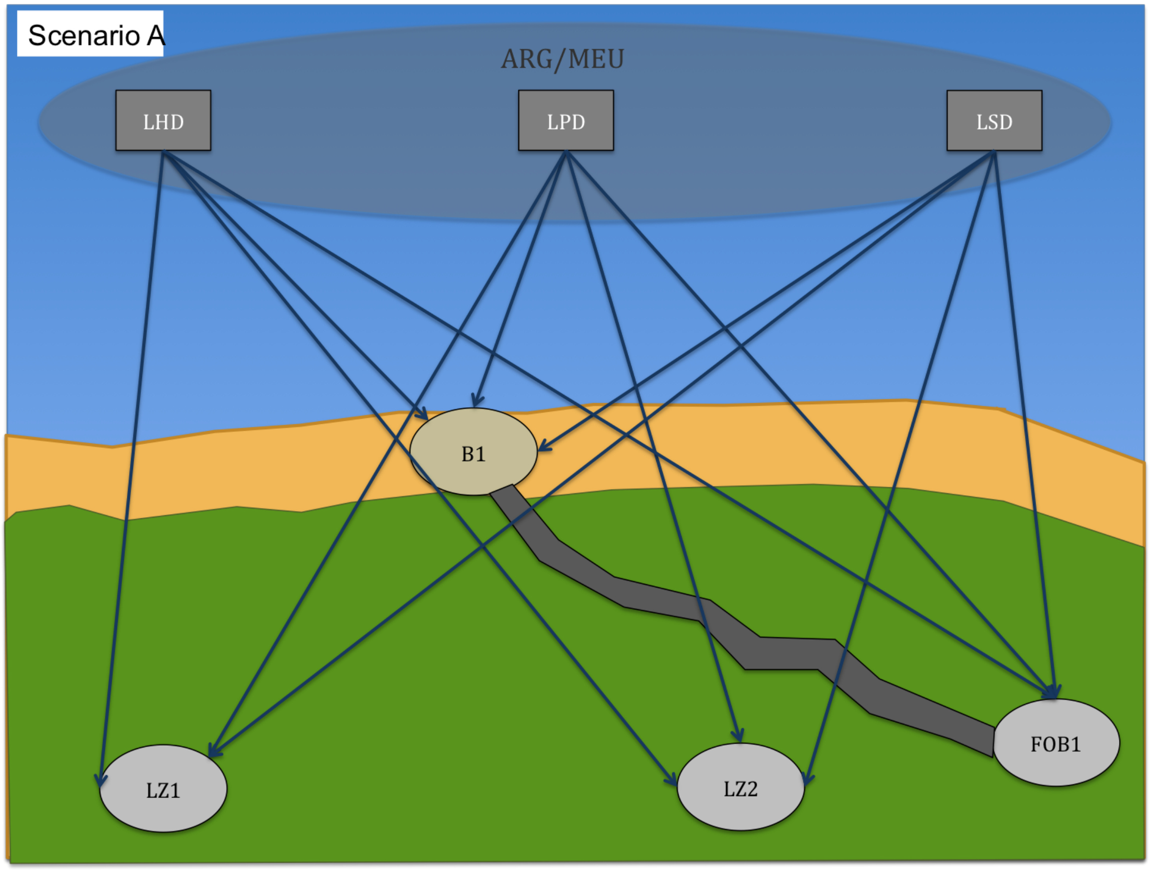


Figure 2.1. Scenario A Network Illustration

The MACS model formulates a feasible schedule for Scenario A in less than one minute. The MACS tool satisfies approximately 54% of the demand ashore for a total movement of 48,954 gallons of fuel. Table 2.2 shows the MOPs as well as the number of runs for each connector type.

MOPs			
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)
54.39	48,954	46	9,578

Connector Run Count		
Connector	Land Node	Run Count (Total=46)
LCAC	B1	2
LCU	B1	1
CH-53	LZ1	4
	LZ2	2
	FOB1	2
MV-22	LZ1	20
	LZ2	8
	FOB1	7

Table 2.2. Scenario A Output

We illustrate the fraction of fuel demand satisfied over time from the Quickest Flow model in Figure 2.2. It is noted all graphs in this thesis utilize the ggplot2 visualization package by Wickam and Chang (2016) in R. We see from the plot the majority of fuel pushed to the land nodes ashore occurs toward the end of the time horizon, with over 10,000 gallons of fuel pushed through the network in the final two time steps.

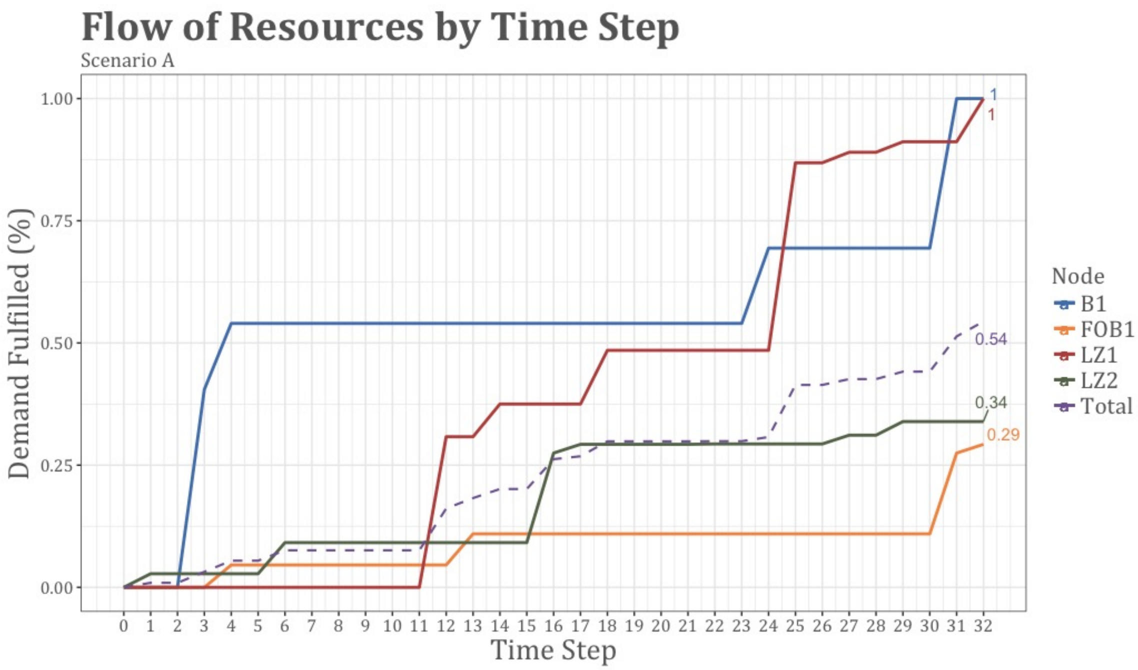


Figure 2.2. Scenario A Output - Flow of Resources by Time Step

Concerns with Scenario A's output are numerous. In the Quickest Flow model, there is no advantage in satisfying fuel demand ashore earlier in the time window; therefore, MACS may satisfy the majority of demand towards the end of the time window as seen in Figure 2.2. This is a flaw with the current Quickest Flow model. Currently, the Quickest Flow model's goal is to transfer fuel to the land nodes within the time horizon with no regard to early arrival. That is, as long as the demand is satisfied before the time horizon ends, MACS has successfully achieved its objective. In Chapter 3, we reformulate the objective function of the Quickest Flow model to reward faster delivery of resources ashore.

Another issue is the MACS tool assumes all connectors are operational for the duration of the scenario. Under Scenario A, the four MV-22s are scheduled to fly a total of 35 sorties, roughly 9 sorties each. The probability that this schedule can be executed without a MV-22 succumbing to maintenance failure is low. The capability to account for these unforeseen maintenance issues will produce a more realistic schedule that can be executed with relative confidence. We perform a thorough analysis on this issue in Chapter 5.

Currently MACS does not include the ability to prioritize fuel demands among the land nodes. The MACS tool simply finds the optimal solution to transport as much fuel as possible among all the land nodes in the network. For example, the demand for FOB1 and LZ2 is 30,000 gallons but only around 30% of the demand is fulfilled at these nodes, while B1's demand is 5,000 gallons and all of the demand is met for this node. By focusing on demand satisfied per node, we can accomplish two things: 1) ensure prioritized nodes receive as much fuel as possible, and 2) ensure there is a notion of "fairness" or a minimum demand satisfied at each node.

2.3.2 Scenario B

Scenario B contains the same inputs from Scenario A with one minor change: the air connector spots located at FOB1 are eliminated. This change ensures the fuel transported to FOB1 occurs with land connectors only. We continue to use an eight-hour time window with 32 time steps. The inputs to Scenario B along with the network illustration appear in Table 2.3 and Figure 2.3, respectively.

Scenario B Inputs			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	4,500
LCU	Surface	1	1,800
CH-53	Air	1	2,400
MV-22	Air	4	800
MTVR	Land (FOB1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	5,000	FOB1
LZ1	Air	25,000	None
LZ2	Air	30,000	None
FOB1	Land	30,000	B1

Table 2.3. Scenario B Inputs

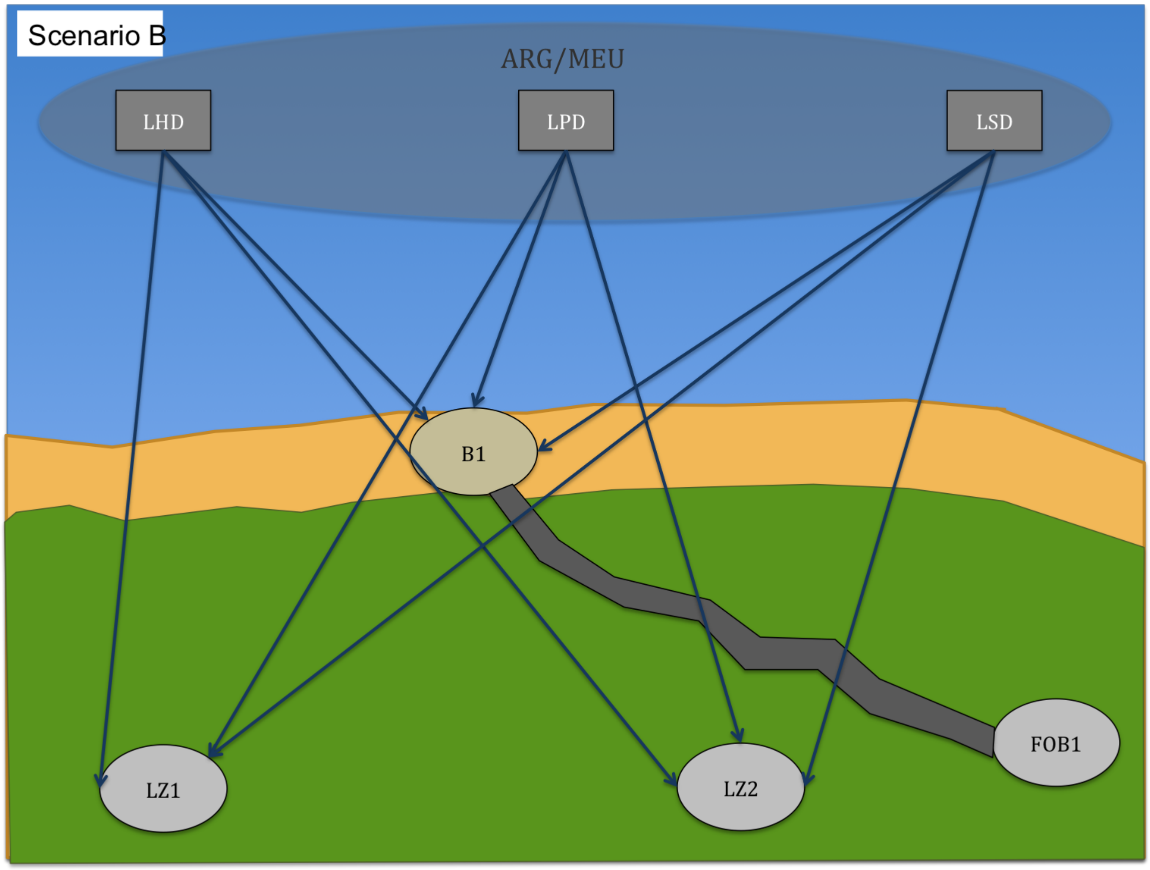


Figure 2.3. Scenario B Network Illustration

In this scenario, MACS transports 41,060 gallons of fuel to the land nodes satisfying approximately 45% of the demand. Comparing this output to Scenario A's 54%, Scenario B's demand satisfaction is lower because the seabase does not have direct access to FOB1 through air or surface connectors. Table 2.4 depicts the MOPs as well as the number of runs for each connector type.

MOPs			
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)
45.62	41,060	39	8,294

Connector Run Count		
Connector	Land Node	Run Count (Total=39)
LCAC	B1	2
LCU	B1	1
CH-53	LZ1	2
	LZ2	4
MV-22	LZ1	17
	LZ2	13

Table 2.4. Scenario B Output

Figure 2.4 illustrates the fraction of fuel demand fulfilled over time. Again, it is noted the majority of fuel pushed to the land nodes occurs toward the end of the time horizon.

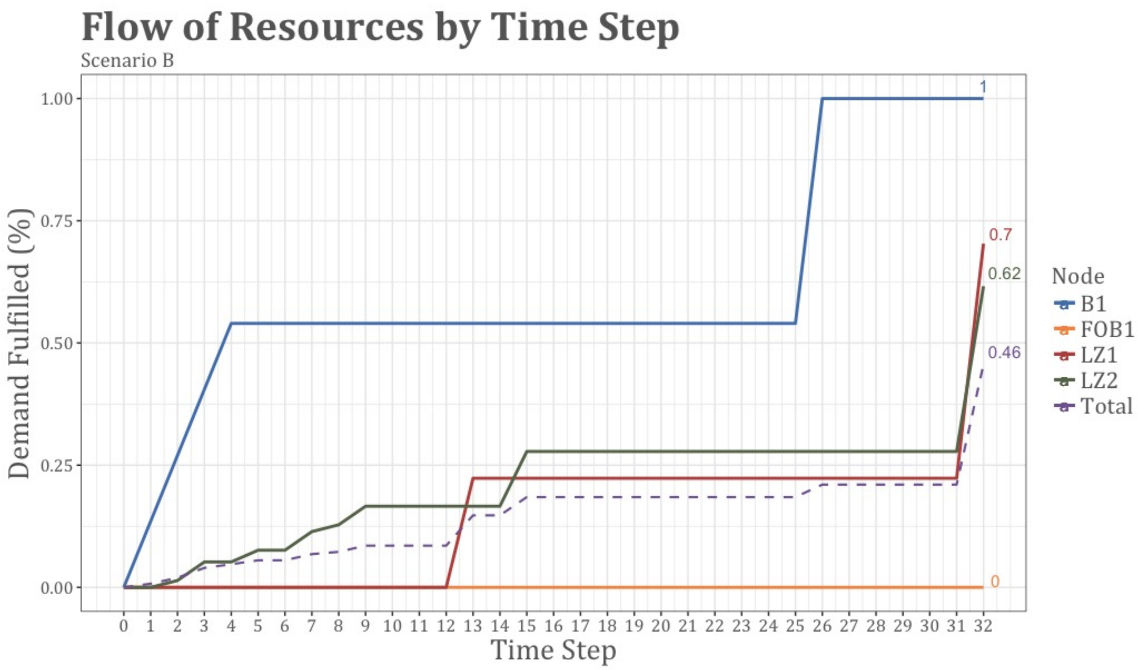


Figure 2.4. Scenario B Output - Flow of Resources by Time Step

FOB1 does not have direct access to the seabase through air or surface connectors. Operationally FOB1 should use its land connector to retrieve fuel from B1; however, this does not occur in the baseline model and FOB1 receives no fuel. Currently, the baseline Quickest Flow model can only push fuel to nodes in the network. For example, flow can only move from node A to node B if node A has a land connector to push fuel to node B.

The formulation of the push approach concept is outlined in Constraint 3.5 of Section 3.1. We run the same scenario again but relocate the MTVR to B1 in order for MACS to push fuel to FOB1. Results appear in Figure 2.5.

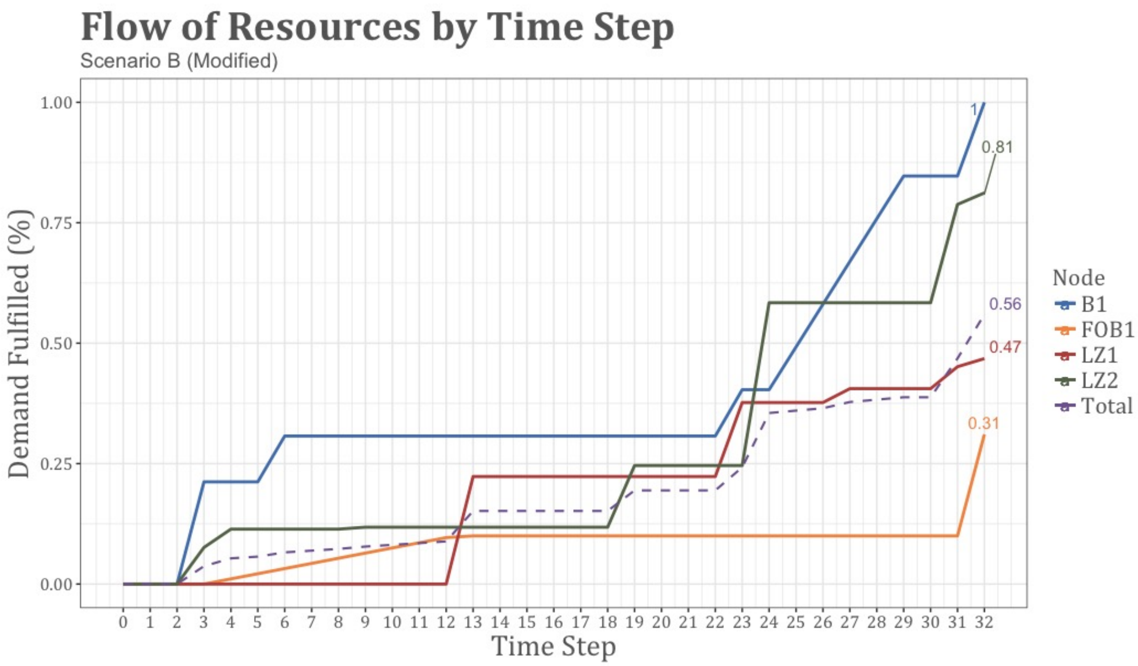


Figure 2.5. Scenario B (Modified) Output - Flow of Resources by Time Step

The relocation of the MTVR from FOB1 to B1 results in over 9,000 gallons of fuel transported to FOB1 satisfying 31% of FOB1's fuel demand. The small change in the scenario provides excellent insight into how MACS is able to move fuel across the network. In reality we will not only push fuel from node A to node B but also pull fuel from node A to node B as long as a land connector is located at either of the nodes. In Section 3.3.1, we reformulate the model to enable push and pull dynamics between nodes in order to more accurately replicate a real world scenario.

2.3.3 Scenario C

A completely new scenario is considered in Scenario C. This scenario incorporates the reallocation of bulk fuel assets to decrease the number of connector runs to the land nodes. In the original MACS model, all connectors of the same type have the same fuel capacity to transport fuel ashore. In reality the connector capacity is not a fixed amount but depends upon the total number of containers placed on the connector. For example, if each LCAC has two SIXCON Fuel Storage Modules (SIXCON)s, then each LCAC has an 1,800-gallon bulk fuel carrying capacity. If there are three LCACs in the network we have a total of six SIXCONs available for transporting fuel. The need to allocate different containers quantities across the same connector type is a consideration for the MEU staff. Instead of three LCAC's with 1,800-gallon capacities, why not two LCAC's of 2,700 gallon capacities, would this result in fewer runs and consume less fuel?

To examine this consideration further, we establish a fixed quantity of containers in Scenario C. In Part 1 we build a schedule with the containers allocated one way, then reallocate the containers across the connectors and build another schedule (Part 2). It is noted there are no bulk fuel container assets added or subtracted between each run of the scenario, the assets are simply relocated to another connector type. In Part 1 the LCAC contains two SIXCONs for moving fuel while the LCU contains three. Similarly, the CH-53 contains two TBFDSs and the MV-22 has one. In Part 2, the assets are reallocated and all five SIXCONs are placed on the LCU and all three TBFDSs are allocated to the MV-22, while the LCAC and CH-53 are unused.

Scenario C consists of two land nodes: B1 and LZ1. Inputs for both parts of Scenario C appear in Table 2.5 along with an illustration of the network shown in Figure 2.6. Note the difference in connector capacities between Part 1 and Part 2 of the scenario. In this scenario, we continue to hold the time horizon at eight hours with 32 time steps of 15 minutes each.

