NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

IMPROVING OPTIMIZATION MODELS SUPPORTING USMC CONNECTOR EMPLOYMENT IN EABO THROUGH ROUTE ENUMERATION

by

Amy M. Scheumann

June 2023

Thesis Advisor:	Emily M. Craparo
Co-Advisor:	W. Matthew Carlyle
Second Reader:	Thomas W. Lucas

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE	*Form Approved OMB No. 0704-0188*

1. AGENCY USE ONLY ***(Leave blank)***	**2. REPORT DATE** June 2023	**3. REPORT TYPE AND DATES COVERED** Master's thesis

4. TITLE AND SUBTITLE IMPROVING OPTIMIZATION MODELS SUPPORTING USMC CONNECTOR EMPLOYMENT IN EABO THROUGH ROUTE ENUMERATION	**5. FUNDING NUMBERS**
6. AUTHOR(S) Amy M. Scheumann	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000	**8. PERFORMING ORGANIZATION REPORT NUMBER**
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A	**10. SPONSORING / MONITORING AGENCY REPORT NUMBER**

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.	**12b. DISTRIBUTION CODE** A

13. ABSTRACT (maximum 200 words)

The U.S. Marine Corps requires a method to analyze force closure and logistics requirements in the Expeditionary Advanced Base Operations (EABO) concept. EABO is the doctrine specifying the methodology by which the Marine Corps conducts sea denial, sea control, power projection, and fleet sustainment in support of naval power projection. Deploying and sustaining forces conducting EABO requires logistics networks consisting of surface and airborne connector platforms with a variety of capabilities. The Marine Corps currently utilizes two models to analyze these logistics networks, a heuristic model known as SMASH and an optimization model known as the Path Enumeration Mixed Integer Program (PE-MIP). The computational difficulty of PE-MIP limits its utility in large-scale applications, while the SMASH solution quality is unreliable due to its heuristic nature. This thesis employs a connector-based route enumeration reformulation known as the Path and Route Enumeration Mixed Integer Program (PRE-MIP). PRE-MIP is designed to decrease the model's computational complexity and diminish runtimes while producing solutions of high quality. We compare the runtime and solution quality of PE-MIP and PRE-MIP and find that PRE-MIP significantly improves the time required to find a feasible solution, but both models continue to struggle to close the relative optimality gap.

14. SUBJECT TERMS optimization, enumeration, logistics, EABO, Expeditionary Advanced Base Operations, Path Enumeration Mixed Integer Program, PE-MIP, Path and Route Enumeration Mixed Integer Program, PRE-MIP	**15. NUMBER OF PAGES** 65
	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.

IMPROVING OPTIMIZATION MODELS SUPPORTING USMC CONNECTOR EMPLOYMENT IN EABO THROUGH ROUTE ENUMERATION

Amy M. Scheumann
Captain, United States Army
BS, United States Military Academy, 2014

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
June 2023

Approved by: Emily M. Craparo
Advisor

W. Matthew Carlyle
Co-Advisor

Thomas W. Lucas
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The U.S. Marine Corps requires a method to analyze force closure and logistics requirements in the Expeditionary Advanced Base Operations (EABO) concept. EABO is the doctrine specifying the methodology by which the Marine Corps conducts sea denial, sea control, power projection, and fleet sustainment in support of naval power projection. Deploying and sustaining forces conducting EABO requires logistics networks consisting of surface and airborne connector platforms with a variety of capabilities. The Marine Corps currently utilizes two models to analyze these logistics networks, a heuristic model known as SMASH and an optimization model known as the Path Enumeration Mixed Integer Program (PE-MIP). The computational difficulty of PE-MIP limits its utility in large-scale applications, while the SMASH solution quality is unreliable due to its heuristic nature. This thesis employs a connector-based route enumeration reformulation known as the Path and Route Enumeration Mixed Integer Program (PRE-MIP). PRE-MIP is designed to decrease the model's computational complexity and diminish runtimes while producing solutions of high quality. We compare the runtime and solution quality of PE-MIP and PRE-MIP and find that PRE-MIP significantly improves the time required to find a feasible solution, but both models continue to struggle to close the relative optimality gap.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