Scenario C Inputs (Part 1)			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	1,800
LCU	Surface	1	2,700
CH-53	Air	1	1,600
MV-22	Air	1	800
MTVR	Land (B1, LZ1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	20,000	LZ1
LZ1	Air, Land	15,000	B1

Scenario C Inputs (Part 2)			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	0
LCU	Surface	1	4,500
CH-53	Air	1	0
MV-22	Air	1	2,400
MTVR	Land (B1, LZ1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	20,000	LZ1
LZ1	Air, Land	15,000	B1

Table 2.5. Scenario C Inputs

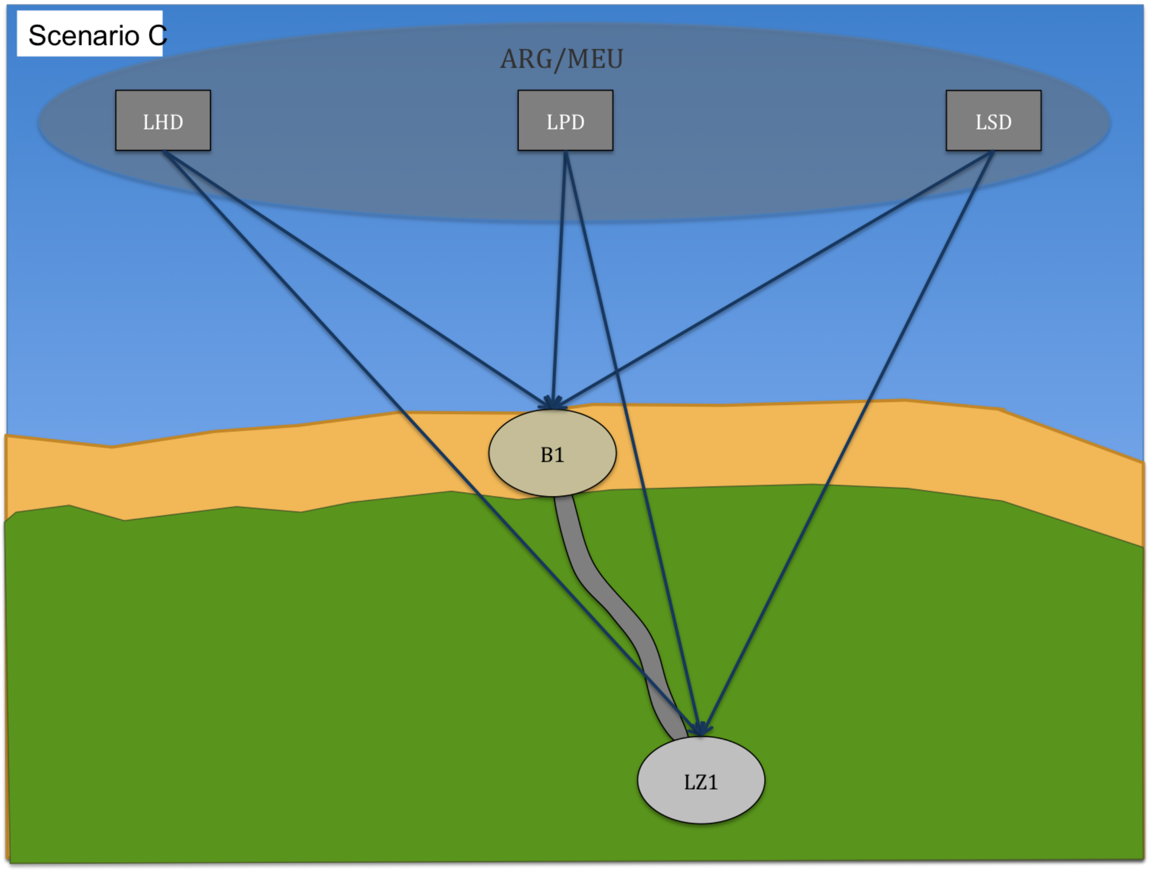


Figure 2.6. Scenario C Network Illustration

For both parts of Scenario C, the MACS tool delivers a total of 34,620 gallons of fuel to B1 and LZ1 satisfying 98.14% of the demand. However, each scenario produces a considerable difference in the total number of runs needed to achieve these quantities. Part 1 produces a schedule that equals 25 runs across all connector types, while Part 2 calculates a schedule needing only 13 runs of two connector types to achieve the same results. The capacity configuration in Part 2 reduces the total run count by almost 50% and decreases the total fuel consumption by 80%. Outputs for each part appear in Tables 2.6 and 2.7.

MOPs			
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)
98.14	34,620	25	8,334

Connector Run Count		
Connector	Land Node	Run Count (Total=25)
LCAC	B1	3
LCU	B1	3
CH-53	B1	2
	LZ1	7
MV-22	B1	5
	LZ1	5

Table 2.6. Scenario C Output (Part 1)

MOPs			
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)
98.14	34,620	13	1,634

Connector Run Count		
Connector	Land Node	Run Count (Total=13)
LCU	B1	3
MV-22	B1	3
	LZ1	7

Table 2.7. Scenario C Output (Part 2)

We conclude the MACS model calculates the most optimal solution available with the given inputs from the user; however, the user inputs might not be adequate in achieving a schedule that is most efficient. The development of an algorithm to find an efficient allocation of bulk fuel containers across the connectors is imperative in reducing the fuel consumption during an amphibious operation and producing more efficient schedules.

2.4 Conclusion

While the MACS tool effectively calculates feasible connector schedules, there is room for improvement to make the tool more robust when considering real world applications. The incorporation of many of the concerns stated above is considered in the reformulation of the model in Chapter 3. Additional testing of the tool is compared and contrasted with the results above in order to illustrate improved capabilities and robustness of the updated tool.

CHAPTER 3:
Improvements to the Quickest Flow Model

As discussed in Chapter 2, the MACS tool consists of three sub-models: Quickest Flow, Assignment Heuristic, and Scheduler. We concern ourselves only with the Quickest Flow model in this thesis and this chapter provides the mathematical approach to changes in Quickest Flow. The outputs of the Quickest Flow model produce the connector run counts needed to feed into the subsequent models of the MACS tool. For example, we see the run counts from Scenarios A and B in Chapter 2 in Tables 2.2 and 2.4. These run counts are utilized in the Assignment Heuristic and Scheduler to produce the minute-by-minute connector schedule.

In Section 3.1, we provide an overview of the Quickest Flow model from Christafore (2017). The subsequent sections feature an addition or improvement made to the MACS tool along with a discussion of the mathematical approach used to implement each change. Our changes made to the model incorporate better connector efficiency, consider a more realistic flow of resources inside the land network, and achieve more control of the fuel flow in the model. We end the chapter by discussing the different schedules created using the additions made to the model, and how the different schedule types are utilized to further benefit the end user.

3.1 Quickest Flow Model Overview

In this section, we provide a summary overview of the Quickest Flow model from Section 3.1 of Christafore (2017). For additional information please see his thesis.

The Quickest Flow model maximizes the throughput of fuel across all connectors inside a given time horizon. We use the Dynamic Network Flow (DNF) modeling approach from Hamacher and Tjandra (2001) to model the flow of resources throughout our network. Hamacher and Tjandra model the evacuation of personnel from a building utilizing a DNF model with the nodes representing rooms and a super sink node designated as the safety area (Hamacher and Tjandra 2001). Their DNF model creates copies of the static network flow problem and expands these copies over a discrete time horizon. This approach allows a user to solve a DNF problem as a static network flow problem through measuring the resources consumed by the super sink node (Hamacher and Tjandra 2001).

Using a similar approach, the Quickest Flow model seeks to maximize the movement of flow throughout the network to a super sink node over a discrete set of time periods. The Quickest Flow model contains a variety of constraints to ensure the appropriate amount of fuel is pushed ashore, each land node's fuel storage threshold is not surpassed, the flow-balance at each node is satisfied, and capacities across all nodes are not exceeded during each time period. In the MACS tool, the super sink node can be interpreted as a holding area for the total amount of fuel that is consumed at various nodes in the network. All land nodes ashore are connected to the super sink node through directed arcs, which are not limited by capacity or travel time constraints.

In the MACS model there are three primary nodes utilized to represent the network: supply nodes, land nodes, and super sink nodes. Supply nodes are indexed as $s \in S$, where S is the set of connector types: LCAC, LCU, CH-53, and MV-22. The land nodes are indexed as $l \in L$, where L is the set of all land nodes in the network. Finally, the super sink node is represented as $ss \in SS$, where SS is the set of super sink nodes. For our network we consider only one super sink node to measure the amount of fuel consumed ashore by the land nodes. We define a set of discrete time periods $t \in T$, where T is the time horizon considered for scheduling. Arcs between each node are represented as $(i, j) \in A$ with A representing all arcs in the network.

We incorporate several parameters into the Quickest Flow model to ensure the model is adequately representing the network. We denote $MaxTime$ to represent the maximum amount of time in the time horizon. For example, from Chapter 2 we implement an eight hour time horizon with 32 time steps of 15-minute increments. The amount of supply each connector can push ashore is denoted by $supply_s$ with s representing each connector type. The fuel demand at each land node is denoted by $demand_l$ and is a direct input from the user. The storage capacity of each node is designated as $nodeCap_l$ with l representing each land node in the set of all land nodes L. The travel time across each arc in the network is denoted as tau_{ij} and its value represents the number of time steps to travel from node i to node j. Finally, we denote the capacity of flow emanating in each time period from node n as u_n where n is a node in the set of all nodes N.

A key decision variable $X_{i,j,t}$ accounts for the flow of resources from node i to node j starting at time period t. More specifically, we measure the amount of flow consumed in the network by measuring the flow of resources from land node l to the super sink node ss across all time periods t. We then convert this measurement into a discrete number of runs for further processing and schedule development. The objective of the Quickest Flow program is to maximize the flow of consumed fuel across all land nodes $l \in L$ over the given time horizon T. We accomplish this objective by measuring the amount of flow that reaches the super sink node across all time periods $t \in T$. There are 11 total constraints with the baseline Quickest Flow linear program and each constraint is developed to ensure the network's attributes are not violated. We present the formulation of the Quickest Flow linear program below and give a brief description of each constraint. A more detailed explanation of the Quickest Flow model is outlined in Section 3.1 of Christafore (2017).

Indices and Sets

$t \in T$	Time periods
$s \in S$	Supply connectors
$l \in L$	Land nodes with fuel demand
$ss \in SuperSink$	Single $ss \in SuperSink$
$(i, j, ss) \in N$	All nodes in the network: $n \in N$
$(i, j) \in A$	Directed arcs from node i to node j for all nodes $n \in N$

Parameters [units]

$MaxTime$	Max time considered for operations [minutes]
$supply_s$	Amount of supply from each connector s [gallons]
$demand_l$	Fuel demand at land node l [gallons]
$nodeCap_l$	Storage capacity at land node l [gallons]
$tau_{i,j}$	Travel time on arc (i, j) [minutes]
u_n	Flow capacity leaving node n for each time period t [gallons]

Decision Variables [units]

$X_{i,j,t}$	Flow of fuel across arc (i, j) at time t [gallons]
$Y_{n,t}$	Amount of fuel stored at node n at time period t [gallons]

Formulation

$$\max_{X,Y} \quad \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SS \\ (l,ss) \in A}} X_{l,ss,t} \tag{3.1}$$

$$\text{s.t.} \quad Y_{s,0} = supply_s \qquad \forall s \in S \tag{3.2}$$

$$Y_{l,0} = 0 \qquad \forall l \in SuperSink \tag{3.3}$$

$$Y_{l,t} \leq nodeCap_l \qquad \forall l \in L, \forall t \in T \tag{3.4}$$

$$\sum_{\substack{l \in L \\ (n,l) \in A}} X_{n,l,t} \leq u_n \qquad \forall n \in S \cup L, \forall t \in T \tag{3.5}$$

$$\sum_{t \in T} \sum_{\substack{ss \in SuperSink \\ (l,ss) \in A}} X_{n,l,t} \leq demand_l \qquad \forall l \in L \tag{3.6}$$

$$Y_{n,t+1} = Y_{n,t} + \sum_{\substack{i \in N: \\ (i,n) \in A}} X_{i,n,t-tau_{i,n}} - \sum_{\substack{j \in N: \\ (n,j) \in A}} X_{n,j,t} \qquad \forall n \in N, \forall t \in T \setminus \max(T) \tag{3.7}$$

$$\sum_{\substack{j \in N \\ (l,j) \in A}} X_{l,j,\max(T)} \leq Y_{l,\max(T)} + \sum_{\substack{i \in N \\ (i,l) \in A}} X_{i,l,\max(T)-tau_{i,l}} \qquad \forall l \in L \tag{3.8}$$

$$\sum_{\substack{l \in L \\ (s,l) \in A}} X_{s,l,\max(T)} \leq Y_{s,\max(T)} \qquad \forall s \in S \tag{3.9}$$

$$\sum_{t \in T} \sum_{s \in S} \sum_{\substack{j \in N \\ (s,j) \in A}} X_{s,j,t} \leq \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SuperSink \\ (l,ss) \in A}} X_{l,ss,t} \tag{3.10}$$

$$X_{i,j,t} \geq 0 \qquad \forall i \in N, \forall j \in N, \forall t \in T \tag{3.11}$$

$$Y_{n,t} \geq 0 \qquad \forall n \in N, \forall t \in T \tag{3.12}$$

The following are descriptions of the constraints associated with this model formulation to ensure arc and node capacities are not breached, the balance of flow is properly maintained throughout the time horizon, and the fuel flow is appropriately tracked for the duration of the model.

1. Constraint (3.2) requires that the supply is initiated at supply node s at time 0.

2. Constraint (3.3) ensures each land node's beginning supply is zero at time 0.
3. Constraint (3.4) ensures land node's l storage capacity is not breached during each time period t.
4. Constraint (3.5) requires that at each time period the outgoing total flow from each land node l does not exceed the land node's capacity to push fuel from the node.
5. Constraint (3.6) guarantees each land node's demand is not exceeded during the time horizon by calculating the total the flow emanating from land node l to the super sink node ss across the entire time horizon T.
6. Constraint (3.7) calculates the storage at node n by calculating the difference between the flow out of the node and the flow into the node for each time period t and adding this value to the prior storage amount at the node $t-1$.
7. Constraint (3.8) requires the flow across land node l during the last time period must be less than the current storage at that node plus the flow into the node.
8. Constraint (3.9) ensures the flow balance at supply node s is not exceeded during the last time period; therefore, flow out of the node must be less than the current storage at that node.
9. Constraint (3.10) requires that the flow of fuel pushed from the seabase is equal to the amount of fuel received by the land nodes.
10. Constraints (3.11 and 3.12) ensure each of the decision variables X and Y are non-negative values.

Using the output from the model, we find the flow of fuel through the network across the entirety of the time horizon. More importantly, we know the flow of fuel across all the arcs in the network to include the flow across each connector type s. Utilizing this information along with the fuel carrying capacities of the connectors, we can calculate the total number of runs from each connector type s to each land node l over the course of the time period T. For example, four LCAC runs to B1 and three LCAC runs to B2. The outputs of the Quickest Flow model appear in the scenario output tables in Chapter 2.

3.2 Quickest Flow Objective Function

In this section, we examine the two main updates added to the Quickest Flow Model's objective function: the introduction of an exponential penalty term and an implementation of node priorities.

3.2.1 Exponential Penalty Term

While the original MACS tool focuses on pushing as much fuel as possible ashore within a specified time frame, there are no benefits to pushing fuel early in the time window. This particular concern is highlighted in Section 2.3.1 in which almost 20% of the fuel demand is satisfied in the final 30 minutes of the time horizon. Taking this issue into consideration with the current model, we reformulate the Quickest Flow's objective function in order to ensure the flow of fuel ashore is achieved in the quickest time feasible.

The current objective function, Equation (3.1), from the Quickest Flow model appears below:

$$\max_{X,Y} \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SupSink \\ (l,ss) \in A}} X_{l,ss,t}$$

The objective function seeks to maximize fuel consumed ashore over the time horizon of interest. Through maximizing the throughput of flow across the arcs $(l, ss) \in A$ over all time periods in T, we obtain an accurate measurement of the amount of fuel consumed by the land nodes in the network.

As the objective function is written, there is not a forcing feature to push resources earlier in the model. The model only concerns itself with satisfying the fuel demand ashore before the time horizon expires. For example, Figure 3.1 features a simple three node network, one supply node S, one demand node D, and one super sink node SS. A standard network flow model consists of arc capacities to ensure the flow of resources across each arc are not violated. In our model, arc capacities represent the amount of flow that can travel on the arc (i, j) per time period. The network below contains two arcs (S, D) and (D, SS) with arc capacities of 1 gallon of flow per time step for arc (S, D) and an infinite arc capacity on arc (D, SS). The infinite arc capacity on arc (D, SS) represents the demand being consumed immediately in the network. Finally, the supply node S contains a supply value of 5 gallons while the demand node D contains a demand value of 5 gallons.

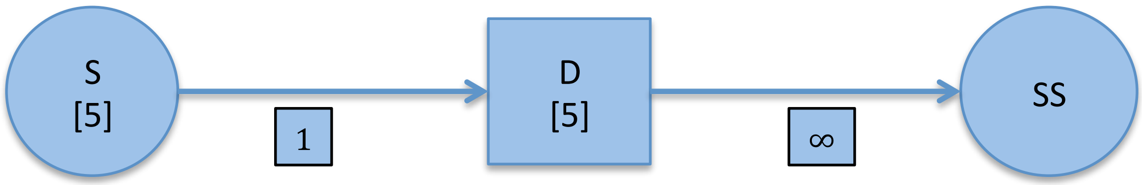


Figure 3.1. Example Network

Given a time horizon of 10 time steps, the current Quickest Flow model attains a maximum value of 5: the demand at node D is fully satisfied. With no forcing function to push fuel earlier in the 10 period time horizon, the maximum value will be attained at time step 10. It can be seen that this value could be achieved at time step 5; however, the current implementation may use the entire time horizon because there is no benefit to pushing fuel earlier. Table 3.1 represents the values of $X_{S,D,t}$, the amount of flow at each time period, as well as other possible solutions under the original objective function. Any number of these solutions can be returned as "the" optimal solution by the solver, but in practice the model most often returns Solution 1. We see Solution 4 is the ideal solution as it pushes all fuel through the network in the first five time steps, we aim to achieve this solution.

$X_{S,D,t}$ Values by Time Step				
Time Step	Solution 1	Solution 2	Solution 3	Solution 4
1	0	0	0	1
2	0	0	0	1
3	0	0	0	1
4	0	0	1	1
5	0	1	1	1
6	1	1	1	0
7	1	1	1	0
8	1	1	1	0
9	1	1	0	0
10	1	0	0	0
Obj Fun	**5**	**5**	**5**	**5**

Table 3.1. Example Output

The current formulation impedes our ability to satisfy the fuel demand ashore as quickly as possible; therefore, we introduce an exponential penalty term to increasingly penalize the objective function as time increases. The new formulation of the objection function appears in Equation (3.13).

$$\max_{X} \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SupSink \\ (l,ss) \in A}} \alpha^t X_{l,ss,t} \quad ; \quad \alpha = 0.999 \tag{3.13}$$

The implementation of the exponential penalty term progressively penalizes the objective function value for every time step the model uses to push fuel ashore. We implement the new objective function featured in Equation (3.13) and now the model will only achieve its maximum value by time step 5 if the flow of fuel is initiated at time step 1. Comparative results of Solution 1 from Table 3.1 and the solution from the updated objective function value are compared in Table 3.2.

$\alpha^t X_{S,D,t}$ Values by Time Step (Updated)				
Time Step	Solution 1	Solution 2	Solution 3	Solution 4
1	0	0	0	0.999
2	0	0	0	0.998
3	0	0	0	0.997
4	0	0	.996	0.996
5	0	.995	.995	0.995
6	.994	.994	.994	0
7	.993	.993	.993	0
8	.992	.992	.992	0
9	.991	.991	0	0
10	.990	0	0	0
Obj Fun	**4.960**	**4.965**	**4.970**	**4.985**

Table 3.2. Example Output (Updated)

The results clearly show the maximum objective function value with the penalty term can only be achieved if flow is pushed immediately in the time horizon. We now obtain a

Quickest Flow model that seeks to flow resources to land nodes ashore in the quickest amount of time possible.

3.2.2 Priority Node Assignment

We introduce priority node assignment to place fuel where it is needed most in the network. For example, if B1's fuel demand is more critical than any other land node in the network, the model should be biased so that it focuses on pushing as much fuel as possible to the priority nodes first before considering other nodes in the network.

We start by assigning a generic weight w_l to each land node. We then apply the new input parameter into the objective function. The updated objective function appears in Equation (3.14).

$$\max_{X} \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SupSink \\ (l,ss) \in A}} w_l \alpha^t X_{l,ss,t} \quad ; \quad \alpha = 0.999 \tag{3.14}$$

For our analysis, we define $w_l = \frac{1}{P_l}$, where P_l is an integer value greater than one but not greater than the number of nodes in the network. For example, if the network consists of two beaches and three landing zones, the priority values for each land node are integers no less than one and no greater than five. The objective function value will only achieve its maximum value if it pushes as much fuel as possible to the nodes with the highest priority first. In practice the user may specify any weighting scheme for w_l, but our delineation of the weight parameter seems most reasonable. We provide the updated objective function with our weighting scheme in Equation (3.15):

$$\max_{X} \sum_{t \in T} \sum_{l \in L} \sum_{\substack{ss \in SupSink \\ (l,ss) \in A}} \frac{1}{P_l} \alpha^t X_{l,ss,t} \quad ; \quad \alpha = 0.999 \tag{3.15}$$

It is noted that nodes can have the same priority value because the model weights the priority values for those nodes the same in the objective function. Additionally, if node priorities are not needed, the priority values for all land nodes are set to one and the objective function value remains unchanged.

3.3 Constraint Improvements to Quickest Flow

In this section, we reformulate and add specific constraints inside the Quickest Flow model. These improvements feature better control of the flow of fuel in the land network, the addition of a constraint to control equity across the land nodes, and a constraint to limit the fuel consumption of the connectors.

3.3.1 Push/Pull Constraint

In Section 2.3.2, we identified the inability to "pull" flow from a land node in the network. From Scenario B, FOB1 contains no air spots or surface spots; therefore, the MEU seabase cannot deliver fuel to FOB1 directly with surface or air connectors. In Scenario B, there exists an arc between nodes B1 and FOB1. FOB1 only contains one land connector, an MTVR, which in practice would be utilized to travel along the arc $(FOB1, B1)$ to retrieve fuel from B1 and transport it back to FOB1. Unfortunately, the flow capacity constraint in the baseline Quickest Flow model is only formulated to push fuel along the arcs in the network. We see the effects of this constraint in Scenario B, as the model cannot transport fuel to FOB1 via B1 because the MTVR is not located at B1.

Recall the original flow capacity from Equation (3.5), as displayed below. The constraint measures the flow of fuel being pushed by the land connectors out of node n along the arc (n, l) at time period t. This constraint also ensures the flow does not exceed the flow capacity u_n which is based on the number of land connectors located at node n.

$$\sum_{\substack{l \in L \\ (n,l) \in A}} X_{n,l,t} \leq u_n \quad ; \quad \forall n \in S \cup L, \forall t \in T$$

In Scenario B, the flow capacity of B1 (u_{B1}) is zero because B1 does not contain a land connector to push fuel to any other nodes in the network; therefore, B1 is limited in its ability to push flow along arc $(B1, FOB1)$. We reformulate the flow capacity constraint above to enable the flow of fuel between nodes so long as a land connector exists at one of the nodes in the arc.

Constraint (3.5) from the Quickest Flow model applies to the origination of flow for both supply and land nodes. We separate this constraint by removing the land node indexing and apply the constraint only to the supply nodes. The reformulation ensures the upper bound of flow emanating from supply node s to land node l is not exceeded during any time period t. The updated constraint for the supply nodes appears in Equation (3.16).

$$\sum_{\substack{n \in N \\ (s,n) \in A}} X_{s,l,t} \leq u_s \quad ; \quad \forall s \in S, \forall t \in T \tag{3.16}$$

We now handle the flow for the land nodes with separate constraints. We introduce two new decision variables, $Xpush_{i,j,t}$ and $Xpull_{j,i,t}$, to measure the amount of fuel being pushed along arc (i, j) at time period t and measure the amount of fuel being pulled across arc (i, j) at time period t, respectively. The assets utilized to push flow across arc (i, j) are located at node i, and the assets utilized to pull flow across arc (i, j) are located at node j. Next, we introduce two new constraints to manage the flow of resources being push or pulled in the network.

The first constraint appears in Equation (3.17). The purpose of this constraint is to ensure the amount of flow traveling along arc (i, j) is equal to the total amount of flow being pushed or pulled along arc (i, j).

$$Xpush_{i,j,t} + Xpull_{j,i,t} = X_{i,j,t} \quad ; \quad \forall arc \in A, \forall t \in T \tag{3.17}$$

The second constraint, Equation (3.18), restricts the upper bound of flow emanating from the nodes in the network. We ensure the total amount of flow being pushed out of node n and being pulled to node n using land assets based at n is bounded.

$$\sum_{\substack{n \in N \\ (n,j) \in A \\ j \notin SS}} Xpush_{n,j,t} + \sum_{\substack{n \in N \\ (n,j) \in A \\ j \notin SS}} Xpull_{n,j,t} \leq u_n \quad ; \quad \forall n \in N, \forall t \in T \tag{3.18}$$

These changes enable us to control the flow of fuel in the network to mirror real world operations. We are now able to push and pull flow between land nodes in the network so long as a land connector exists at one of the nodes in the arc being considered.

3.3.2 Equity Constraint

The original Quickest Flow model does not have the capability to ensure all land nodes in the network receive at least some level of fuel. For example, the model might push more resources to one node leaving other nodes to receive a small amount of resources before the time window expires. One way to mitigate this concern is with the priority node assignment discussed in Section 3.2.2; however, this approach targets only a small set of specific nodes. We implement a more general approach by introducing an equity constraint to ensure at least some minimum measure of fuel is delivered to all land nodes in the network.

The equity constraint is outlined in Equation (3.19). The constraint ensures the total amount of flow consumed at land node l is greater than or equal to a percentage p of the demand value at that node. The percentage p is an inputted value that can be tailored by the user as necessary.

$$\sum_{t \in T} \sum_{\substack{n \in N \\ (n,ss) \in A}} X_{n,ss,t} \geq p * demand_l \quad ; \quad \forall l \in L \tag{3.19}$$

With the above constraint, we ensure the model is satisfying a minimum amount of demand at each node. Furthermore, if the constraint is not needed, a value of zero is inputted for p and the constraint is no longer impacting the flow of fuel in the network. It is noted that the implementation of this constraint may lead to an infeasible solution. In this case, the user must reduce the value of p in order to produce a feasible schedule.

3.3.3 Fuel Consumption Constraint

The original model does not control or limit the amount of fuel consumed by each connector type. We construct a fuel constraint to limit the amount of fuel consumed during the time horizon.

In order for us to properly build the constraint, several new parameters are presented. We introduce the parameter $fuelLoad_s$, to denote the bulk fuel carrying capacity of connector type s. The parameter $fuelConsume_s$ represents the fuel consumption rate of connector s per time period t. The parameter $fuelTank_s$ is the onboard fuel tank capacity of connector s. The inputted value Q_s is an integer value representing the total number of tanks of fuel connector s can consume during the time horizon.

Before presenting the fuel consumption constraint, we walk through an example to illustrate the logic. We consider a small network in which an LCAC is required to make multiple runs to a single land node (B1). Our goal is to restrict the amount of fuel the connector will consume through controlling the number of runs the connector can make to B1. The input parameters for this example appear in Table 3.3.

Parameter	**Value**	**Description**
$tau_{LCAC,B1}$	2 time steps	Time required to travel to $B1$
$fuelConsume_{LCAC}$	3 gal/time step	Fuel consumption rate
$fuelTank_{LCAC}$	24 gal	Fuel tank capacity
$fuelLoad_{LCAC}$	20 gal	Bulk fuel carrying capacity
Q_{LCAC}	1	Quantity of fuel tanks permitted to consume
T	12 time steps	Time horizon
$X_{LCAC,B1,t}$	5 gal	Flow of fuel across arc $(LCAC, B1)$ per time period t

Table 3.3. Fuel Example Parameters

Our first step is to calculate the number of time steps the LCAC requires to conduct a complete round trip to B1. Multiplying the tau parameter by two we obtain the needed value and the calculation is represented in the equation below.

$$tau_{LCAC,B1} * 2 = 4 \; time \; steps$$

The next goal is to calculate the amount of fuel the LCAC consumes while conducting a complete round trip to B1. We multiply the above equation by the $fuelConsume_{LCAC}$ parameter to attain the total amount of fuel consumed for a single round trip.

$$tau_{LCAC,B1} * 2 * fuelConsume_{LCAC} = 12 \; gallons/round \; trip$$

Now that we know the total amount of fuel the LCAC will consume for one complete round trip to B1, the next step is to compute the number of runs the LCAC will make to B1. We accomplish the task by summing the decision variable $X_{LCAC,B1,t}$ across all times steps in the time horizon resulting in the total amount of fuel the LCAC is responsible for delivering to B1. To find the number of runs, we divide the total amount of fuel by the $fuelLoad_{LCAC}$ parameter, resulting in the number of runs the LCAC will make to B1. The equation for this calculation appears in Equation (3.20).

$$\frac{\sum_{t \in T} X_{LCAC,B1,t}}{fuelLoad_{LCAC}} = runCount_{LCAC,B1} \tag{3.20}$$

Our last step is to limit the number of runs by restricting the total amount of fuel the LCAC can consume. We restrict the number of runs by ensuring the total amount of fuel consumed by the LCAC does not exceed a specified amount. We understand Equation (3.20) could result in a non-integer solution; however, our approach allows us to establish an approximate upper-bound on the number of runs needed to stay below the fuel consumption limit. In our example, the total amount of fuel the LCAC can consume is limited to one tank of fuel as depicted below.

$$fuelTank_{LCAC} * Q_{LCAC} = 24 \ gallons$$

Now, we compile all of the above equations into one. In the example above, we have only one connector delivering fuel to one land node; however, in practice we generalize this approach to enable the fuel constraint to be implemented across all connectors and all land nodes. We provide the new fuel constraint in Equation (3.21).

$$\sum_{\substack{l \in L \\ (s,l) \in A}} 2 * tau_{s,l} * fuelConsumed_s * \frac{\sum_{t \in T} X_{s,l,t}}{fuelLoad_s} \leq fuelTank_s * Q_s \quad ; \quad \forall s \in S \tag{3.21}$$

Utilizing the LCAC example above, we compare the implementation of the fuel constraint across two different scenarios: fuel unrestricted and fuel restricted. In the unrestricted scenario, the LCAC delivers 60 gallons of fuel to B1 across three round trips, and consumes 36 gallons of fuel during the time horizon. In the second scenario, we ensure the LCAC does not consume more than one tank of fuel (24 gallons) throughout the scenario. Scenario 2 results in 40 gallons of fuel being delivered to B1 across two round trips to the land node. Additionally, Scenario 2 only consumes 24 gallons of fuel. We see in Table 3.4 the fuel constraint is properly restricting the number of runs the LCAC is making to B1 by ensuring the LCAC does not exceed the specified amount of fuel inputted by the user.

	Scenario 1: Fuel Unrestricted		**Scenario 2: Fuel Restricted**	
Time Step	$\mathbf{X_{LCAC,B1,t}}$	**Fuel Consumed (gal)**	$\mathbf{X_{LCAC,B1,t}}$	**Fuel Consumed (gal)**
1	5	3	5	3
2	5	3	5	3
3	5	3	5	3
4	5	3	5	3
5	5	3	5	3
6	5	3	5	3
7	5	3	5	3
8	5	3	5	3
9	5	3	0	0
10	5	3	0	0
11	5	3	0	0
12	5	3	0	0
Total	**60**	**36**	**40**	**24**
	Total Number of Runs: 3		**Total Number of Runs: 2**	

Table 3.4. Fuel Example Output

The fuel constraint enables us to track the fuel consumption of connector s across arcs $(s,l) \in A$ over all time periods t. The constraint incorporates the fuel consumption rate of connector s, the round trip travel time to land node l, and the bulk fuel carrying capacity of the connector $fuelLoad_s$ in order to calculate the total amount of fuel utilized across all runs of the connector. We constrain the fuel consumption value by ensuring it does not exceed a user specified amount of fuel. The end result is the ability to restrict the amount of fuel a connector can utilize by limiting the amount of tanks of fuel the connector can consume.

3.4 Rounding Techniques for Connector Run Counts

The Quickest Flow model outputs the total amount of fuel each connector type will deliver to the land nodes ashore. Utilizing Equation (3.20), we determine the number of runs each connector will complete by dividing the output from Quickest Flow by the capacity of the connector as outlined in Equation (3.22).

$$\frac{\sum_{t \in T} X_{s,l,t}}{fuelLoad_s} = runCount_{s,l} \quad ; \quad \forall (s,l) \in A \tag{3.22}$$

Often Equation (3.22) results in a decimal solution. For example, over the course of a time horizon an LCAC is responsible for transporting 1000 gallons of fuel to B1 and the LCAC's bulk fuel carrying capacity $fuelLoad_{LCAC}$ is 900 gallons. The resulting run value is 1.11. In these cases, the original method used in the MACS tool is to round the value up to the closest integer. In our example, the LCAC would be required to make two trips in order to completely meet the demand at B1.

The current method to find the number of runs to each land node is simple but results in delivering more fuel than necessary to shore; potentially a non-trivial amount. We define "excess" fuel delivered to the land nodes as fuel that cannot be consumed by the node. The MACS model assumes a connector departs the seabase carrying its total capacity of bulk fuel; therefore, the schedule implemented in the LCAC example above results in the connector transporting an extra 800 gallons of fuel to B1. We identify the 800 gallons of extra fuel as excess. Depending upon the scenario, delivering excess fuel may pose no issue, as the fuel will be consumed in the near future. However, in other cases it may be preferable to deliver slightly less than possible fuel amounts if it results in more connector efficiency. We implement three different approaches for rounding the calculation from Equation (3.22). Each approach's description along with pros and cons are depicted in Table 3.5.

Rounding Methods			
Method	***Description***	***Pros***	***Cons***
1	Round up to the nearest integer value	Ensures the maximum amount of fuel feasible is delivered to each land node	Extra runs are unavoidable and, in some cases, excess fuel delivered to a node cannot be consumed by that node
2	Round to the nearest integer value	Combination of Method 1 and Method 3, a variety of nodes will receive excess and deficient fuel deliveries	Difficult to know which nodes will receive excess fuel and which nodes will not
3	Round down to the nearest integer value	Extra runs are eliminated and excess fuel deliveries are no concern	Cannot achieve the maximum amount of fuel to be delivered

Table 3.5. Rounding Methods

Each approach contains its own set of pros and cons; however, we give the user greater flexibility to control the amount of excess fuel delivered to shore by the MACS tool. In Chapter 4, we perform an analysis of each approach to compare the average amount of excess fuel delivered along with the average amount of demand met at each node. The analysis highlights some of the concerns mentioned in Table 3.5 and provides a thorough understanding of the implications of each rounding approach.

3.5 Building Multiple Schedules

With the above additions implemented in the Quickest Flow model, the updated MACS2 tool is now capable of producing various unique schedules:

1. **Base Schedule**: Baseline schedule incorporates the push/pull constraint with the addition of the exponential penalty term. No fuel, equity, or rounding restrictions are considered in this schedule.

2. **Priority Node Assignment**: Nodes assigned as a high priority take precedence over other nodes in the network.

3. **Fuel Restricted**: The amount of fuel utilized by the connectors is limited to the number of tanks of fuel the connectors will consume.

4. **Equity**: The schedule is constructed based on the maximum equity achieved across all land nodes.

5. **Rounding Method**: Each connector schedule is built with the intention of mitigating excess fuel pushed ashore due to run count rounding.

Each of the five schedules offer the MEU staff an opportunity to take advantage of the unique features the MACS2 model now incorporates. For example, the Fuel Restricted schedule can be chosen when resources are depleted and the MEU is ensuring they are using a specified amount of fuel during operations. The Equity schedule may be chosen when the amount of fuel delivered ashore must be distributed evenly across the land nodes. Furthermore, each of these schedules can be layered to produce a unique schedule tailored specifically for the user. For example, the user can build a schedule which features priority nodes, fuel restrictions, and a specified rounding technique all in one package.

In Chapter 4, we evaluate the effectiveness of the MACS2 tool and compare the outputs against the original tool in Christafore (2017). We also demonstrate the capability of layering the schedules to provide a schedule specific to the user's desires.

CHAPTER 4:
MACS2 Evaluation

In this chapter, we provide a comparative analysis of the outputs from our MACS2 tool against the results attained with the original MACS model. We end the chapter with a demonstration of how each of the five different modifications discussed in Section 3.5 can be layered to create a unique schedule specific to a user's needs.

4.1 MACS vs. MACS2

Our baseline schedule incorporates the addition of the exponential penalty term and the reformulation of the push/pull constraint. We integrate the other additions from Chapter 3 where appropriate. In the following sections, we apply our reformulated MACS2 model against Scenarios A and B created in Chapter 2. Using the same MOPs as before, we output the results and compare the new outputs to the original output obtained from the original MACS tool.

4.1.1 Scenario A

As discussed in Chapter 2, Scenario A features four land nodes: B1, LZ1, LZ2, and FOB1. We restate Scenario A's inputs from Table 2.1 in Table 4.1 and we utilize the same time horizon of eight hours to produce 32 time steps of 15-minute increments.

Scenario A Inputs			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	4,500
LCU	Surface	1	1,800
CH-53	Air	1	2,400
MV-22	Air	4	800
MTVR	Land (FOB1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	5,000	FOB1
LZ1	Air	25,000	None
LZ2	Air	30,000	None
FOB1	Air, Land	30,000	B1

Table 4.1. Scenario A Inputs

The MACS2 model calculates a feasible schedule for all connector types within one minute and we see no significant change in computation time from the original model. Our updated model satisfies over 69% of the demand ashore for a total movement of 62,193 gallons. A comparison of the MOPs from the original model and MACS2 model appear in the Table 4.2. We see a 14,000-gallon increase in the amount of fuel pushed ashore and we conclude the increase in fuel delivery is due to the reformulated push/pull constraints that better model how fuel is transported between the land nodes. More notably, the substantial increase in fuel delivered results in a minimal rise in fuel consumed. Our model now utilizes the MTVR located at FOB1 to retrieve fuel from B1, an unattainable task using the original tool. We also see an increase in the number of runs from 46 to 49. Comparing the number of runs for each connector type in Table 4.3 we see the LCAC performs an extra run to B1 to account for the MTVR pulling fuel from B1 and delivering it to FOB1.

Scenario A: MOP Comparison		
MOPs	Original MACS	MACS2
Demand Satisfied (%)	54.39	69.10
Bulk Fuel Delivered (gal)	48,954	62,193
Number of Runs	46	49
Fuel Consumed (gal)	9,578	9,607

Table 4.2. Scenario A: MOP Comparison

Scenario A: Run Count Comparison			
Connector	Land Node	Original MACS	MACS2
LCAC	B1	2	3
LCU	B1	1	1
CH-53	LZ1	4	1
	LZ2	2	0
	FOB1	2	9
MV-22	LZ1	20	9
	LZ2	8	26
	FOB1	7	0

Table 4.3. Scenario A: Run Count Comparison

Figure 4.1 illustrates the flow of resources to each of the land nodes across the entirety of the time horizon. From Figure 2.2 in Chapter 2, the original MACS model pushed 20% of the resources in the final two time steps. Our reformulated model begins pushing fuel immediately, and we see a consistent flow of resources across the time horizon throughout the entire network.

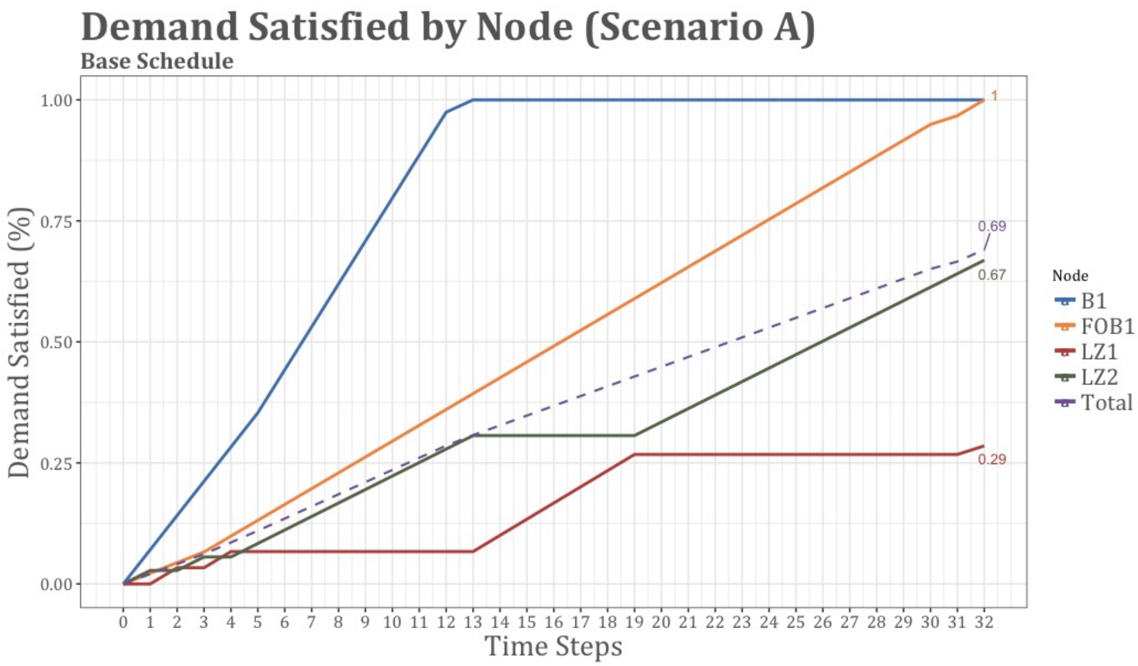


Figure 4.1. Scenario A Output - Reformulated Model

A concern discussed in Chapter 2 is the large disparity between the quantities of fuel delivered across the nodes. For example, the output from Figure 2.2 shows only 29% of the demand is met at FOB1, Figure 4.1 has a similar disparity with LZ1. With the reformulated MACS2 tool, we can now easily manage this issue in two different ways: 1) prioritize the nodes or 2) run the model utilizing the equity constraint to find the best minimum value of fuel received by the land nodes. We apply each approach below and compare the results.

Our first approach prioritizes land nodes by ranking the nodes in increasing order of demand satisfied based on Figure 4.1 and rerunning the model. From Section 3.2.2, we utilize inverse ranks to assign node priorities. This approach results in $P_{LZ1} = 1$, $P_{LZ2} = 2$, $P_{FOB1} = 3$, and $P_{B1} = 4$. Output of the flow of resources utilizing priority node assignment appears in Figure 4.2. We see assigning priority nodes reduces the total amount of demand fulfilled ashore from 69% to 61%. Furthermore, we see LZ1 is now receiving 100% of its demand; however, FOB1 is only receiving 33% of its needed fuel. To mitigate this issue, we next utilize the second approach and apply the equity constraint in addition to the priority node assignment.

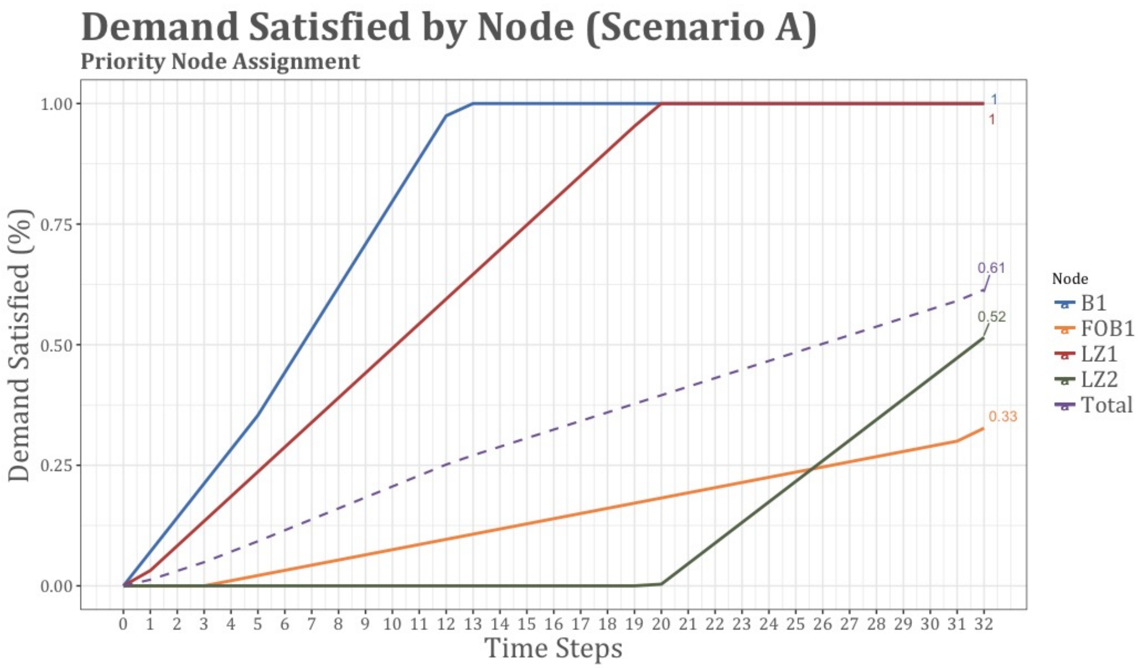


Figure 4.2. Scenario A Output - Reformulated Model (Approach 1: Priority Node Assignment)

In the second approach, we begin by setting the minimum value of fuel received by the land nodes at 0%, we then increase this value until we reach an infeasible solution. This approach results in each node receiving a minimum of 59% of their fuel requirements and the model ensures we meet the best minimum amount of fuel satisfied across the land nodes. Outputs from this approach appear in Table 4.4 and Figure 4.3, and we conclude this schedule is the best option to ensure we meet as much demand as possible across all land nodes in the network.