1 Introduction **1**

1.1 Research Questions 2

1.2 Organization of the Thesis. 2

2 Literature Review **5**

2.1 Network Optimization in Logistics Modeling 5

2.2 Warm Starts in Optimization Modeling 7

3 Model Formulation **9**

3.1 PRE-MIP Model History 9

3.2 Route Enumeration 10

3.3 Warm Start. 12

3.4 PRE-MIP Formulation 12

3.5 PRE-MIP Input Data 17

4 Experimentation **19**

4.1 Software. 19

4.2 Two MLR Deployment Scenario 19

4.3 Experiments and Results 21

5 Conclusions and Future Work **37**

5.1 PRE-MIP Parameters. 37

5.2 PRE-MIP vs PE-MIP. 38

5.3 Future Work 38

List of References **41**

Initial Distribution List **43**

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

Figure 3.1 Example network 11

Figure 4.1 Two MLR deployment scenario network 20

Figure 4.2 Warm start experiment results 23

Figure 4.3 PRE-MIP runtime for varying MIN_PATH_ARCS values . . . 26

Figure 4.4 PRE-MIP last serial delivery times for varying MIN_PATH_ARCS values 27

Figure 4.5 PRE-MIP runtime for varying WS_PATH_NODES values . . . 28

Figure 4.6 PRE-MIP last serial delivery times for varying WS_PATH_NODES values 29

Figure 4.7 Relationship between warm start and main model runtimes 30

Figure 4.8 PRE-MIP and PE-MIP objective function values over time 35

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

Table 3.1 Summary of model input data. 18

Table 4.1 Model parameters with definitions and default values. 21

Table 4.2 Warm start experiment design non-default parameter values. 22

Table 4.3 Warm start experiment results. 24

Table 4.4 Route filtering experiment design non-default parameter values. . . 25

Table 4.5 Route filtering experiment results for all design points with relative gap termination condition. 31

Table 4.6 Route filtering experiment results for all design points with absolute gap termination condition. 32

Table 4.7 PRE-MIP versus PE-MIP experiment design non-default parameter values. 33

Table 4.8 PE-MIP vs PRE-MIP experiment results. 34

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

C2ISR	command, control, intelligence, surveillance, and reconnaissance
DP	design point
EABO	Expeditionary Advanced Base Operations
INFORMS	Institute for Operations Research and the Management Sciences
MACS	Marine Expeditionary Unit (MEU) Amphibious Connector Scheduler
MEF	Marine Expeditionary Force
MEU	Marine Expeditionary Unit
MLR	Marine Littoral Regiment
NPS	Naval Postgraduate School
OAD	Operations Analysis Directorate
OR	operations research
ORION	On-Road Integrated Optimization and Navigation
PE-MIP	Path Enumeration Mixed Integer Program
PRE-MIP	Path and Route Enumeration Mixed Integer Program
SMASH	Strategic Marine Analytical Solving Heuristic
S-MIP	Schedule Mixed Integer Program
UPS	United Parcel Service
VRP	vehicle routing problem