MOPs			
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)
61.40	55,260	46	9,882

Connector Run Count		
Connector	Land Node	Run Count (Total=46)
LCAC	B1	4
LCU	B1	1
CH-53	LZ1	3
	LZ2	3
	FOB1	2
MV-22	LZ1	12
	LZ2	15
	FOB1	6

Table 4.4. Scenario A Output (Approach 2: Priority Node Assignment with Equity Constraint)

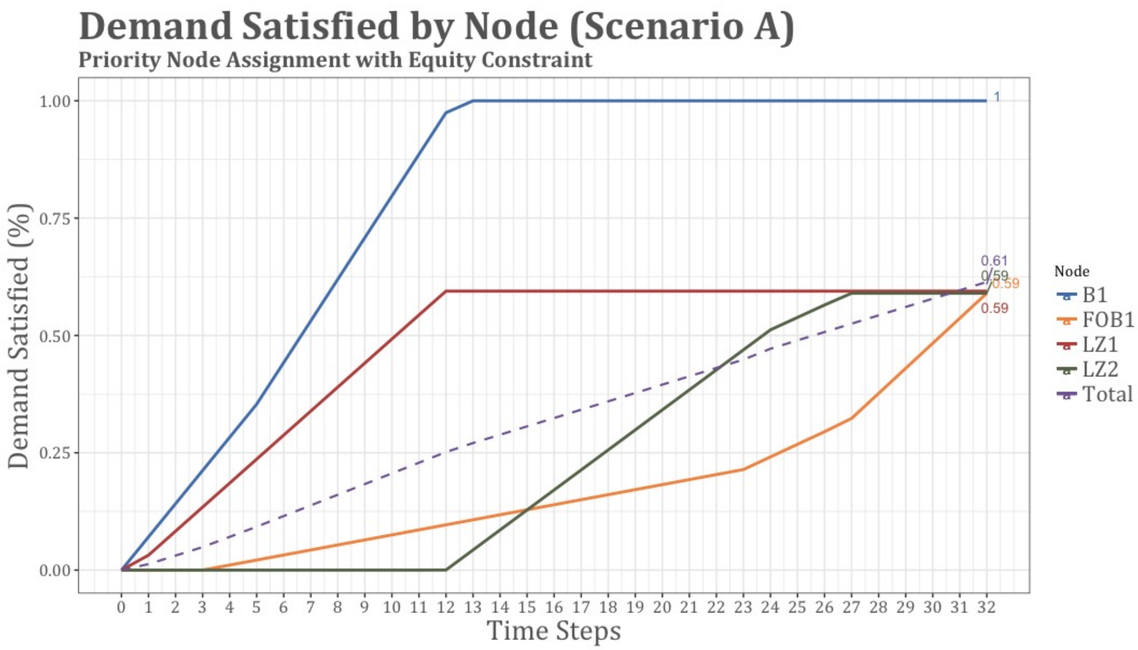


Figure 4.3. Scenario A Output - Reformulated Model (Approach 2: Priority Node Assignment with Equity Constraint)

The results from applying the MACS2 tool to Scenario A have demonstrated a substantial improvement from the original MACS model. We see an increase of over 14,000 gallons of fuel delivered while only adding three additional runs to the schedule. In addition, we see the opportunity to answer some of the concerns raised with the model from Chapter 2. Specifically, the ability to ensure the highest minimum amount of fuel is delivered across all land nodes in the network; however, through this added capability, we decrease the amount of demand satisfied from 69% to 61%.

4.1.2 Scenario B

In Scenario A, we demonstrate the ability for the MACS2 tool to utilize land connectors to retrieve fuel from another node given a land connection exists with that node. Scenario B enables us to specifically highlight this concern to confirm the model is functioning appropriately.

In Chapter 2, the purpose of Scenario B is to demonstrate the inability for the original MACS tool to "pull" resources from another node using a land connector so long as a land connection existed between the nodes. Scenario B contains the same inputs from Scenario A; however, the air connector spots located at FOB1 are eliminated. The only method of delivering fuel to FOB1 is through the use of the land connector located at FOB1. We restate the inputs for Scenario B from Table 2.3 in Table 4.5 and we utilize an eight-hour time horizon consisting of 32 time steps.

Scenario B Inputs			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	4,500
LCU	Surface	1	1,800
CH-53	Air	1	2,400
MV-22	Air	4	800
MTVR	Land (FOB1)	1	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	5,000	FOB1
LZ1	Air	25,000	None
LZ2	Air	30,000	None
FOB1	Land	30,000	B1

Table 4.5. Scenario B Inputs

The comparative output from the original MACS model and the MACS2 model appear in Table 4.6. We also provide an output of the flow of resources from the updated model in Figure 4.4.

Scenario B: MOP Comparison		
MOPs	Original MACS	MACS2
Demand Satisfied (%)	45.62	55.98
Bulk Fuel Delivered (gal)	41,060	50,381
Number of Runs	39	41
Fuel Consumed (gal)	8,294	9,265

Table 4.6. Scenario B: MOP Comparison

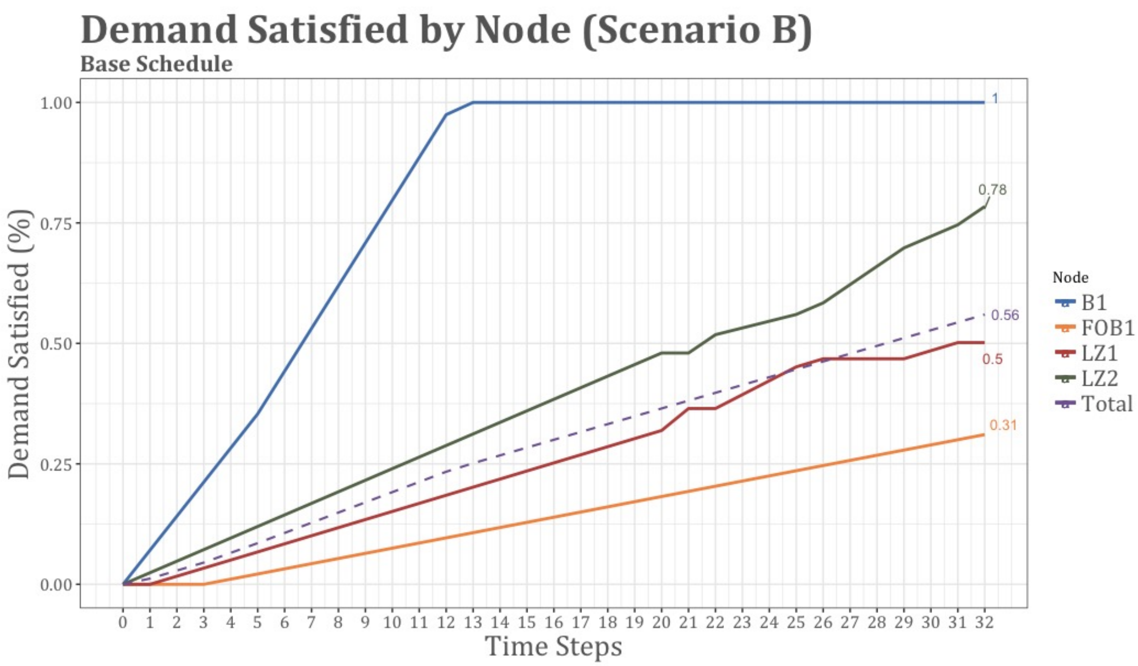


Figure 4.4. Scenario B Output - Reformulated Model

From Chapter 2, the original MACS model is unable to satisfy any demand at FOB1 due to the inability to pull resources into the node. The reformulated tool delivers 9,321 gallons of fuel to meet 31% of FOB1's fuel demand by pulling resources across FOB1's land connection with B1. We confirm the reformulation of the push/pull constraint is working properly and provides a substantial benefit to the model's ability to utilize all connectors in the network to their full potential.

4.2 Rounding Techniques

In Chapter 3, we introduce the capability to apply different rounding techniques to the run counts calculated for each connector. In this section, we perform an analysis of each of the techniques to illustrate the pros and cons of the three different approaches.

We provide the fractional run counts for Scenario B in Table 4.7. We see the current rounding approach (Method 1: round up to the nearest integer) delivers 5,719 gallons of excess fuel to shore. In this section, we perform an analysis on the three different rounding methods to illustrate the differences with each method.

Scenario B Run Counts			
(s, l)	$runCount_{s,l}$	Rounding Method 1	Excess Fuel (gal)
LCAC, B1	2.952	3	214
LCU, B1	0.575	1	765
CH-53, LZ1	4.025	5	2,340
CH-53, LZ2	1.400	2	1,440
MV-22, LZ1	3.600	4	320
MV-22, LZ2	25.200	26	640

Table 4.7. Scenario B Run Counts

We create a scenario builder model to generate 100 scenarios with random land networks, demands, and available connectors. The details of the scenario builder model are further discussed in Chapter 5. We then apply our tool across the 100 different scenarios and calculate averages for each of the MOPs. The comparative results appear in Table 4.8.

Rounding Method Comparison			
Method	Average Demand Satisfied (%)	Average Excess Flow (gal) [Fraction of Total Demand]	Average Number of Runs
1	93.55	9,321 [20.24%]	34
2	88.19	2,263 [4.67%]	30
3	75.93	0 [0.00%]	26

Table 4.8. Rounding Method Comparison

As defined in Section 3.4, excess flow is the amount of fuel delivered ashore not consumed (i.e., demanded) by the land nodes. In real world operations, extra fuel may not be a concern; however, we see from our results the MACS2 tool will deliver an average of 9,300 gallons of excess fuel when utilizing rounding Method 1 (round up to the nearest integer value). The amount of excess fuel delivered by Method 1 is over 20% of the total demand requested by the land nodes. These results provide us and the user with a thorough understanding that the amount of excess fuel expected to be delivered in a given scenario is substantial. Moreover, the user must assume the risk that some fuel might be returned to the seabase, as ground forces cannot consume or store it ashore. In these cases, there is the possibility that connectors are being scheduled to perform runs in which they might only deliver a few hundred gallons of fuel resulting in inefficiencies in the schedule. While almost guaranteeing excess fuel will be delivered to the land nodes, Method 1 provides the highest amount of demand satisfied and should be used in cases where the delivery of excess fuel causes no concern.

Utilizing Method 2, rounding to the nearest integer value, results in scheduling approximately five fewer runs than Method 1 and we see a significant decrease in the amount of excess fuel delivered. At just over 2,200 gallons, the amount of excess fuel is more manageable and reduces the risk of scheduling runs in which a connector will deliver a minimal amount of fuel. However, the reduction in excess fuel has reduced the amount of demand fulfilled at the land nodes by 5%. We determine Method 2 provides us the ability to mitigate the amount of excess fuel with the assumption we will not be able to consistently attain the same amount of demand satisfied as Method 1.

When rounding down to the nearest integer, Method 3 fulfills on average 75% of the demand ashore; however, the risk of delivering excess fuel is completely eliminated. Moreover, Method 3 is scheduling eight fewer runs than Method 1 stemming the reduction of almost 20% of the demand satisfied ashore. This method guarantees there is no amount of excess fuel delivered to the land nodes and is a preferred choice when the fuel capacities ashore are limited.

The introduction of different rounding techniques provides the commander with a list of options to incorporate during amphibious operations. Each of the three rounding techniques has their own set of tradeoffs; however, it is important for commanders to understand these tradeoffs and apply their own utility to each unique situation presented.

4.3 Constructing the Best Schedule

The MACS2 tool is now capable of incorporating five different modifications for building a particular schedule, each offering a unique set of capabilities to meet the user's needs. In this section, we demonstrate the capability of layering these constraints to build a schedule relating directly to what the user desires.

Consider a situation in which we are on the MEU staff and are tasked with delivering fuel to a land network ashore with specific commander's guidance. The land network, referred to as Scenario D, is illustrated in Figure 4.5 and consists of three nodes: B1, LZ1, and FOB1. Inputs and further details of the network appear in Table 4.9. Additionally, the MEU commander has issued the following guidance for supporting land forces ashore:

1. FOB1 is the priority node and must receive as much fuel as possible in order to meet its demand, followed by B1 and LZ1.
2. Due to limited resources, fuel consumption cannot exceed 10,000 gallons across all connectors during the course of the day (assuming an eight hour work day).
3. All land nodes ashore must receive at least 25% of their requested fuel demand.
4. Due to limited storage capacities ashore, the seabase cannot deliver any amount of excess fuel to land nodes ashore. We assume each connector departs the seabase carrying its full capacity of bulk fuel.

Scenario D Inputs			
Connectors			
Type	Category	Quantity	Capacity
LCAC	Surface	1	1,800
LCU	Surface	1	2,700
CH-53	Air	2	1,600
MV-22	Air	2	800
MTVR	Land (B1,FOB1)	2	1,800
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air, Land	20,000	LZ1,FOB1
LZ1	Air, Land	15,000	B1
FOB1	Air, Land	15,000	B1

Table 4.9. Scenario D Inputs

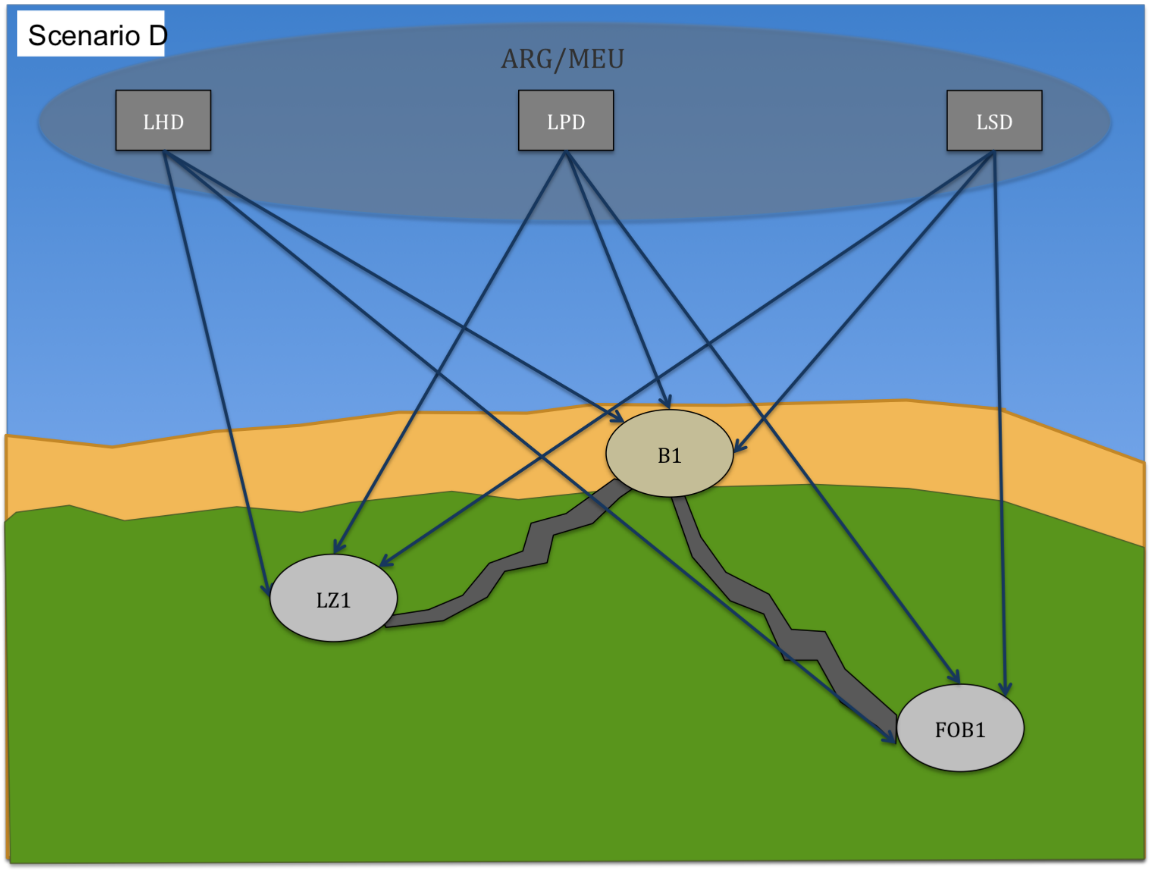


Figure 4.5. Scenario D Network Illustration

We first implement a base schedule using priority node assignment as given in the commander's guidance. Next, we analyze the output and determine if further action is required in order to obtain a solution that aligns more closely to the commander's desires. The base schedule MOPs appear in Table 4.10 along with the flow output illustrated in Figure 4.6. Additionally, we have added "Excess Fuel" as an MOP to measure the amount of unconsumed fuel delivered to the land nodes. In our base schedule, we utilize Rounding Method 1 (round up to the nearest integer).

MOPs				
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)	Excess Fuel (gal)
97.72	48,860	42	11,653	5,940

Connector Run Count		
Connector	Land Node	Run Count (Total=42)
LCAC	B1	3
LCU	B1	2
CH-53	B1	3
	LZ1	6
	FOB1	9
MV-22	B1	12
	LZ1	7
	FOB1	0

Table 4.10. Scenario D Output (Baseline)

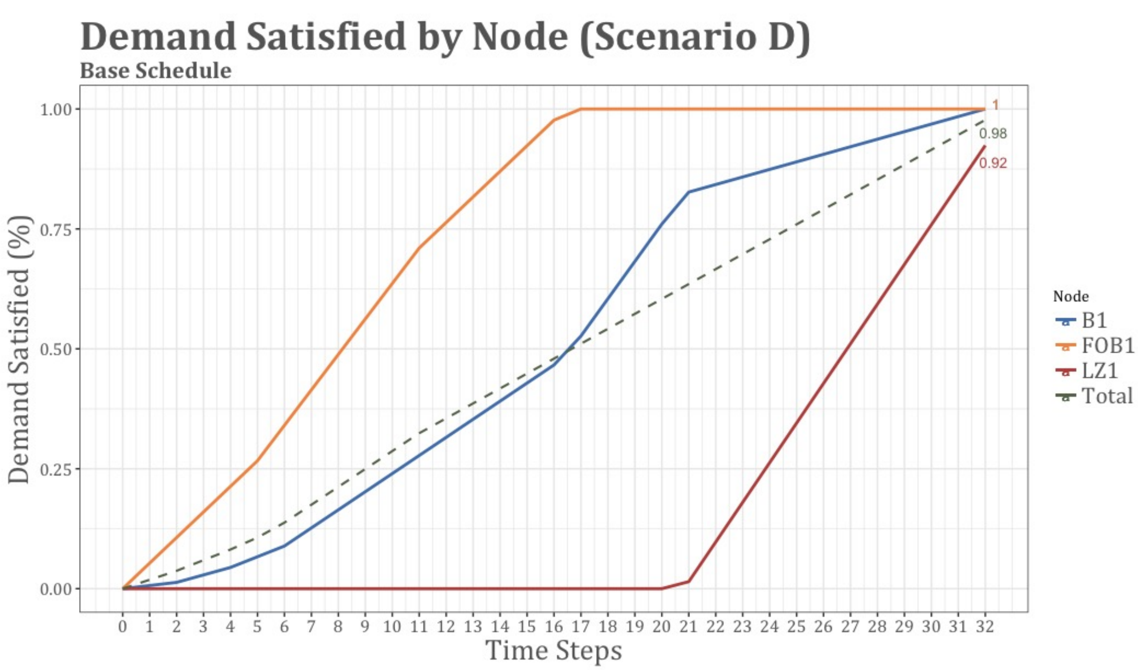


Figure 4.6. Scenario D Output (Baseline)

The base schedule enables us to satisfy over 97% of the demand of fuel needed ashore; however, this schedule results in pushing almost 6,000 gallons of unconsumed excess fuel. Additionally, the fuel needed to carry out this schedule exceeds the 10,000-gallon fuel consumption limit.

Using the results from our initial output, we utilize the fuel constraint to limit the amount of fuel consumed by the connectors. We set our fuel constraint to limit the total amount of fuel consumed by each connector type to one tank of fuel over the course of the eight-hour time horizon. Considering the quantity of connectors being utilized in the scenario, consuming only one tank of fuel per connector type should keep the schedule's fuel consumption below the 10,000-gallon limit. Results of this change are outlined in Table 4.11 and Figure 4.7.

MOPs				
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)	Excess Fuel (gal)
63.49	31,748	25	8,591	3,051

Connector Run Count		
Connector	Land Node	Run Count (Total=25)
LCAC	B1	3
LCU	B1	2
CH-53	B1	1
	LZ1	0
	FOB1	9
MV-22	B1	10
	LZ1	0
	FOB1	0

Table 4.11. Scenario D Output (Fuel Constrained)

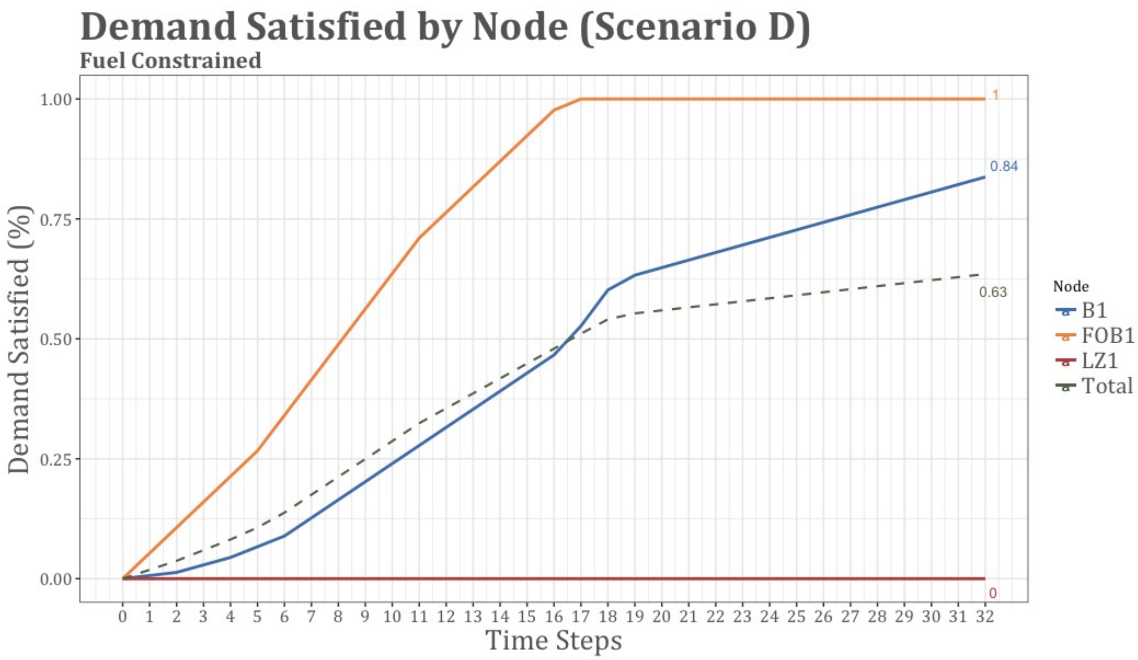


Figure 4.7. Scenario D Output (Fuel Constrained)

From the results, applying the fuel constraint produces a schedule that meets the commander's criteria of not exceeding the 10,000-gallon fuel consumption limit; however, this schedule results in delivering over 3,000 gallons of excess fuel ashore. More importantly, applying the fuel constraint has resulted in LZ1 receiving no deliveries of bulk fuel.

We next implement the equity constraint and set our minimum demand value p to 0.25 to ensure all nodes receive at least 25% of their requested fuel demand. Additionally, to reduce the amount of unconsumed excess fuel we apply Method 3 for our rounding technique to ensure no amount of excess fuel is pushed ashore. Outputs from the updated schedule appear in Table 4.12 and Figure 4.8.