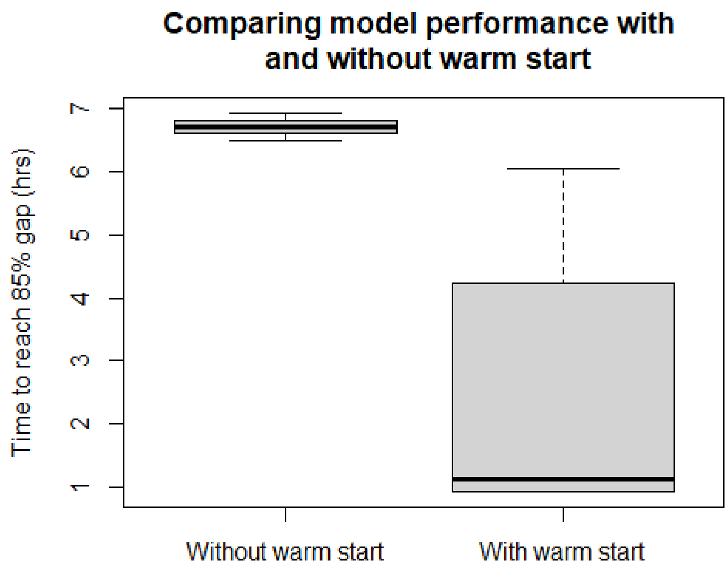
THIS PAGE INTENTIONALLY LEFT BLANK

Executive Summary

The Marine Corps planning concept for conflicts with near-peer adversaries in the Western Pacific, particularly the South China Sea, is called Expeditionary Advanced Base Operations (EABO). EABO focuses on littoral operations in contested environments and requires rapid force closure. Marine Corps planners at Operations Analysis Directorate (OAD), Marine Corps Pacific, and III Marine Expeditionary Force (MEF) intently analyze how the Marine Corps can improve through force structure and acquisition decisions, in addition to conducting wargames to analyze various concepts of operation. These planners require a tool to assist them in analyzing force closure times and EABO sustainment. OAD currently has an optimization model named Path Enumeration Mixed Integer Program (PE-MIP) to model force closure and sustainment. While PE-MIP returns solutions of known high quality, its runtime is unacceptably slow. This thesis seeks to improve PE-MIP's runtime without sacrificing solution quality. This thesis contributes to this research problem by seeking answers to the following questions:

1. Does column generation via route enumeration and route filtering result in a faster solution time without sacrificing solution quality?
2. Does employing a warm start result in a faster solution without sacrificing solution quality?

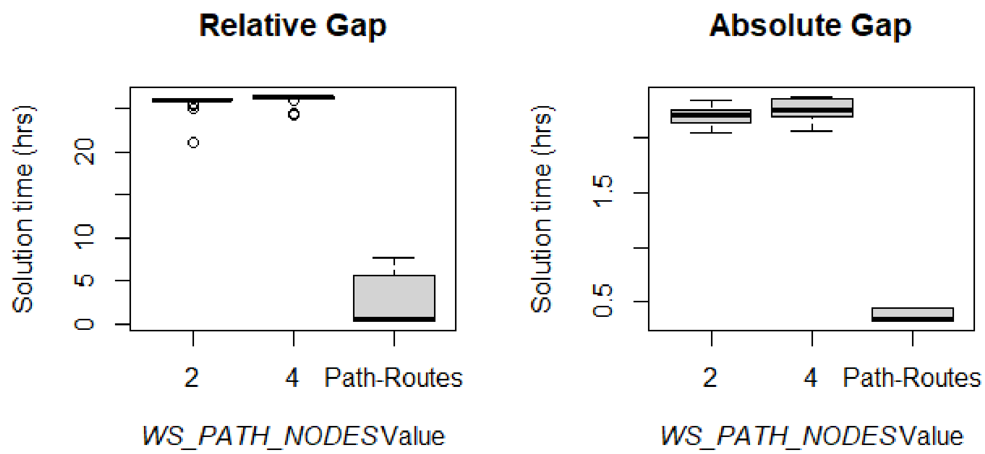
This research enumerates and filters the connector routes and uses column generation to add them to the model. This results in a reformulation of the model, known as Path and Route Enumeration Mixed Integer Program (PRE-MIP). Additionally, this thesis develops a warm start procedure for PRE-MIP where routes consist of exact copies of the serial paths generated by the path generation and filtering algorithm. We describe multiple experiments on a large-scale instance known as the Two Marine Littoral Regiment (MLR) Problem that represents the use-case of deploying two MLRs from their current bases to initial operating positions using current and near-future connectors. The experiments evaluate the impact of employing various methodologies and model parameter values on the solution quality and solution time.



Warm start experiment results grouped by warm start parameter value. Notice that the "With warm start" solution times not do overlap at all with the "Without warm start" solution times, indicating that employing the warm start is always the best decision.

Our results demonstrate that employing a warm start provides significant benefit to PRE-MIP. Additional experimentation further reveals that the WS_PATH_NODES parameter, which controls the size of the trimmed warm start network, works best when WS_PATH_NODES = Path-Routes, indicating that the only routes generated are exact copies of the serial paths.