MOPs				
Demand Satisfied(%)	Bulk Fuel Moved (gal)	Number of Runs	Fuel Consumed (gal)	Excess Fuel (gal)
59.40	29,700	22	8,591	0

Connector Run Count		
Connector	Land Node	Run Count (Total=22)
LCAC	B1	3
LCU	B1	1
CH53	B1	0
	LZ1	1
	FOB1	8
MV22	B1	7
	LZ1	2
	FOB1	0

Table 4.12. Scenario D Output (Fuel Constrained/Equity/Round Down)

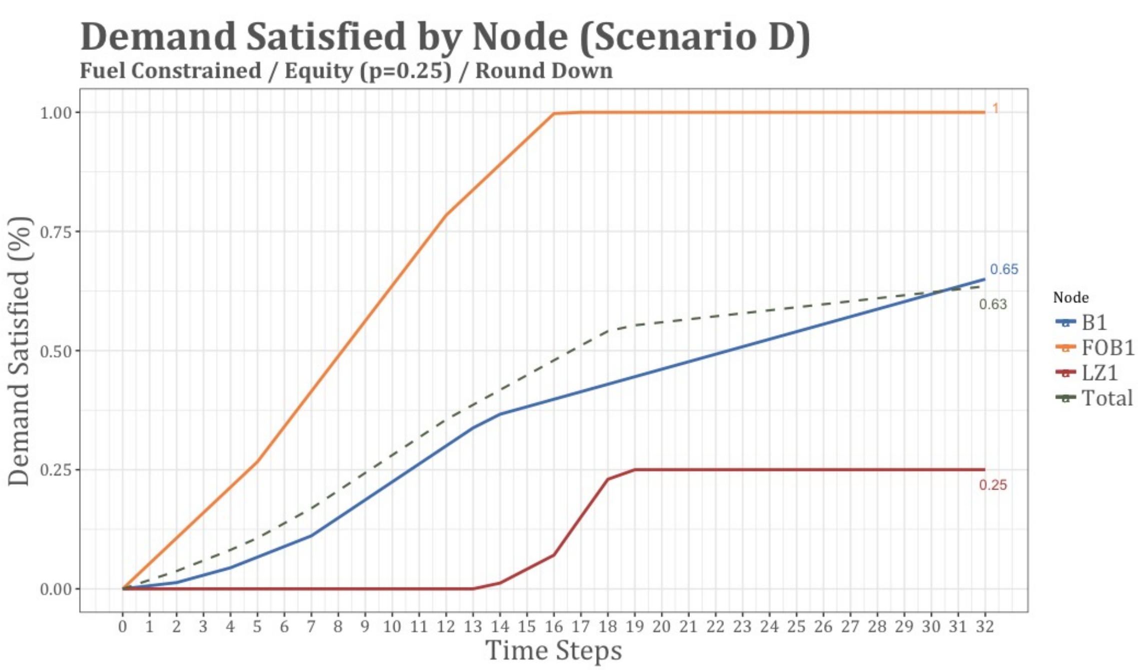


Figure 4.8. Scenario D Output (Fuel Constrained/Equity/Round Down)

The output from this schedule results in all nodes receiving at least 25% of their respective fuel demands with FOB1 receiving 100% as it is designated the highest priority node. Furthermore, the fuel consumption rate does not exceed the 10,000-gallon limit as set by the commander. While the output from Figure 4.8 shows a maximum of 63% total demand satisfied across the land nodes, utilizing rounding method 3, we only deliver 59.4% or 29,700 gallons to guarantee we do not deliver any amount of excess fuel to shore.

The above example demonstrates the capability of layering the five different modifications discussed in Chapter 3 to produce a precise schedule to meet the user's requirements. In our example, the MACS2 tool is capable of building a schedule to meet 97% of the fuel demand ashore; however, under the commander's guidance we can build a schedule to only meet 59%. The MACS2 model has the capability of providing a unique set of options to meet the commander's intent offering a product that creates a feasible and efficient connector schedule specific to what the MEU staff requires.

4.4 Conclusion

We provide a comparison of our reformulated model against the original model implemented in Christafore (2017). While effective, the original MACS tool requires several improvements in order to be more applicable to the MEU staff. These results prove our MACS2 tool is now more capable than the original model and offers a set of unique capabilities to construct a schedule more specific to the user's needs. In the next chapter, we perform a set of simulations in which we test our schedules under various adverse conditions to determine how well our schedules perform in real world applications.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 5:
Analysis of Uncertainty

In this chapter, we evaluate how well the MACS2 model performs under uncertainty. In these cases, we implement a variety of threat conditions, connector failure rates, and weather concerns to measure the vulnerability of a connector's schedule when such events occur. We then implement different approaches to develop a schedule less vulnerable to adverse situations.

5.1 Overview

In previous chapters, we assume perfect conditions (no weather considerations, threat concerns, or connector failures); however, our aim in this chapter is to provide an estimate of how much fuel we will actually deliver under real-world conditions. Throughout the course of an amphibious operation connectors may fail. Weather considerations may make it impossible to use certain types of connectors; for example, if the sea state is too high LCACs will be unavailable for the day's operation. Moreover, specific threat concerns in the Area of Operations (AOR) such as mines or Surface-to-Air Missile (SAM) sites leave us unable to operate specific connectors. These adverse circumstances are challenging to anticipate and difficult to manage when they inevitably occur. One solution is to re-run the MACS2 tool and build a new schedule as these situations develop; however, the implementation of an entirely new schedule based on changing events during the middle of day may not be feasible. To analyze how our MACS model performs against these unpredictable events, we perform two different analyses: 1) Initial Connector Availability and 2) Intra-Day Connector Failures.

Analysis 1 (Initial Connector Availability) emphasizes knowing specific information at the start of the day. For example, we know there is a mine threat in the area and we are unable to utilize surface connectors, or we anticipate only performing a particular number of runs with the MV-22 before the connector has a mechanical failure. Analysis 1 portrays the expected performance of delivering resources to shore on a day-to-day basis during a deployment. Moreover, Analysis 1 can be executed across a wide range of connector portfolio configurations to educate the MEU staff on which configuration maximizes the delivery of supplies ashore for a particular deployment over a wide range of uncertainties.

Analysis 2 (Intra-Day Connector Failures) involves applying uncertainty during day-to-day operations of the MEU, focusing specifically on connector failure. In this analysis, we begin executing a schedule and, as the day progresses, connectors begin to break down due to mechanical failures. This analysis enables us to develop more robust schedules in an attempt to anticipate connector failures before they occur.

Before executing our two different analyses we first build a set of “pre-built” scenarios to encompass a multitude of circumstances that could develop during a MEU deployment. Each pre-built scenario is considered as the baseline parameters for a given amphibious operation. We utilize a stochastic approach to build sets of random input parameters for each scenario, and write the input parameters to the necessary CSV files outlined in Chapter 2. Each scenario encompasses a variety of demands, containers, and connectors. Furthermore, we construct diverse networks in which we vary the number of land nodes, the connector spots available at each land node, and the distances across both the land network and seabase. A more detailed discussion of the scenario builder script can be found in Appendix C. Our scenarios are then applied to our two analyses to produce a detailed evaluation of the MACS2 tool’s performance when uncertainty arises.

5.2 Analysis 1: Initial Connector Availability

As stated, Analysis 1 focuses on expected resource delivery performance during a MEU deployment. In this analysis, we construct a set of 30 “sub-scenarios” that we apply to each pre-built scenario mentioned above. Each sub-scenario accounts for a set of diverse events such as mechanical failures, weather conditions, or threat concerns that can occur. The 30 sub-scenarios are considered as 30 specific days during a deployment to capture a range of uncertainties that arise. These sub-scenarios perturb the inputs from the CSVs to reduce connector availability before a schedule is constructed. These inputs constrain the MACS2 tool’s ability to utilize specific connector types and limit the number of runs each connector can perform to deliver fuel ashore.

Each sub-scenario differs according to five factors outlined below. Table 5.1 depicts the 30 different sub-scenarios constructed utilizing these factors.

1. **Sea State:** Sea states are based on the Beaufort scale and range in value from 0 (calm) to 5 (moderate waves) (The Editors of Encyclopaedia Britannica 2018). We choose the values of sea states based on a uniform distribution with minimum value zero and maximum value five. With each sea state above sea state three, we degrade the velocity at which the surface connectors can operate by 20%. For example, operating in sea state four, an LCAC's maximum velocity is degraded from 30 knots to 24 knots. If the sea state is five, we decrease the LCAC's velocity to 18 knots.

2. **Weather (Air):** We use a binary value to implement weather conditions for aircraft. Zero represents weather conditions do not facilitate the use of aircraft, and one represents weather conditions are adequate for air operations. We use a Bernoulli random variable which takes the value of zero with probability of 0.20 and one with probability of 0.80.

3. **Threat (Surface):** Surface threats are represented as a binary variable. Zero signifies a surface threat is perceived and the use of surface connectors is not available, and one represents no threat, the use of surface connectors is allowed. Once again, we use a Bernoulli random variable which takes the value of zero with probability of 0.20 and one with probability of 0.80.

4. **Threat (Air):** Similar to the surface threat parameter, air threats are based on a binary variable. Zero represents an air threat is apparent and the use of air connectors is not allowed, and one represents no threat, the use of air connectors is approved. Probabilities of an air threat are consistent with the probabilities of a surface threat.

5. **Maximum Number of Runs per Connector:** We implement connector failures into the analysis by limiting the maximum number of runs a specific connector type can perform. For each connector type, we utilize a Geometric random variable with probability of 0.10 for surface connectors and 0.20 for air connectors to determine the number of successful round trips until failure. Once a connector type has reached its maximum number of round trips, we assume all individual connectors of the connector type has had a maintenance failure and can no longer be used to deliver fuel.

Sub Scenario Inputs (Analysis 1)								
Sub Scenario #	***Sea State***	***Weather (Air)***	***Threat (Surface)***	***Threat (Air)***	***Maximum Number of Runs Per Connector***			
					LCAC	***LCU***	***CH53***	***MV22***
1	5	1	0	1	0	0	3	1
2	0	1	1	1	19	27	4	15
3	5	1	1	1	7	28	8	2
4	1	0	1	1	13	7	0	0
5	2	0	1	1	15	17	0	0
6	3	1	1	1	15	8	1	2
7	5	1	1	0	5	7	0	0
8	4	1	1	1	8	10	13	1
9	4	1	1	1	17	1	4	9
10	0	1	1	1	1	4	6	8
11	4	1	1	1	3	2	3	16
12	5	1	1	1	15	9	2	2
13	4	1	0	1	0	0	7	2
14	0	1	1	1	3	7	2	9
15	2	0	0	1	0	0	0	0
16	5	1	1	1	2	13	22	13
17	3	0	1	0	4	3	0	0
18	1	1	0	1	0	0	4	1
19	4	0	1	1	1	2	0	0
20	4	1	1	1	7	13	3	8
21	2	0	1	1	2	10	0	0
22	1	1	1	1	6	3	3	5
23	5	1	0	1	0	0	3	1
24	1	1	0	0	0	0	0	0
25	3	1	1	1	4	6	10	2
26	0	1	1	1	15	4	2	8
27	0	1	1	1	58	22	12	41
28	5	0	1	1	16	10	0	0
29	4	1	0	0	0	0	0	0
30	3	1	0	1	0	0	3	6

Table 5.1. Sub-Scenario Inputs (Analysis 1)

Outputs from Analysis 1 appear in Table 5.2, along with the 95% confidence intervals for each MOP. Additionally, we provide comparative results using the "baseline" schedules: schedules produced assuming perfect conditions (no weather considerations, connector failures, or threat concerns). Constraining MACS2 with adverse events results in Analysis 1 executing almost half the number of runs compared to the baseline schedule. The reduction in the number of successful runs decreases the average amount of fuel successfully delivered by over 20,000 gallons. This reduction in fuel delivery results in accomplishing an average of 60% of demand needed ashore, a decrease of 33% from the baseline schedules.

Outputs (Analysis 1)			
Type	Average Demand Fulfilled (%)	Average Bulk Fuel Delivered (gal)	Average Number of Runs
Baseline	93.55 (±1.89)	51,520 (±3,488)	34 (±2.56)
Analysis 1	60.27 (±2.29)	31,480 (±1,910)	19 (±1.14)

Table 5.2. Outputs (Analysis 1)

We conclude the number of runs a connector can execute has a direct impact on the amount of demand successfully delivered to shore. A 40% decrease in the number of runs reduces the amount of fuel delivered by over 30%. The introduction of uncertainties into the model has provided results that enable us to understand the implications of adverse conditions that arise in real world operations. In addition, Analysis 1 offers MEU planners an understanding of the estimated amount of resources they can realistically expect to push ashore during an amphibious operation.

5.3 Analysis 2: Intra-Day Connector Failure

The second analysis focuses on constructing an initial schedule at the beginning of the day, and analyzing the performance of the schedule as connector failures occur throughout its execution. For this analysis, we begin with an initial quantity of available connectors and construct schedules on the pre-built scenarios only. As connectors begin to execute their schedule, we introduce failure probabilities to mimic mechanical failures. With each successive run of the connector, the probability of failure increases to simulate the mechanical degradation of connector assets throughout the day. Analysis 2 affords us the capability of examining how the initial schedule created at the beginning of the day performs with respect to day-to-day operations on the MEU.

Analysis 2 contains four inputs:

1. **Probability of Failure:** The failure probability is defined as the probability a connector succumbs to maintenance failure on a given run. We initialize the failure probabilities for each connector at the start of the day's operations, and define this probability as the initial failure probability.

2. **Degradation Value**: The degradation value is the amount we add to the failure probability with the success of each round trip. For example, a degradation value of 5% means a connector's probability of failure increases by 5% (in absolute terms) with each successful delivery. If an LCAC begins with an initial probability of failure of 10%, the connector's probability of failure increases to 15% on the next run given the first trip was successful.

3. **Repair Probability:** The repair probability is the probability a connector can be repaired given it broke-down on a given run. If a connector can be repaired, the failure probability of the connector is reset to its initial failure probability when it returns to service. For our analysis we utilize three different repair probability values to range from worst-case (connectors cannot be repaired) to best case (connectors are repaired after every failure).

4. **Repair Time (minutes):** Repair time is defined as the amount of time needed to repair a connector given the connector can be repaired. The repair time is based on a normal random variable with a specified mean and a standard deviation of five minutes.

We conduct Analysis 2 in two parts: Part 1) hold initial probabilities of failure constant, and vary the probability a connector can be repaired and how much time it takes to repair it, and Part 2) hold repair probabilities and repair times constant, and vary the initial probabilities of connector failure.

5.3.1 Analysis 2: Part 1

In Part 1, we hold the initial probabilities of failure constant and vary the repair probabilities and repair times. Inputs for the analysis appear in Table 5.3.

Analysis 2 (Part 1) Inputs				
Connector Category	Initial Failure Probability	Degradation Value	Probability of Repair	Mean Repair Time (min)
Surface	0.05	0.05	0, 0.5, 1	30, 60, 90, 120, 180
Air	0.15	0.05	0, 0.5, 1	30, 60, 90, 120, 180

Table 5.3. Analysis 2 (Part 1) Inputs

We first compare the amount of demand satisfied for each of the experiments from Table 5.3. We also compare our results against the amount of demand satisfied under perfect conditions (no connector failures), as denoted by the dotted line in Figure 5.1. Additionally, we provide the 95% confidence intervals for each analysis.

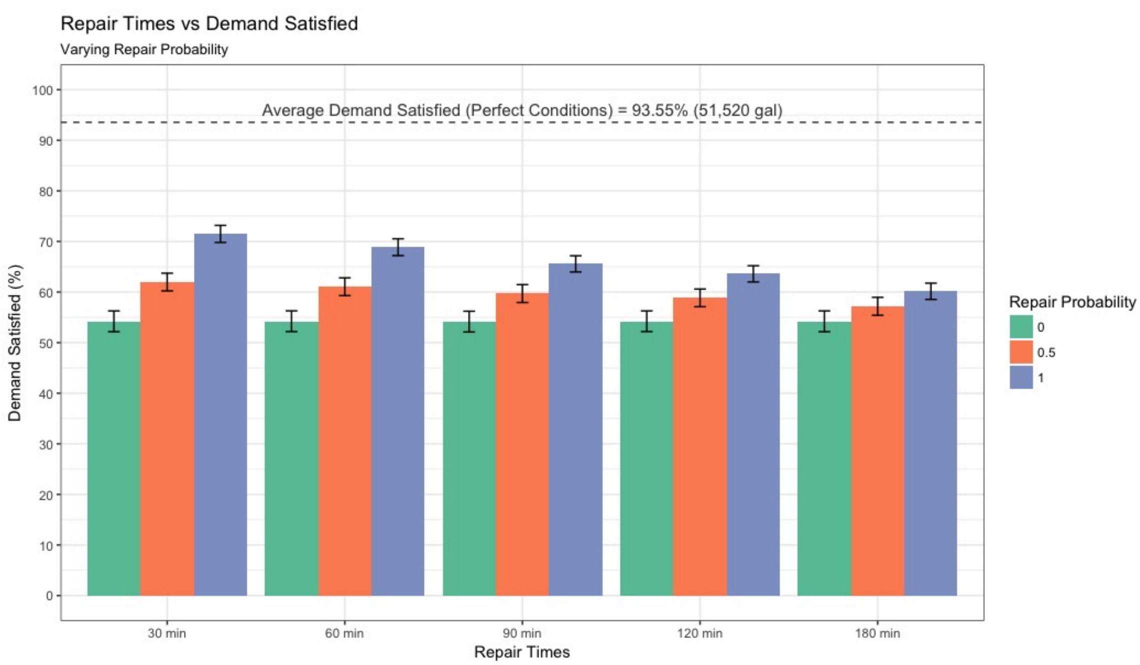


Figure 5.1. Analysis 2 (Part 1) - Demand Satisfied

As the repair probability increases, the amount of demand satisfied ashore increases substantially for smaller repair times. We determine the possibility of repairing a connector and returning the equipment to service produces a more noteworthy impact than the repair time itself. Given there is a 50% chance we can repair a connector; we see an average increase of 5.54% (3,515 gallons) of fuel delivered over a repair probability of 0%.

Figure 5.2 displays the average number of runs for each of the experiments in our analysis, and the average number of runs under perfect conditions is displayed by the dotted line with a value of 34.23. With a repair probability of 0% (worst-case), the analysis indicates only 19.49 runs are successfully performed. The average number of runs achieves a value of 22.41 when the repair probability increases to 50%. The three additional runs directly contribute to the 3,515-gallon increase in the amount of fuel delivered to shore as discussed earlier. This observation sheds light on a possible approach to build a more robust schedule. Through ensuring a connector can achieve more successful runs to shore, we increase the amount of fuel we can deliver to the land nodes.

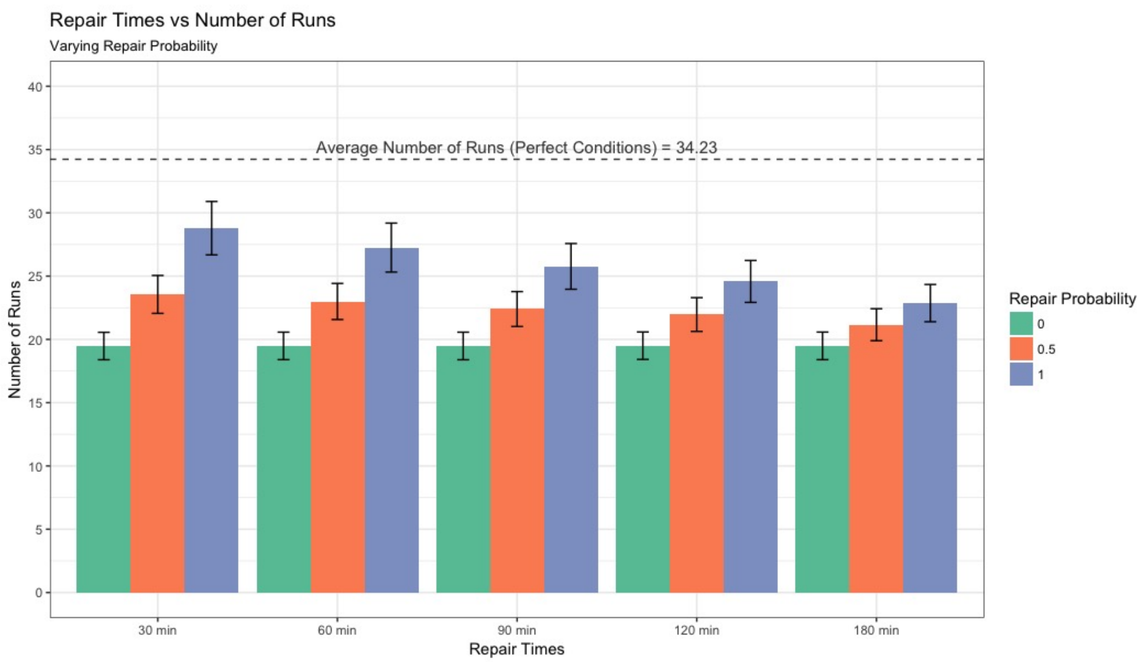


Figure 5.2. Analysis 2 (Part 1) - Number of Runs

Figure 5.3 illustrates the average finish times of the connectors' schedules across varying repair times. We define the finish time as the moment in which the last successful run is completed across all connectors. With a typical well deck/flight crew day consisting of 10 hours (600 minutes), we witness an opportunity to schedule additional runs within the time constraints of an average workday. Under perfect conditions, we obtain a mean finish time of 425 minutes, leaving us an average of 2.5 hours to schedule additional connector runs.

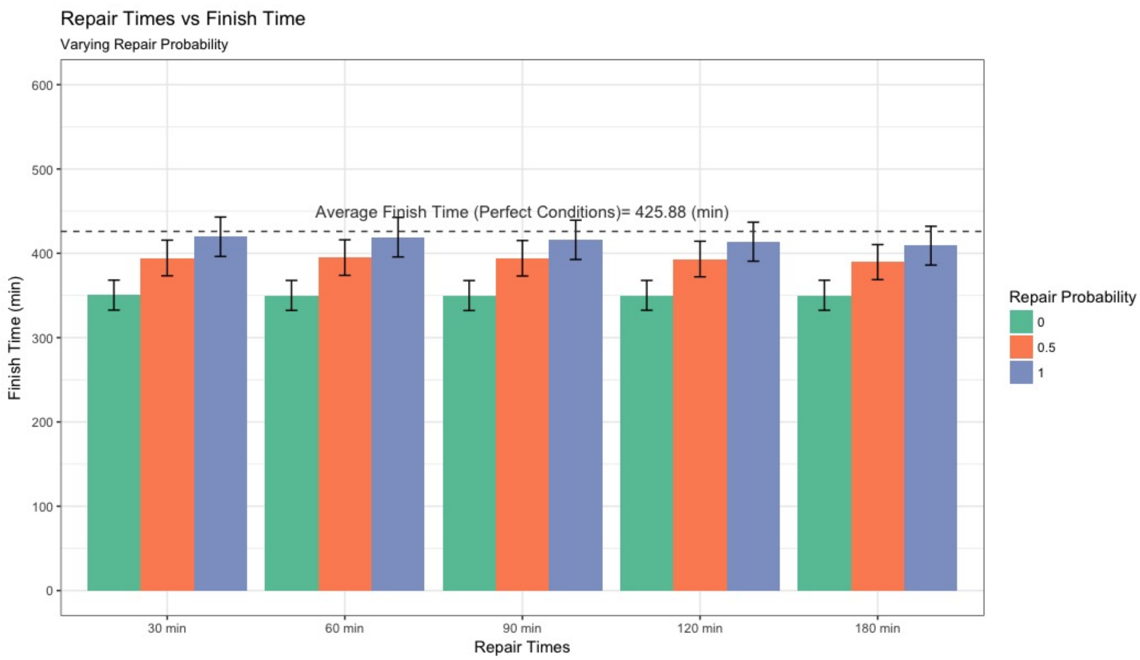


Figure 5.3. Analysis 2 (Part 1) - Finish Time

Part 1's results provide us with insight into how we can increase the amount of bulk fuel delivered to shore. Through increasing the number of runs a connector can successfully make, we deliver more fuel and satisfy more demand to the land nodes. Moreover, we see there is adequate time available during the day to schedule additional runs.

5.3.2 Analysis 2: Part 2

In Part 2, we hold the repair probability and repair time constant at 0.5 and 120 minutes, respectively. We then differ the initial probability of failure across the connectors to analyze the change in the amount of fuel delivered to the land nodes. The various initial failure probabilities appear in Table 5.4, and we perform the analysis on the same set of pre-built scenarios.

Analysis 2 (Part 2) Inputs				
Connector Category	Initial Failure Probability	Degradation Value	Probability of Repair	Mean Repair Time (min)
Surface	0.05, 0.10, 0.15, 0.20	0.05	0.5	120
Air	0.15, 0.20, 0.25, 0.30	0.05	0.5	120

Table 5.4. Analysis 2 (Part 2) Inputs

Figure 5.4 presents comparative results of each experiment along with the output under perfect conditions. Not surprisingly, as the initial probability of failure increases the amount of fuel delivered ashore decreases. We also see a reduction in the number of successful runs ranging from 16 to 22 aggregated across all connector types. Once again, as the number of successful runs decreases, we see a reduction in the amount of fuel we deliver to shore. An average reduction of two runs, results in a decrease of approximately 2,450 gallons of fuel.

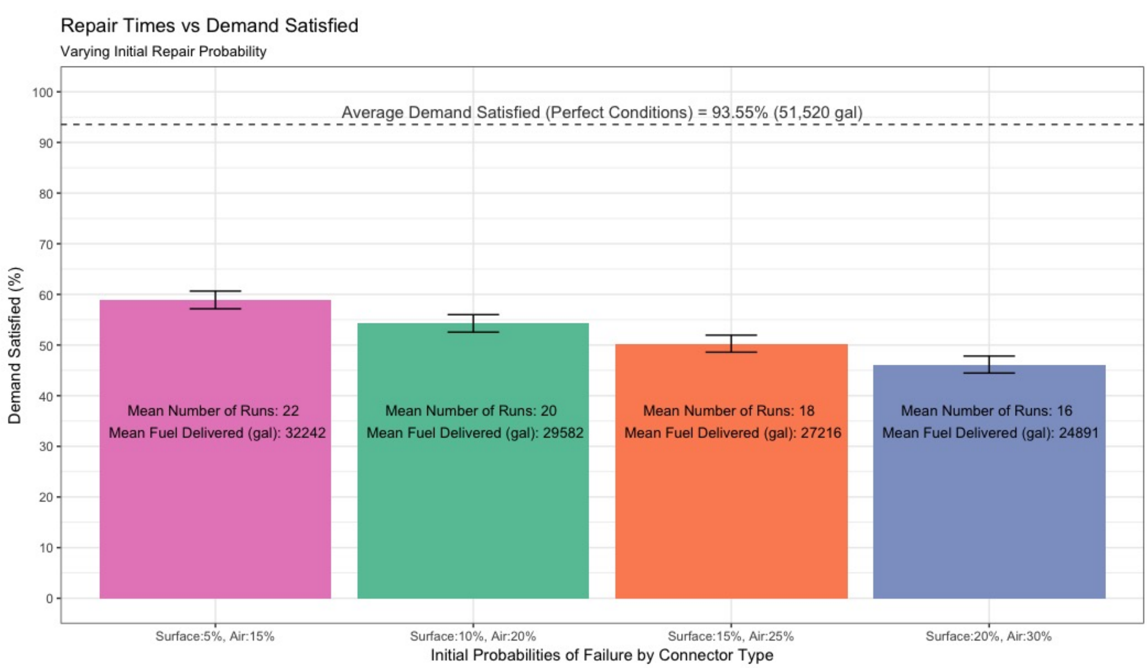


Figure 5.4. Analysis 2 (Part 2) Comparative Results

5.4 Building Robust Schedules

Currently MACS2 produces a schedule that, when implemented successfully, delivers a maximum amount of fuel to shore with the specified inputs. However, if a connector has a maintenance failure and can no longer operate for the remainder of the day, the remaining connector's runs are not executed. Furthermore, if a connector fails and needs repair, it misses critical runs during its repair time. One solution is to rerun the model when these failures occur; however, this solution is inefficient and not conducive to high tempo operations. Our aim is to create a fixed schedule at the beginning of the day capable of handling connector failures without significant disruptions in the amount of fuel delivered to shore.

To accomplish a more robust schedule, we generate a baseline schedule according to the MACS2 model. An example of this schedule is provided in Table 5.5. We see the Quickest Flow model has assigned one LCU run to B1 and two runs to B2. The Assignment Heuristic assigns the sequencing of the runs to occur as one run to B1 followed by two runs to B2. Finally, Scheduler determines the time points. Note in Table 5.5 the LCU's "regular" schedule finishes at 322.5, leaving over 4.5 hours in the work day to schedule any potential lost runs. We aim to take advantage of this remaining time to maximize the number of successful runs the LCU will execute.

We incorporate a more robust schedule using similar logic as the baseline model but we implement a set of "buffer" runs at the end of our baseline schedule to increase the likelihood all the desired fuel will be delivered throughout the day. We define "regular" runs as the runs that are originally scheduled to occur. We define a "buffer" run as a run scheduled subsequent to a connector's regular set of runs. Our approach is to allow the Assignment Heuristic to schedule a regular set of runs for each connector utilizing the same logic in Christafore (2017). Once the heuristic has completed assigning regular runs, we begin assigning buffer runs given there is time remaining in the schedule. We discuss the buffer run logic in the next subsection.

Ship	Run Number	Node	Load	Depart Ship	Arrive at Node	Unload	Depart Node	Return to Ship	Run Type
LPD	1	B1	0	17.5	47.5	47.5	77.5	107.5	Regular
LPD	2	B2	107.5	125	155	155	185	215	Regular
LPD	3	B2	215	232.5	262.5	262.5	292.5	322.5	Regular
LPD	4	B1	322.5	340	370	370	400	430	Buffer
LPD	5	B1	430	447.5	477.5	477.5	507.5	537.5	Buffer

Table 5.5. LCU Schedule with Buffer Runs

Table 5.5 provides an example output of what a connector's schedule looks like after the buffer runs are implemented. In our example, the LCU is scheduled to execute one regular run to B1 and two regular runs to B2. After the regular runs are scheduled, the connector is scheduled to make two buffer runs to B1. In practice we only perform the runs necessary to deliver the required amount of fuel to the appropriate nodes. In the LCU example, if the connector cannot execute some of the regular runs, we have a schedule that can be implemented later in the day if/when the LCU is available.

5.4.1 Buffer Run Assignment

Our fundamental approach to assigning buffer runs is to designate the runs to land nodes which have the greatest prospect of having lost runs during the day. Our approach assigns buffer runs based on a random assignment policy: a connector's next buffer run is scheduled by randomly choosing the next available land node with a probability in proportion to the number of regular runs. Utilizing the LCU example above, there is one regular run scheduled to B1 and two regular runs assigned to B2; therefore, we randomly choose the next buffer run assignment to B1 with a probability of 33% and B2 with a probability of 66%. While simple, this policy considers the number of regular runs previously scheduled across a connector and utilizes this information to randomly add buffer runs.

We developed an additional assignment policy in which the buffer runs are assigned deterministically based on a proportion. In this policy, we allocate buffer runs to land nodes in an effort to ensure the proportion of buffer runs scheduled across the land nodes mirrors the proportion of regular runs. While more sophisticated, this policy produces similar results to the random policy described above, and in the interest of space we omit this approach.

5.4.2 Buffer Run Comparative Analysis

We apply our buffer run assignment approach to Analysis 2 and provide the inputs to the analysis in Table 5.6.

Buffer Run Comparative Analysis Inputs				
Connector Type	Initial Failure Probability	Degradation Value	Probability of Repair	Mean Repair Time (min)
Surface	0.05	0.05	0.5	60, 120, 180
Air	0.15	0.05	0.5	60, 120, 180

Table 5.6. Buffer Run Comparative Analysis Inputs

Figure 5.5 provides the outputs from the analysis and we also provide the output from the current schedule: schedule containing no buffer runs. The application of buffer runs into the schedule produces an improvement in the amount of fuel successfully delivered to shore. With an average increase of 2,600 gallons from the current schedules, our buffer run technique facilitates the delivery of over 5% more fuel to the land nodes. Moreover, the schedules successfully complete an average of three additional runs to shore directly contributing to the additional fuel delivered.

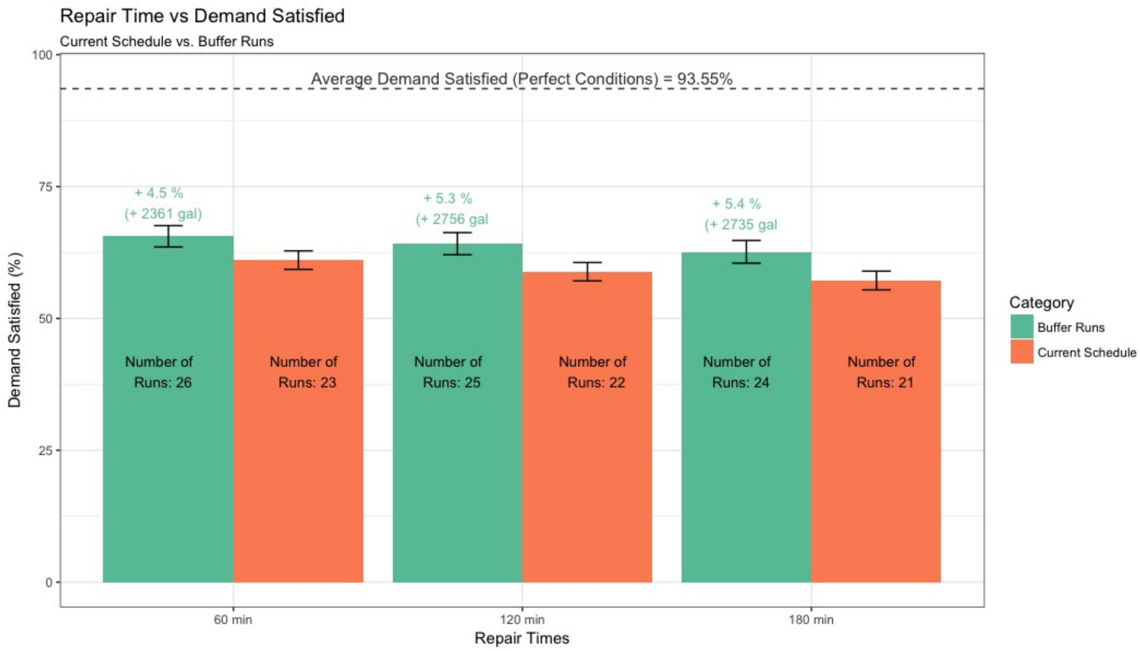


Figure 5.5. Robust Schedule Comparative Results

5.4.3 Excess Runs

We identify a critical issue as buffer runs are added to the schedule: we schedule connectors to land nodes without knowing if the run needs to occur. Without knowing which runs should occur, we could be scheduling unneeded runs. For example, utilizing the LCU example from Table 5.5, an LCU is scheduled to make one regular run to B1 and two regular runs to B2. Additionally, the MACS2 model has allocated two buffer runs to B1 to fill the rest of the time horizon. During execution of the schedule, the LCU successfully executes the regular run to B1, but cannot execute the two regular runs to B2 due to a maintenance failure. If the LCU is repaired in time to begin executing the buffer runs, the LCU does not need to execute the two buffer runs to B1 because the connector has already delivered the required amount of fuel to that node. In this situation, we define these two buffer runs as "unnecessary." In the LCU example, the optimal buffer run assignment results in the two buffer runs being assigned to B2 as this node did not receive any resources. In this section, we rerun our comparative analysis of introducing buffer runs and count the number of unnecessary runs that occur.

An "unnecessary" run must meet three criteria in order to be classified unnecessary: 1) the run must be designated as a buffer run, 2) the buffer run's land node assignment has already received its fuel requirement, and 3) there are other nodes the connector can be assigned to that need fuel. The table below provides a comparison of the average number of unnecessary runs that occur per connector type during our analysis. We can see from Table 5.7, the average number of unnecessary runs per connector type is approximately 1.3; therefore, we determine the average number of unnecessary runs produced for a given scenario across all connectors types is approximately five. We conclude the assignment of buffer runs with our current assignment policy is not producing a significant number of unnecessary runs and our policy is adequately scheduling runs with minimal waste.

Average Number of Unnecessary Runs	
Repair Time (min)	Average Unnecessary Runs per Connector Type
60	1.32
120	1.39
180	1.28

Table 5.7. Average Excess Runs

In practice, one could reschedule missed runs later in the day after all lost runs are known. This approach may not be operationally feasible or practical during high tempo operations. Moreover, this approach may not facilitate the ability to build a new schedule at the end of the day because there is not sufficient time remaining in the work day. Our approach of assigning buffer runs gives a worst case in terms of preparing for lost runs without knowing which runs have been missed. Our buffer assignment policy provides us a reasonable technique to supplement our connector schedules with additional runs in an effort to achieve more successful runs during a work day.

5.5 Conclusion

We provide considerable insight into the amount of fuel the MACS2 tool can successfully deliver to shore in the presence of uncertainty. Through our analysis, we implement a set of buffer runs to build a schedule less vulnerable to unpredictable situations that arise during amphibious operations. The implementation of buffer runs produces a 5% increase in the amount of resources we can successfully deliver to shore without needing to update the inputs into the MACS2 tool and rerun the model.

CHAPTER 6: Allocating Fuel Containers to Connectors

Scenario C from Chapter 2 presents the possibility of producing more efficient schedules through reallocating container assets across the various connectors. In the scenario, we identify the reallocation of containers across connectors can result in delivering the same amount of fuel to shore while utilizing fewer resources. In this chapter, we introduce an algorithm to automate the process of allocating container assets across connectors in order to produce more efficient schedules. Our algorithm is incorporated inside the MACS2 model.

Currently, connector inputs are assumed to be the same across all connectors. For example, a user inputs a capacity of 2,700 gallons (three SIXCONs) for an LCAC. The current MACS2 tool assumes all LCACs in the model have a capacity of 2,700 gallons. In reality, this assumption may not be feasible due to limited container assets. Furthermore, this assumption constrains the user's ability to allocate different capacities across connectors of the same type. For example, the user might employ one LCAC to carry two SIXCONs and another LCAC to carry four.

The workflow of our container assignment algorithm appears in Figure 6.1. We explain the details of the workflow in the following sections, but the figure below illustrates the overall concept. We first build a set of possible container combinations across all connector types. Next we run the Quickest Flow model for each combination. Using the output from Quickest Flow, we calculate an efficient assignment method to assign containers to connectors and connectors to land nodes. Once a proper assignment method has been achieved, we output the total number of runs. Finally, we compare the container combinations and choose the combination with the highest amount of delivered fuel with the fewest quantity of runs.

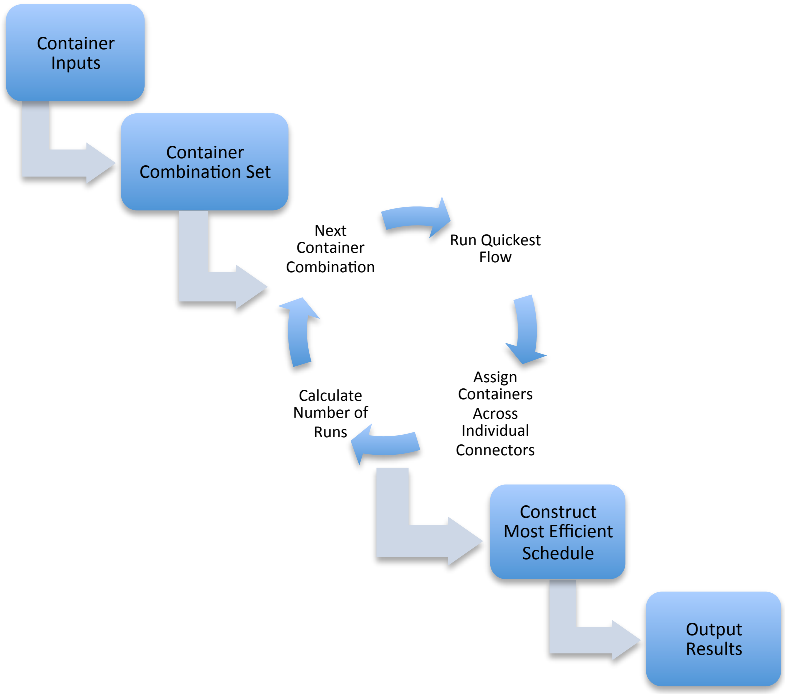


Figure 6.1. Container Allocation Workflow

6.1 Updated Inputs to the MACS2 Model

The algorithm implemented to formulate a feasible and efficient allocation of resources begins with re-examining the connector inputs to the Quickest Flow model. Below we implement an additional set of inputs to establish the quantity of assets the user employs for fuel delivery.

We construct a new CSV file to obtain the information needed to understand the delivery assets available. The CSV contains the type of container, connector category the container is associated with, container capacity (gal), load rate (gal/min), and the quantity of containers available. An example of the CSV file appears in Figure 6.2:

type	category	capacity	loadRate	quantity
sixcon	surface	900	120	24
tbf	air	800	100	18

Figure 6.2. Container CSV Example

6.2 Aggregate Container Combinations

Utilizing the inputs from our container CSV, we construct a set of possible container combinations across the connector types. The set of aggregated container combinations provides all possible groupings in which the containers can be allocated across each connector type: LCAC, LCU, CH-53, and the MV-22. To maximize the use of all available container assets, we require the sum of all containers allocated across the connector types be equal to the total amount of containers inputted in the container CSV. The table below provides an example of the set of aggregated container combinations for distributing three SIXCONs and two TBFDSs across the connector types. We see there are four ways to split the three SIXCONs across two surface connectors and three ways to split two Tactical Bulk Fuel Delivery System (TBFDS) across two air connectors. These different combinations result in 12 different arrangements for distributing the containers.

Aggregate Container Combination Example				
	Surface		*Air*	
Combination	LCAC	LCU	CH-53	MV-22
1	3	0	2	0
2	3	0	1	1
3	3	0	0	2
4	2	1	2	0
5	2	1	1	1
6	2	1	0	2
7	1	2	2	0
8	1	2	1	1
9	1	2	0	2
10	0	3	2	0
11	0	3	1	1
12	0	3	0	2

Table 6.1. Aggregate Container Combination Example

The combination set outlined in Table 6.1 provides the aggregate total quantity of containers allocated to each connector type. In order to achieve a final schedule we must run Quickest

Flow for each combination above and choose the combination that produces the maximum delivery of fuel with the minimum number of runs. We calculate the amount of fuel delivered from the output of Quickest Flow; however, to obtain the run count we must split the containers among the individual connectors inside of each connector type, which we demonstrate in the following section.

6.3 Fuel Container to Individual Connector Assignment Policies

From a fixed container combination set (i.e., one row of Table 6.1), we know the quantity of containers allocated for each container type, the next step is to assign the containers across the quantity of connectors within the connector types. For example, we have 10 SIXCONs assigned to two LCACs; we do not want to examine all possible ways to allocate 10 containers to two LCACs, and then combine these allocations with all combinations of the other connector types. We must find an efficient solution for allocating the 10 SIXCONs across both LCACs.

The output from the Quickest Flow model specifies the amount of flow each connector type is responsible for delivering to the associated land nodes based on the aggregated quantity of containers allocated. We use this output to drive the decision making to assign the containers across the individual connectors. Our base models from Chapters 2 through 5 produce results in which the capacities are consistent across the individuals connectors. For example, we provide an example connector land flow output in the Table 6.2.

Connector Land Flow Example		
Connector Type	Land Node	Flow (gal)
LCAC	B1	3600
LCU	B2	5400
CH-53	LZ1	6400
MV-22	LZ2	4800

Table 6.2. Connector Land Flow Example

In previous chapters, we calculate the run counts to each land node by dividing the connector land flow by the capacity of the connector type. This calculation is represented in Equation (3.22) in Chapter 3. Unlike the previous models, we cannot easily convert the connector land flow to run counts because the capacity of each connector is not constant but depends upon the allocation of containers. For example, from Table 6.2, the LCAC is responsible for delivering 3,600 gallons of fuel to B1. If all LCACs have a capacity of 900 gallons, the LCACs are required to make four runs to B1. However, in this chapter the LCACs do not have the same capacity. For example, we could have two LCACs, one with a capacity of 900 gallons and one with 2,700 gallons, or both with 1,800-gallon capacities. In these cases it is unclear how many runs are required to deliver 3,600 gallons of fuel to B1.

We identify two levels of assignment: 1) aggregated quantity of containers to connector type as discussed in Section 6.2, and 2) allocation of containers across individual connectors. We assign containers across individual connectors by assigning connectors to land nodes based on the containers they are allotted. Specifically, we designate individual connectors to land nodes based on the minimum run count needed to meet the demand at that node. For example, if B1 and B2 require 900 and 1,800 gallons of fuel, respectively, and we have two LCACs with 900 and 1,800-gallon capacities each, we want to ensure each LCAC is assigned to the beaches in a way that minimizes the run count. In our example, the 900-gallon capacity LCAC is assigned to B1 and the 1,800-gallon capacity LCAC is assigned to B2. To appropriately assign the containers to connectors and connectors to land nodes, we adopt four different policies to classify the container to individual connector assignment. We outline the four different policies below.

6.3.1 Assignment Policy 1

Given there is only one connector of a particular connector type, Assignment Policy 1 assigns all containers to that connector. This assignment policy is the simplest and most straightforward.

6.3.2 Assignment Policy 2

Assignment Policy 2 is applied when the number of land nodes m equals the quantity of individual connectors n and the number of individual connectors is greater than one. For example, a scenario contains three LCACs and output from the Quickest Flow model assigns the LCAC to deliver fuel to three land nodes. Our goal is to ensure we properly allocate the containers across these LCACs in an efficient manner.

We begin by creating a set of possible container-connector combinations in order to calculate all possible ways to allocate x containers across n connectors; again n must be greater than one. Table 6.3 provides the set of combinations for the LCAC example above using a total of six SIXCONs allocated across three LCACs. For our individual combination sets, order does not matter but all containers must be utilized; therefore, all values in Table 6.3 are greater than 0.

LCAC Container-Connector Combination		
LCAC-1	LCAC-2	LCAC-3
1	1	4
1	2	3
2	2	2

Table 6.3. LCAC Container-Connector Combination

We now iterate through each possible container-connector combination in Table 6.3 and determine which connectors should be assigned to each land node. We utilize the Hungarian Method from Kuhn (1955) to find the best assignment of connectors to land nodes for a given container-connector combination. The Hungarian Method is an algorithm to assign n personnel to m jobs through minimizing cost. In its simplest terms, the Hungarian Method iterates through all possible assignments and sums the associated assignment costs to find the optimal solution (Kuhn 1955). If n is equal to m, the problem is classified as a Balanced Assignment Problem (Kuhn 1955). For Assignment Policy 2, the number of connectors equals the number of land nodes; therefore, the problem is balanced. We assign the cost as the number of runs to each land node. We apply the Hungarian Method utilizing the Munkres package by Clapper (2017) in Python.

Utilizing the LCAC example in Table 6.3, we step through the first container-connector combination to illustrate the effectiveness of using the Hungarian Method. In the above example, the first container-connector combination represents two LCACs carrying one SIXCON each, and one LCAC carrying four SIXCONs. Below is the cost matrix associated with this container-connector combination with the cost representing the number of runs required for the LCAC to meet the demand at the associated land node. Additionally, we denote the amount of fuel required to be delivered to each land node and have provided the fuel capacity of each LCAC.

	B1 [4000 gal]	B2 [3000 gal]	B3 [2000 gal]
LCAC-1 [900 gal]	5	4	3
LCAC-2 [900 gal]	5	4	3
LCAC-3 [3600 gal]	2	1	1

Table 6.4. Hungarian Method Example (Assignment Policy 2)

Values in the cost matrix are obtained by dividing the required amount of fuel for each land node by the capacity of the connectors. Additionally, we apply Rounding Technique 1 (round up) to ensure we deliver the total amount of fuel required. We then apply the Hungarian Method to our cost matrix to determine the optimal connector to land node assignment. The minimum cost associated with the matrix above is highlighted in green, and we calculate the total number of runs needed to execute this policy is nine. We repeat the same operation for all combinations in Table 6.3, and choose the optimal connector-land node assignment based on the smallest number of total runs.

Iterating through each of the possible container-connector combinations from Table 6.3, we find the most efficient combination and provide the results in Table 6.5. This combination minimizes the number of runs throughout the LCAC's schedule with a resulting run count of seven.

Efficient LCAC Container-Connector Combination Assignment			
Connector ID	Container Qty (SIXCONs)	Land Node	Number of Runs
LCAC-1 [900 gal]	1	B3	3
LCAC-2 [1800 gal]	2	B2	2
LCAC-3 [2700 gal]	3	B1	2

Table 6.5. Efficient LCAC Container Combination Assignment (Assignment Policy 2)

To summarize the combination assignment approach for Section 6.3.2 we determine from the Quickest Flow that the LCAC should utilize six SIXCONs to maximize the delivery of fuel to shore. Moreover, the output from Quickest Flow specifies that the LCACs will deliver fuel to three beaches: B1, B2, and B3. Given we have three LCACs in the model, we apply Assignment Policy 2 because the number of individual connectors equals the number of assigned land nodes. Table 6.5 identifies how we assign the containers and runs from the three LCACs to the three beaches. Note in this case each connector only travels to one beach.

6.3.3 Assignment Policy 3

Assignment Policy 3 is an extension of Assignment Policy 2; however, the number of connectors is greater than the number of land nodes. We begin by subtracting the quantity of connectors until the number of connectors equals the number of land nodes. For example, if we have five LCACs to deliver fuel across three land nodes, we remove two of the LCACs to ensure the quantity of connectors is equal to the number of land nodes. By removing two LCACs from the model we only use three LCACs to deliver fuel; the other two are unused. With this policy we ensure a balanced assignment problem. We then apply Assignment Policy 2 to find the optimal container distribution.

6.3.4 Assignment Policy 4

The most complex assignment policy is when the quantity of connectors is less than the number of land nodes, $n < m$. In these scenarios, we have an unbalance assignment problem as discussed in Betts and Vasko (2016). For these problems, we must implement a unique approach to find an efficient assignment of connectors to land nodes. As discussed in Betts and Vasko (2016), the basic technique to solve an unbalance assignment problem is to make copies of the assets in which we are assigning. By creating copies of the assets we transform the unbalanced assignment problem into a balanced one. We will demonstrate this technique with an example.

We consider the same LCAC example from Section 6.3.2 in which we have three LCACs, six SIXCONs, and three land nodes (B1, B2, B3). To introduce an unbalanced problem, we add an additional land node (B4) so the quantity of connectors is less than the number of land nodes.

We begin by building the same set of possible container-connector combinations as outlined in Table 6.3. Next we must expand our cost matrix in order to accommodate the additional land node. As discussed in Betts and Vasko (2016), we duplicate each of the LCACs and add "dummy" nodes in order to construct a square cost matrix. Additionally, we assign the cost of dummy nodes as zero. We then apply the same Hungarian Method approach to solve the assignment problem. An example of the cost matrix for container-connector combination 1 from Table 6.3 appears in Table 6.6 along with the optimal connector to land node assignment outlined in green.

	B1 [4000 gal]	B2 [3000 gal]	B3 [2000 gal]	B4 [2000 gal]	Dummy-1	Dummy-2
LCAC-1 [900 gal]	5	4	3	3	0	0
LCAC-2 [900 gal]	5	4	3	3	0	0
LCAC-3 [3600 gal]	2	1	1	1	0	0
LCAC-1 [900 gal]	5	4	3	3	0	0
LCAC-2 [900 gal]	5	4	3	3	0	0
LCAC-3 [3600 gal]	2	1	1	1	0	0

Table 6.6. Hungarian Method Example (Assignment Policy 4)

We can see from Table 6.6 the total number of runs utilizing container-connector combination 1 is nine. After iterating through all possible container-connector combinations, the most efficient assignment of assets appears in Table 6.7.

Efficient LCAC Container-Connector Combination Assignment			
Connector ID	Container Qty (SIXCONs)	Land Nodes	Number of Runs
LCAC-1 [900 gal]	1	Dummy-1, Dummy-2	0
LCAC-2 [1800 gal]	2	B2, B3	4
LCAC-3 [2700 gal]	3	B1, B4	3

Table 6.7. Efficient LCAC Container-Connector Combination Assignment (Assignment Policy 4)

The most efficient allocation of assets to land nodes produces a total run count of seven. We see this process once again provides us the number of LCAC runs to each beach just as before and we attempt to adhere to these recommendations but may deviate slightly with the final schedule. For example, LCAC-1 has been assigned to a dummy node. To ensure maximum utilization of resources, the connectors assigned solely to dummy nodes are still utilized by the scheduler. In this example, we assign LCAC-1 as a "free agent" to be assigned to any accessible land node. Within the Assignment Heuristic, LCAC-1, as a free agent, can be assigned to any of the four beaches depicted in Table 6.7. We assign LCAC-1 to all four beaches to ensure the model utilizes the LCAC to the fullest extent possible.

6.4 Integration with the MACS2 Tool

As discussed, the container allocation algorithm is implemented in conjunction with the Quickest Flow model in order to provide the necessary inputs into the tool. Output from the Quickest Flow model provides us with the necessary information to begin assigning containers to connectors and connectors to land nodes. After the assignment of the proper container-connector combination is complete, we must feed these inputs into the Assignment Heuristic for schedule development. The Assignment Heuristic receives the suggested run allocation from the container-connector combination analysis described in Section 6.3 (see Tables 6.5 and 6.7) and produces a rough schedule that feeds into Scheduler for final schedule development.

The Scheduler model requires slight modification in order to support the updated assignment policies. The current MACS2 model assumes the load capacities are the same across all connectors of a given connector type. Under our new assignment policy, this assumption is no longer true as different quantities of container assets are allocated across multiple connectors. We provide the updated formulation of the Scheduler model in Appendix B.2.

6.5 Demonstration of the Container Allocation Model

In this section, we employ our container allocation algorithm inside the MACS2 model to demonstrate its ability to formulate an efficient distribution of resources across the connectors while building feasible connector schedules. We begin by applying our tool to Scenario C featured in Chapter 2. We then employ the tool on a more complex scenario

(Scenario E) to demonstrate the tool's capability to scale to larger networks.

6.5.1 Scenario C

Scenario C features a two-node network (B1, LZ1) and the inputs to Scenario C appear in Table 6.8. We have also outlined the quantity of containers in the table to better understand the inputs into the scenario.

Scenario C Inputs			
Connectors			
Type	Category	Quantity	Containers
LCAC	Surface	1	5 x SIXCON
LCU	Surface	1	
CH-53	Air	1	3 x TBFDS
MV-22	Air	1	
MTVR	Land (B1, LZ1)	1	2 x SIXCON
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface	20,000	LZ1
LZ1	Air, Land	15,000	B1

Table 6.8. Scenario C Inputs

Using the inputs from the Table 6.8, our tool constructs an efficient schedule based on allocating the containers across connectors. In Section 2.3.3, we calculate the optimal container assignment by hand; however, our container allocation algorithm automates this process. Comparing both solutions in Tables 6.9 and 6.10, our algorithm obtains the exact same solution as the one found in Chapter 2.

Scenario C: MOP Comparison		
MOPs	Scenario C: Part 2	Container Allocation Algorithm
Demand Satisfied (%)	98.14	98.14
Bulk Fuel Delivered (gal)	34,620	34,620
Number of Runs	13	13
Fuel Consumed (gal)	1,634	1,634

Table 6.9. Scenario C: MOP Comparison

Scenario C Run Count Comparison			
Scenario C (Part 2)			
Connector	Containers	Land Node	Run Count (Total=13)
LCU	5 x SIXCON	B1	3
MV-22	3 x TBFDS	B1 LZ1	3 7
Container Allocation Algorithm			
Connector ID	Containers	Land Node	Run Count (Total=13)
LCU-1	5 x SIXCON	B1	3
MV22-1	3 x TBFDS	B1 LZ1	3 7

Table 6.10. Scenario C: Run Count Comparison

6.5.2 Scenario E (Complex)

The next scenario features a larger network and significantly more containers to allocate across various quantities of connectors. This scenario demonstrates our algorithm's ability to navigate through a multitude of combinations and find the most efficient distribution of container assets within a reasonable computation time.

We provide the inputs to this scenario in Table 6.11. As before, we continue to implement a time horizon of eight hours with 32 time steps of 15 minutes each.

Scenario E Inputs			
Connectors			
Type	Category	Quantity	Containers
LCAC	Surface	3	18 x SIXCON
LCU	Surface	2	
CH53	Air	2	12 x TBFDS
MV22	Air	4	
MTVR	Land (B2, B3, FOB1)	5	10 x SIXCON
Land Nodes			
Node	Spots	Demand	Land Connection
B1	Surface, Air	15,000	LZ1
B2	Surface, Land	9,000	B3, FOB1
B3	Surface, Land	5,000	B2
B4	Surface	8,000	B2
LZ1	Air	12,000	B1
LZ2	Air	15,000	None
FOB1	Land	20,000	B2

Table 6.11. Scenario E Inputs

Our algorithm iterates through 247 aggregated container combinations producing a total of 2,229 individual container-connector combinations in under three minutes to provide a feasible and efficient connector schedule. Outputs from the model appear in the Table 6.12. The results show a total of 84,000 gallons of fuel is scheduled for delivery, completely satisfying the demand ashore. Furthermore, the tool allocates the SIXCONs across three LCACs and one LCU, and assigns all TBFDSs to the MV-22. We note two LCACs and all CH-53s are not utilized in our schedule. The schedule outputted consists of a total of 28 runs with a total fuel consumption of 2,275 gallons.

Scenario E Outputs			
Demand Satisfied (%)	Bulk Fuel Delivered (gal)	Number of Runs	Fuel Consumed (gal)
100	84,000	28	2,275
Connector Run Count			
Scenario C (Part 2)			
Connector ID	Container Assignment	Land Node	Run Count (Total=28)
LCAC-1	8 x SIXCON	B2	3
LCU-1	5 x SIXCON	B2, B3	2, 1
LCU-2	5 x SIXCON	B4	2
MV22-1	3 x TBFDS	B3	1
MV22-2	3 x TBFDS	LZ1	5
MV22-3	3 x TBFDS	B1	7
MV22-4	3 x TBFDS	LZ2	7

Table 6.12. Scenario E Outputs

The following tables feature the minute-by-minute schedule for each of the connectors in Table 6.12. The tables denote not only the six time points of the schedule but also the ship, connector ID, and node the individual connector is delivering fuel to. We conclude the container allocation algorithm is capable of scaling to larger networks with substantial quantities of assets while still observing a satisfactory computation time.

Ship	Node	Connector ID	Load	Depart Ship	Arrive at Node	Unload	Depart Node	Return to Ship
LHD	B2	LCAC-1	0.0	70.0	80.0	80.0	150.0	160.0
LHD	B2	LCAC-1	160.0	230.0	240.0	240.0	310.0	320.0
LHD	B2	LCAC-1	320.0	390.0	400.0	400.0	470.0	480.0

Table 6.13. LCAC Schedule (Scenario E)

Ship	Node	Connector ID	Load	Depart Ship	Arrive at Node	Unload	Depart Node	Return to Ship
LPD	B3	LCU-1	0.0	47.5	77.5	77.5	125.0	155.0
LPD	B4	LCU-2	47.5	95.0	125.0	125.0	172.5	202.5
LPD	B2	LCU-1	155.0	202.5	238.5	238.5	286.0	322.0
LPD	B4	LCU-2	202.5	250.0	280.0	280.0	327.5	357.5
LPD	B2	LCU-1	322.0	369.5	405.5	405.5	453.0	489.0

Table 6.14. LCU Schedule (Scenario E)

Ship	Node	Connector ID	Load	Depart Ship	Arrive at Node	Unload	Depart Node	Return to Ship
LHD	LZ1	MV22-2	0.0	34.0	40.0	40.0	74.0	80.0
LHD	LZ2	MV22-4	0.0	34.0	40.6	40.6	74.6	81.2
LHD	LZ1	MV22-2	80.0	114.0	120.0	120.0	154.0	160.0
LHD	LZ2	MV22-4	81.2	115.2	121.8	121.8	155.8	162.4
LHD	LZ1	MV22-2	160.0	194.0	200.0	200.0	234.0	240.0
LHD	LZ2	MV22-4	162.4	196.4	203.0	203.0	237.0	243.6
LHD	LZ1	MV22-2	240.0	274.0	280.0	280.0	314.0	320.0
LHD	LZ2	MV22-4	243.6	277.6	284.2	284.2	318.2	324.8
LHD	LZ2	MV22-2	320.0	354.0	360.6	360.6	394.5	401.2
LHD	LZ2	MV22-4	324.8	388.0	394.6	394.6	428.6	435.2
LSD	B3	MV22-1	0.0	34.0	35.5	35.5	69.5	71.0
LSD	B1	MV22-3	34.0	68.0	70.4	70.1	104.4	106.8
LSD	B1	MV22-1	71.0	105.0	107.4	107.4	141.4	143.8
LSD	B1	MV22-3	106.8	140.8	143.2	143.2	177.2	179.6
LSD	B1	MV22-1	143.8	177.8	180.2	180.0	214.2	216.6
LSD	B1	MV22-3	179.6	213.6	216.0	216.0	250.0	252.4
LSD	B1	MV22-1	216.6	250.6	253.0	253.0	287.0	289.4
LSD	B1	MV22-3	252.4	286.4	288.8	288.8	322.8	325.2
LSD	LZ1	MV22-1	289.4	323.4	329.7	329.7	363.7	370.0
LSD	LZ2	MV22-3	388.3	422.3	428.6	428.6	462.6	468.9

Table 6.15. MV22 Schedule (Scenario E)

6.6 Conclusion

Our algorithm has provided a reasonable and efficient approach to allocating different quantities of containers across various connectors. Furthermore, our approach does not produce a significant increase in computation time. More importantly, we provide the user greater flexibility through assigning different capacities across connectors.

CHAPTER 7:
Conclusion

7.1 Summary

We provide a tool more proficient and flexible than the original model created in Christafore (2017). The updates to the Quickest Flow model enables us to consider a more realistic flow of resources to shore through satisfying demand in the shortest amount of time feasible, and incorporating the ability to not only push but also pull resources between land nodes. We also provide a tool capable of producing a wide range of schedules with benefits to each such as fuel efficiencies, equity across nodes, and minimizing the delivering of excess fuel. Additionally, our tool contains the capability of layering the different schedule types to provide increased flexibility to the user.

The application of two different analyses facilitate the introduction of a variety of threat conditions, weather considerations, and maintenance concerns to our schedules in order to offer the MEU staff a greater understanding of how our model behaves under real-world conditions. Moreover, we find implementing buffer runs create schedules more immune to uncertainty. Our buffer run technique provides one method of anticipating unforeseen issues inside of the connectors' delivery schedule, and attempts to mitigate these issues through assigning additional connector runs. The incorporation of buffer runs provide a significant increase in the amount of fuel successfully delivered to shore and establishes a minute-by-minute schedule to execute throughout the entirety of the MEU workday.

A critical drawback to the MACS2 tool is its inability to vary the bulk fuel carrying capacity across connectors of the same type. Our container allocation algorithm provides a solution to this shortfall. Our algorithm calculates an efficient allocation of containers across all connectors to deliver resources ashore based on the minimum number of runs. The algorithm achieves the same solution as the MACS2 model but has the capability of handling different bulk fuel carrying capacities across the individual connectors. Furthermore, through the use of our algorithm we do not see a substantial increase in computation time.

7.2 Future Work

The creation and execution of larger and more complex scenarios into our analysis could provide results that more accurately test the scheduling capabilities of the MACS2 tool. For example, our scenarios focus on MEU offloads alone; however, the MACS tool has the capability of scaling to larger scenarios such as a MEB. Moreover, the implementation of a variety of connector portfolios will provide the MEU staff expected performance results across a range of connector options for deployment.

Our analysis emphasizes the use of connectors and containers currently utilized by the MEU. Introducing a range of future technologies such as Unmanned Aerial Vehicles (UAV) is needed to understand the performance of these technologies during real world operations. Furthermore, our analysis did not utilize the M970 Refueler for transporting fuel to the land nodes ashore or throughout the land network. Incorporating the M970 Refueler into the transportation capabilities of the model is needed to fully understand its impact during amphibious operations.

In conjunction with this thesis, Captain Matt Danielson (2018) develops a multi-commodity scheduling model to construct connector schedules for MEU offloads. His thesis utilizes the network flow model as mentioned in this thesis and Christafore (2017) to produce connector schedules for amphibious sustainment across a multitude of commodities and connector configurations. In his thesis, Danielson (2018) takes into account storage and weight constraints aboard connector types, and incorporates Combat Logistic Patrols to push multiple commodities through the land network. Furthermore, his thesis contains a penalty mechanism to tailor schedules for specific MEU requirements such as utilizing more surface connector assets than air connectors based on the MEU's mission set (Danielson 2018). The developments in Danielson's model coupled with the improvements achieved in this thesis provides a tool with not only direct application to the MEU but also efficient logistical solutions to support the battlefield of the future.

APPENDIX A:
MEU Assets

A.1 MEU Ship Types

The MEU is equipped with the personnel and equipment to provide support and execute a wide range of functions anywhere in the world. Transporting the immense amount of equipment in a short amount of time requires a specific type of ship. Currently the MEU features three different classes of ships for supporting the MEU force: LHD, LSD, and the LPD. Specific descriptions and illustrations of each ship type are depicted in the Figures A.1 through A.3.

Landing Helicopter Dock (LHD)

Specifications:
- Speed: 20 knots
- Landing Craft:
  - 3xLCAC or 2xLCU
- Aircraft:
  - Mix: MV-22's and CH53's

Figure A.1. LHD Class Ship. Source: Specifications: Expeditionary Warfare Training Group Pacific (2016), Photo: Webb (2002)

Landing Ship Dock (LSD)	

<u>Specifications:</u>
- Speed: 20 knots
 	
- Landing Craft:
    - 2xLCACs or 1xLCU
 	
- Aircraft:
    - two landing spots for takeoff/ landing only
	

Figure A.2. LSD Class Ship. Source: Specifications: Expeditionary Warfare Training Group Pacific (2016), Photo: Seeley (2017)

Landing Platform Dock (LPD)	

<u>Specifications:</u>
- Speed: 22 knots
 	
- Landing Craft:
    - 2xLCAC or 1xLCU
 	
- Aircraft:
    - 2xMV-22 or
    - 2xCH53
	

Figure A.3. LPD Class Ship. Source: Specifications: Expeditionary Warfare Training Group Pacific (2016), Photo: Vinson (2006)

A.2 Surface, Air, and Land Connectors

To transport personnel, equipment, and resources ashore, the MEU contains a variety of air and surface connectors to accomplish the task. Each connector type is deployed from one of the different types of amphibious assault ships outlined in A.1, and we classify the connector types into three different categories: surface, air, and land. These connectors are the primary means of providing logistical support to forces ashore. For the purposes of this thesis, only connector types that move bulk fuel ashore are considered. Additionally, this thesis focuses primarily on the surface and air connectors. Figures A.4, A.5, and A.6 present an overview of the different connector types used in this thesis.

Surface Connectors	
Landing Craft Air Cushion (LCAC): - Speed: 50 knots (unloaded) - Range: 200 nm - Payload: 70 tons	
Landing Craft Utility (LCU): - Max Speed: 11 knots (loaded) - Range: self-sufficient for up to 10 days - Payload: 160 tons	

Figure A.4. Surface Connector Descriptions. Source: Specifications: Expeditionary Warfare Training Group Pacific (2016), Photos: Roman-Otero (2003) and Yarnall (2004)

Air Connectors	
MV-22 Osprey - Max Speed: 322 mph - Range: indefinite with aerial refueling - Payload: 7.5 tons (internal)	
CH53E Super Stallion - Max Speed: 173 mph - Range: indefinite with aerial refueling - Payload: 10 tons (internal)	

Figure A.5. Air Connector Descriptions. Source: Specifications: United States Marine Corps (2014), Photos: Hall (2013) and Hoogendam (2016)

Land Connectors	
Medium Tactical Vehicle Recovery (MTVR) - Max Speed: 65 mph - Range: 300 miles - Payload: 7 tons	

Figure A.6. Land Connector Descriptions. Source: Specifications: Jane's By IHS Markit (2017), Photo: Garcia (2013)

A.3 Bulk Fuel Delivery Containers

The movement of fuel to shore takes place in a range of storage options. Additionally, the scenarios analyzed in this thesis feature specific storage systems for stockpiling bulk fuel ashore. Both the fuel storage systems and fuel delivery systems appear in Figure A.7.

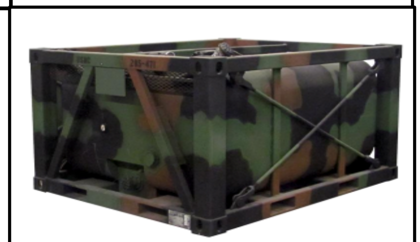
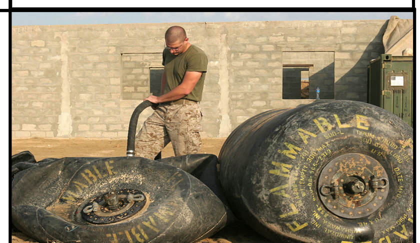
Bulk Fuel Delivery and Storage Containers	
Tactical Bulk Fuel Delivery System (TBFDS) - Capacity: 800 gallons - MV22 and CH53 can carry up to three TBDFSs	 Source: commons.wikimedia.org
Fuel SIXCON - Capacity: 900 gallons - MTVR can carry up to three fuel SIXCONs	 Source: marines.mil
Collapsible Fuel Tank - Capacity: 500 or 3000 gallons - Deployed ashore for fuel storage	 Source: commons.wikimedia.org
M970 Semitrailer Refueler - Capacity: 5000 gallons - Deployed ashore for fuel storage and distribution	 Source: commons.wikimedia.org

Figure A.7. Bulk Fuel Delivery and Storage Containers. Source: Specifications: Department of the Navy, United States Marine Corps (1996), Photos: Kiehl (2011), United States Marine Corps (2009), Rock (2008), and Firehawkv8 (2007)

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX B:
Scheduler Linear Program

B.1 Scheduler Formulation

The Scheduler Linear Program constructs a minute-by-minute schedule utilizing the output from the Assignment Heuristic. The Scheduler incorporates the six time points outlined in Section 2.1.4 to guarantee all events throughout a connector's schedule maintain the correct order. For completeness, we have provided the formulation of the Scheduler Linear Program below. A more detailed discussion of the model can be found in Section 3.3 of Christafore (2017).

Indices and Sets

$s \in S$	Set of amphibious ships
$b \in B$	Set of accessible land nodes by connector type
$i \in I$	Set of six time points outlined in Section 2.1.4
$rs \in Rx_s$	Set of runs for ship s
$rb \in Ry_b$	Set of runs for land node b

Parameters [units]

$numConn_s$	Quantity of connectors for each ship s
$runsShip_s$	Number of runs from ship s
$runsBeach_b$	Number of runs from land node b
$spotsShip_s$	Quantity of welldeck and flightdeck locations for ship s
$spotsBeach_b$	Quantity of landing locations for each land node b
$transT_{s,b}$	Travel time from ship s to land node b [minutes]
ldT_s	Amount of time to load connector s [minutes]
$unldT_b$	Amount of time to unload connector s [minutes]
$dayLength$	Welldeck/flight crew day length [minutes]
$onWSlack$	On water slack time value to limit the wait time of connectors during unloading

$a_{s,rs,b,rb}$	Binary parameter to associate ship departure runs with land node runs. If $a_{s,rs,b,rb} = 1$, the rs departure run of ship s corresponds to the rb run of land node b
$c_{b,rb,s,rs}$	Binary parameter to associate land node runs with ship arrival runs. If $c_{b,rb,s,rs} = 1$, the rb run of land node b corresponds to the rs return run of ship s

Decision Variables

$X_{i,s,rs}$	Time of the ith event of ship run rs departing ship s
$Y_{i,b,rb}$	Time of the ith event of land node run rb departing land node b
$Z_{i,s,rs}$	Time of the ith event of ship run rs returning to ship s
$welldeckStart_s$	Start time of welldeck/flight deck operations on ship s
$welldeckEnd_s$	End time of welldeck/flight deck operations on ship s

Formulation

$$\min_{X,Y,Z} \quad \sum_{s \in S} \sum_{rs \in Rx_s} X_{6,s,rs} \tag{B.1}$$

$$\text{s.t.} \quad welldeckStart_s \leq X_{i,s,rs} \leq welldeckEnd_s \qquad \forall i \in I \forall s \in S \forall rs \in Rx_s \tag{B.2}$$

$$welldeckEnd_s = welldeckStart_s + dayLength \qquad \forall s \in S \tag{B.3}$$

$$X_{2,s,rs} \geq X_{1,s,rs} + ldT_s \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.4}$$

$$X_{3,s,rs} \geq X_{2,s,rs} + \sum_{b \in B} \sum_{rb \in Ry_b} a_{s,rs,b,rb} * transT_{s,b} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.5}$$

$$Y_{4,b,rb} \geq Y_{3,b,rb} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.6}$$

$$Y_{5,b,rb} \geq Y_{4,b,rb} + unldT_b \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.7}$$

$$Y_{6,b,rb} \geq Y_{5,b,rb} + \sum_{s \in S} \sum_{rb \in Ry_b} a_{s,rs,b,rb} * tranT_{s,b} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.8}$$

$$X_{4,s,rs} - X_{2,s,rs} \leq onWSlack \sum_{b \in B} \sum_{rb \in Ry_b} a_{s,rs,b,rb} * tranT_{s,b} \forall s \in S, \forall rs \in Rx_s \tag{B.9}$$

$$X_{i,s,rs} \geq X_{i-1,s,rs} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.10}$$

$$Y_{i,b,rb} \geq Y_{i-1,b,rb} \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.11}$$

$$Z_{i,s,rs} \geq Z_{i-1,s,rs} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.12}$$

$$X_{1,s,rs} \geq X_{2,s,rs-spotsShip_s} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.13}$$

$$Y_{4,b,rb} \geq Y_{5,b,rb-spotsBeach_b} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.14}$$

$$X_{1,s,rs} \geq Z_{6,s,rs-numConns_s} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.15}$$

$$X_{2,s,rs} \geq X_{2,s,rs-1} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.16}$$

$$Z_{6,s,rs} \geq Z_{6,s,rs-1} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.17}$$

$$Y_{4,b,rb} \geq Y_{4,b,rb-1} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.18}$$

$$Y_{i,b,rb} = \sum_{s \in S} \sum_{rs \in Rx_s} a_{s,rs,b,rb} * X_{i,s,rs} \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.19}$$

$$Z_{i,s,rs} = \sum_{\in B} \sum_{rb \in Ry_b} c_{b,rb,s,rs} * Y_{i,b,rb} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.20}$$

$$X_{i,s,rs} \geq 0 \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.21}$$

$$Y_{i,b,rb} \geq 0 \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.22}$$

$$Z_{i,s,rs} \geq 0 \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.23}$$

$$welldeckStart_s \geq 0 \qquad \forall s \in S \tag{B.24}$$

$$welldeckEnd_s \geq 0 \qquad \forall s \in S \tag{B.25}$$

B.2 Scheduler Formulation: Container Allocation

In this section, we highlight the specific changes made to the Scheduler linear program using the container allocation algorithm found in Chapter 6. To properly integrate our container assignments within the Scheduler model we introduce the index $w \in W$, where W is the container combination for a particular connector type. For example, in Section 6.3.2 we identify the most efficient individual container combination across the three LCACs is one container on one LCAC, two containers on the next LCAC, and three containers on the third. In this example, the individual container combination set W takes the value: 1,2,3. We use this index to identify and schedule the individual connectors across the network accordingly.

Additionally, we implement the binary parameter $d_{s,rs,w}$ to link a connector's given departure run from a particular ship with the associated individual container assignment. The parameter $d_{s,rs,w}$ is indexed with s representing the ship the connector is departing from, rs defining the departure run number of ship s, and w depicting the individual container assignment allocated for the connector. We incorporate this parameter to ensure the load/unload times of the connectors are considered during the scheduling process. For example, a connector carrying three containers takes longer to load/unload than a connector carrying only one container. Our aim is to ensure we incorporate the correct load/unload times appropriately in the schedule. We incorporate the parameter $d_{s,rs,w}$ inside of constraint B.29 and constraint B.32 in the Scheduler model formulation. Updates to these constraints ensure the connector does not depart the ship or beach until the containers are fully loaded/unloaded.

Indices and Sets

$s \in S$	Set of amphibious ships
$b \in B$	Set of accessible land nodes by connector type
$i \in I$	Set of six time points outlined in Section 2.1.4
$rs \in Rx_s$	Set of runs for ship s
$rb \in Ry_b$	Set of runs for land node b
$w \in W$	individual connector container configuration

Parameters [units]

$numConn_s$	Quantity of connectors for each ship s

$runsShip_s$	Number of runs from ship s
$runsBeach_b$	Number of runs from land node b
$spotsShip_s$	Quantity of welldeck and flightdeck locations for ship s
$spotsBeach_b$	Quantity of landing locations for each land node b
$transT_{s,b}$	Travel time from ship s to land node b [minutes]
$ldT_{s,w}$	Amount of time required to load connector s with container configuration w [minutes]
$unldT_{b,w}$	Amount of time required to unload connector s with container configuration w [minutes]
$dayLength$	Welldeck/flight crew day length [minutes]
$onWSlack$	On water slack time value to limit the wait time of connectors during unloading
$a_{s,rs,b,rb}$	Binary parameter to associate ship departure runs with land node runs. If $a_{s,rs,b,rb} = 1$, the rs departure run of ship s corresponds to the rb run of land node b
$c_{b,rb,s,rs}$	Binary parameter to associate land node runs with ship arrival runs. If $c_{b,rb,s,rs} = 1$, the rb run of land node b corresponds to the rs return run of ship s
$d_{s,rs,w}$	Binary parameter to associate container allocation w to ship runs. If $d_{s,rs,w} = 1$, the container configuration w is allocated to run rs on ship s

Decision Variables

$X_{i,s,rs}$	Time of the ith event of ship run rs departing ship s
$Y_{i,b,rb}$	Time of the ith event of land node run rb departing land node b
$Z_{i,s,rs}$	Time of the ith event of ship run rs returning to ship s
$welldeckStart_s$	Start time of welldeck/flight deck operations on ship s
$welldeckEnd_s$	End time of welldeck/flight deck operations on ship s

Formulation

$$\min_{X,Y,Z} \sum_{s\in S}\sum_{rs\in Rx_s} X_{6,s,rs} \tag{B.26}$$

s.t.

$$welldeckStart_s \le X_{i,s,rs} \le welldeckEnd_s \qquad \forall i \in I \forall s \in S \forall rs \in Rx_s \tag{B.27}$$

$$welldeckEnd_s = welldeckStart_s + dayLength \qquad \forall s \in S \tag{B.28}$$

$$X_{2,s,rs} \ge X_{1,s,rs} + \sum_{w\in w} ldT_{s,w} * d_{s,rs,w} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.29}$$

$$X_{3,s,rs} \ge X_{2,s,rs} + \sum_{b\in B}\sum_{rb\in Ry_b} a_{s,rs,b,rb} * transT_{s,b} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.30}$$

$$Y_{4,b,rb} \ge Y_{3,b,rb} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.31}$$

$$Y_{5,b,rb} \ge Y_{4,b,rb} + \sum_{s\in S}\sum_{rs\in Rx_s}\sum_{w\in W} unldT_{b,w} * a_{s,rs,b,rb} * d_{s,rs,w} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.32}$$

$$Y_{6,b,rb} \ge Y_{5,b,rb} + \sum_{s\in S}\sum_{rb\in Ry_b} a_{s,rs,b,rb} * tranT_{s,b} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.33}$$

$$X_{4,s,rs} - X_{2,s,rs} \le onWSlack \sum_{b\in B}\sum_{rb\in Ry_b} a_{s,rs,b,rb} * tranT_{s,b} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.34}$$

$$X_{i,s,rs} \ge X_{i-1,s,rs} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.35}$$

$$Y_{i,b,rb} \ge Y_{i-1,b,rb} \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.36}$$

$$Z_{i,s,rs} \ge Z_{i-1,s,rs} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.37}$$

$$X_{1,s,rs} \ge X_{2,s,rs-spotsShip_s} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.38}$$

$$Y_{4,b,rb} \ge Y_{5,b,rb-spotsBeach_b} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.39}$$

$$X_{1,s,rs} \ge Z_{6,s,rs-numConns_s} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.40}$$

$$X_{2,s,rs} \ge X_{2,s,rs-1} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.41}$$

$$Z_{6,s,rs} \ge Z_{6,s,rs-1} \qquad \forall s \in S, \forall rs \in Rx_s \tag{B.42}$$

$$Y_{4,b,rb} \ge Y_{4,b,rb-1} \qquad \forall b \in B, \forall rb \in Ry_b \tag{B.43}$$

$$Y_{i,b,rb} = \sum_{s\in S}\sum_{rs\in Rx_s} a_{s,rs,b,rb} * X_{i,s,rs} \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.44}$$

$$Z_{i,s,rs} = \sum_{\in B}\sum_{rb\in Ry_b} c_{b,rb,s,rs} * Y_{i,b,rb} \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.45}$$

$$X_{i,s,rs} \ge 0 \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.46}$$

$$Y_{i,b,rb} \ge 0 \qquad \forall i \in I, \forall b \in B, \forall rb \in Ry_b \tag{B.47}$$

$$Z_{i,s,rs} \ge 0 \qquad \forall i \in I, \forall s \in S, \forall rs \in Rx_s \tag{B.48}$$

$$welldeckStart_s \ge 0 \qquad \forall s \in S \tag{B.49}$$

$$welldeckEnd_s \ge 0 \qquad \forall s \in S \tag{B.50}$$

APPENDIX C:
Scenario Builder

We provide a detailed overview of the inputs and construction of the pre-scenarios utilized in Chapter 4 and Chapter 5. We consulted with the Marine Corps Expeditionary Energy Office (E2O) to ensure the inputs accurately reflect real world conditions.

C.1 Connector Information Inputs

We begin by randomly selecting the quantity of connectors and allocating the connectors across the various MEU ship types. We utilize the information contained in Section A.1 for each ship type to ensure the quantity of connectors allocated across each ship appropriately matches the number of connectors the ship can carry. For example, the LHD is only capable of carrying three LCACs; therefore, we ensure the the Scenario Builder script does not exceed this amount. Moreover, we ensure we are meeting the typical number of connectors available for use on a MEU as specified in the table below. Finally, we specify the bulk fuel carrying capacity and velocity for each of our connector types. Our inputs for each connector's information appears in Table C.1.

Connector Information Inputs			
Connector Type	Maximum Number Available	Capacity Options (gal)	Velocity (kts)
LCAC	5	900, 1800, 2700, 3600, 4500, 5400	30
LCU	3	900, 1800, 2700, 3600, 4500, 5400	10
CH-53	4	800, 1600, 2400	120
MV-22	5	800, 1600, 2400	200

Table C.1. Connector Information Inputs

We keep the velocity consistent across all connectors, but vary the quantities of connectors and their capacities across each scenario. The selection of the different capacities and connector quantities are based on a uniform random variable to ensure the inputs are varied across each scenario.

C.2 Network Construction

We construct our network by first determining the number of beaches, LZs, and FOBs for a given scenario. These values are obtained by utilizing a uniform random variable with the specified parameters depicted in Table C.2 below. Next, we designate the distances between each individual land node and the seabase. Specifically, we assign beaches closer to the seabaes, and LZs and FOBs further from the seabase in order to build depth inside the network. The distance inputs are represented by a uniform random variable with minimum and maximum values as depicted in the table. Next, we assign the specific land edges across the land network. We ensure all FOBs contain a land connection to at least one land node, and we randomly assign all other edges by setting the quantity of land connections equal to a random number between one and twice the number of land nodes. We then randomly assign land connections until we achieve the quantity of land connections calculated. Finally, we assign distances inside of the land network by choosing a random variable between 1 and 50 nautical miles (nm). Inputs for the network appear in Table C.2.

Network Inputs		
Land Node Type	Quantity (Minimum, Maximum)	Distance from Seabase (nm) (Minimum, Maximum)
Beach (B)	(1, 4)	(2, 12)
Landing Zone (LZ)	(0, 4)	(5, 30)
Forward-Operating Base (FOB)	(0, 3)	(5, 30)

Table C.2. Network Inputs

C.3 Land Node Information

With the connector information selected and the network constructed, we now assign demands, capacities, and connector spots for each land node. For each land node, the demand and capacity is selected from the available values depicted in the Table C.3. We then choose a uniform random variable between one and six and multiply the demand and capacity against this quantity. This method ensures we achieve a multitude of various demands and capacities across all the land nodes that closely mirror real world situations.

Land Node Demand and Capacity Inputs		
Demand (gal)	Capacity (gal)	Multiplied Value (Minimum, Maximum)
1000, 2000, 3000, 4000, 5000	3000, 4000, 5000, 6000	(1, 6)

Table C.3. Land Node Demand and Capacity Inputs

Our final step is to designate connector spots for each of the land nodes. For each land node, we categorize the different spots available to be assigned to each land node and randomly select from the categories based on a uniform random variable. We then randomly choose the quantity of spots for each land node based on the category chosen. Additionally, we vary the quantity of spots per category based on a uniform random variable with a minimum value of zero and a maximum value as depicted per category in Table C.3. We define the spot categories based on the connector categories found in Section A.2: surface, air, and land. The spot categories and the associated quantities are depicted in the Table C.4.

Land Node Spot Categories and Quantities		
Land Node	Spot Categories	Maximum Quantities per Spot Category
Beach (B)	Surface, Air, Land	6, 2, 2
Landing Zone (LZ)	Air, Land	3, 2
Forward-Operating Base (FOB)	Air, Land	2, 2

Table C.4. Land Node Spot Categories and Quantities

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

Betts N, Vasko F (2016) Solving the unbalance assignment problem: Simpler is better. *American Journal of Operations Research* (6):296–299.

Christafore R (2017) Generating ship-to-shore bulk fuel delivery schedules for the marine expeditionary unit. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA, https://calhoun.nps.edu/handle/10945/55582.

Clapper B (2017) Munkres 1.0.12: Munkres algorithm for the assignment problem. Accessed January 15, 2018, https://pypi.org/project/munkres/.

Danielson ME (2018) Scheduling amphibious connectors to deliver multiple commodities. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA, Forthcoming, https://calhoun.nps.edu/.

Department of the Navy, Headquarters United States Marine Corps (2015) *Policy for Marine Expeditionary Units (MEU) and Marine Expeditionary Units (Special Operations Capable) MEU(SOC)* (MCO 3120.9C, Washington, DC).

Department of the Navy, Headquarters United States Marine Corps (2016) *Marine Corps Operating Concept: How an Expeditionary Force Operates in the 21st Century* (MOC, Washington, DC).

Department of the Navy, United States Marine Corps (1996) *Bulk Fuel Operations* (MCWP 4-11.6, Washington, DC).

Expeditionary Warfare Training Group Pacific (2016) *Commander's Quick Reference Amphibious Warfare Handbook* (EWTGPAC, Coronado, CA).

Firehawkv8 (2007) M970b, Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:M970b.jpg.

Garcia G (2013) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:MWSS-271_Ground_Preparation_130807-M-SA716-023.jpghttps://commons.wikimedia.org/wiki/File:MWSS-271_Ground_Preparation_130807-M-SA716-023.jpg.

Hall J (2013) Accessed March 31, 2018, http://www.marcent.marines.mil/Photos/igphoto/2000027150/.

Hamacher H, Tjandra S (2001) Mathematical modelling of evacuation problems: A state of art. Technical Report 24, Institut Techno- und Wirtschaftsmathematik, Kaiserslautern, Germany.

Herendeen M (2015) Movement of fuel ashore: storage, capacity, throughput, and distribution analysis. Master's thesis, Graduate School of Business and Public Policy (GSBPP), Naval Postgraduate School, Monterey, CA. https://calhoun.nps.edu/handle/10945/47960.

Hoogendam L (2016) Accessed March 31, 2018, http://www.marines.mil/Photos/igphoto/2001592926/.

Jane's By IHS Markit (2017) Oshkosk MK31 medium tactical vehicle replacement (MTVR) medium equipment transporter (MET), Accessed March 1, 2018, http://janes.ihs.com/Janes/Display/1501756#blockImage.

Joint Chiefs of Staff (2014) Amphibious operations (JP 3-02, Washington, DC).

Kiehl CR (2011) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:USMC-120509-M-RO494-094.jpg.

Kuhn H (1955) The hungarian method for the assignment problem. *Naval Research Logistics Quarterly* 2:83–97.

National Research Council (1999) *Naval Expeditionary Logistics Enabling Operation Maneuver From the Sea* (The National Academies Press, Washington DC).

Rock AJ (2008) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:USMC-0810226-M-1152R-048.jpg.

Roman-Otero A (2003) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:US_Navy_030113-N-2972R-114_A_Landing_Craft_Air_Cushion_(LCAC)_Vehicle_from_Assault_Craft_Unit_Four_(ACU-4)_transports_Marine_Assault_Vehicles_to_Kearsarge.jpg.

Seeley A (2017) USS Ashland builds relationships with brisbane visit, Accessed March 31, 2018, http://www.c7f.navy.mil/Media/News/Display/Article/1234031/uss-ashland-builds-relationships-with-brisbane-visit/.

Skahen S, Boyett M, Brookhart M, Benner S, Kure J, Maier J (2013) Exploring the reduction of fuel consumption for ship-to-shore connectors of the marine expeditionary brigade. Master's thesis, Department of Systems Engineering, Naval Postgraduate School, Monterey, CA. https://calhoun.nps.edu/handle/10945/39005.

The Editors of Encyclopaedia Britannica (2018) Beufort scale. Accessed May 1, 2018. https://www.britannica.com/science/Beaufort-scale.

United States Marine Corps (2009) *SIXCON, Fuel Storage Tank Module*. TM 12359A-OD, Accessed March 31, 2018, http://www.marcorsyscom.marines.mil/Portals/105/pdmeps/docs/FUEL/B2085.pdf.

United States Marine Corps (2014) *Helicopter Capabilities/Operations* (B2C3197, Camp Barrett, VA).

Viado H (1999) An optimization model for sea-based supply of bulk fuel for a deployed marine expeditionary unit. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA. https://calhoun.nps.edu/handle/10945/13567.

Vinson D (2006) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:USS_San_Antonio_(LPD-17).jpg.

Ward P (2008) Optimizing ship-to-shore movement for hospital ship humanitarian assistance operations. Master's thesis, Department of Operations Research, Naval Postgraduate School, Monterey, CA. https://calhoun.nps.edu/handle/10945/4153.

Webb S (2002) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/Category:General_views_of_USS_Iwo_Jima_(LHD-7)#/media/File:US_Navy_020428-N-2179W-001_Aircraft_assigned_to_the_Iwo_Jima%27s_air_combat_element_fly_over_the_multi-purpose_amphibious_assault_ship_during_operations_in_the_Atlantic_Ocean.jpg.

Wickam H, Chang W (2016) Create elegant data visualisations using the grammar of graphics, version 2.2.1. Accessed December 12, 2017, http://ggplot2.tidyverse.org.

Yarnall D (2004) Accessed March 31, 2018, https://commons.wikimedia.org/wiki/File:US_Navy_040223-M-4806Y-043_A_Landing_Craft_Utility_(LCU)_arrives_just_offshore_to_unload_supplies_and_equipment_in_support_of_exercise_Balikatan_2004.jpg.

THIS PAGE INTENTIONALLY LEFT BLANK

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California