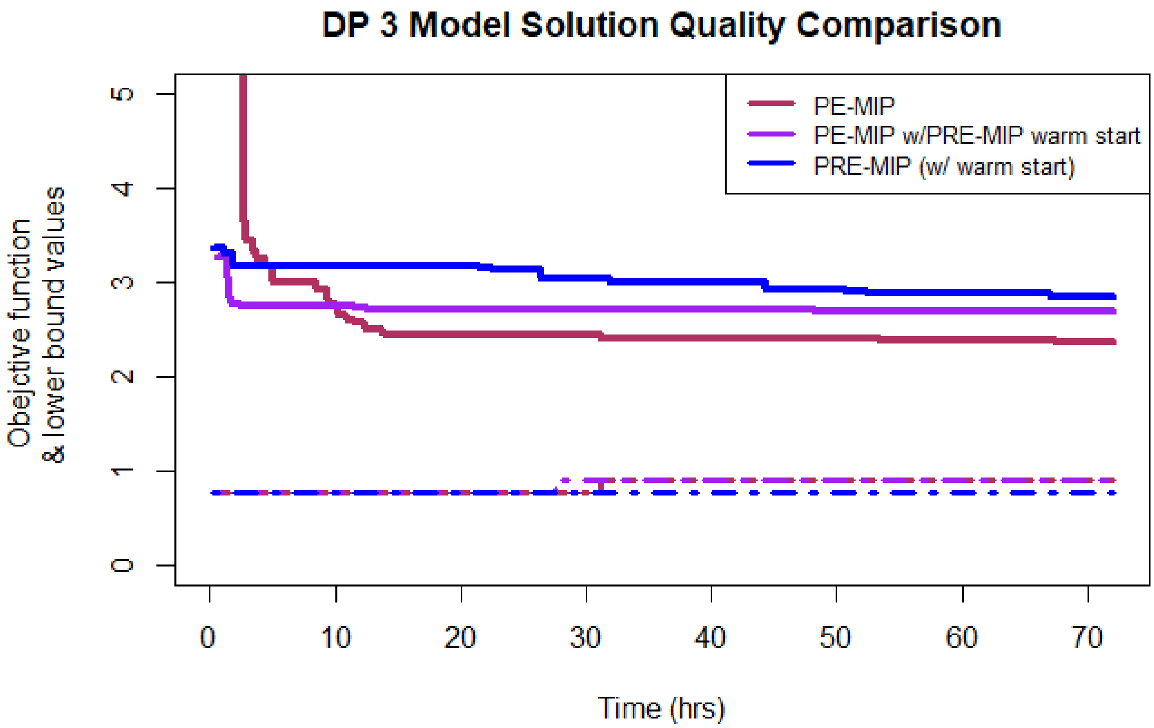
An experiment designed to tune the parameter MIN_PATH_ARCS, which controls the intensity of route filtering by setting a minimum limit on the number of path arcs which a route must contain, does not yield conclusive results regarding the optimal value of MIN_PATH_ARCS. In the absence of an empirically supported default value, the research team opts to keep the filter and selects $MIN\_PATH\_ARCS = 1$ as the default value because a route that does not include any serial path arcs is useless.



Comparison of runtime required to achieve relative gap termination condition for design points with varying values for WS_PATH_NODES.

Finally, this thesis compares the performance of PE-MIP and PRE-MIP using the parameters found during this and previous research to work best with each model. In addition to PE-MIP and PRE-MIP, we evaluate PE-MIP using the PRE-MIP warm start. The experiment includes three design points (DPs): one focused on quickly returning a solution that delivers the maximum number of deliverable serials (DP 1), one that increases select model parameters to create larger and more complex instances of the models (DP 2), and one focused on returning the best solution possible under the same parameters as DP 1 except the termination condition (DP 3). Both models employing the warm start complete DP 1 in less than half the time as PE-MIP while attaining a significantly better objective value. In the large instance, DP 2, no PRE-MIP configuration returns a feasible solution in the available runtime, while PE-MIP returns a feasible solution in about 20 hours.

In DP 3, PE-MIP returns the best solution quality in the 72 hour runtime, however every model demonstrates very little movement in objective function value after about 24 hours runtime. Both models using the warm start achieve an objective function value within 0.5 of PE-MIP's best objective function value in only 1.5 hours. The PRE-MIP warm start improves the model's ability to return a "fast and dirty" solution in both solution quality



DP 3 objective function value and lower bound comparison between PE-MIP, PE-MIP with PRE-MIP warm start, and PRE-MIP over 72 hours. Solid lines represent objective function values while dotted lines represent lower bound values. Notice that the lower bounds move very little, PE-MIP achieves the best objective function value, and PRE-MIP attains a feasible solution much faster than PE-MIP.

and solution speed. This compromise between solution speed and solution quality may be attractive to the model users.

The research team observed during development that many of the routes in PRE-MIP are functionally equivalent, in that many routes may be assigned to a connector without changing the objective function value. We theorize that symmetry in the model may be causing each model's struggle to close the relative gap. Future work may explore adding a tie-breaker to the objective function in order to reduce this symmetry. We propose that a tie-breaker which accounts for the last serial delivery time or rewards not using a connector at all may help reduce symmetry and improve the speed at which the model closes the relative optimality gap.

Acknowledgments

First and foremost, I am sincerely grateful for the patient guidance, support, and expertise of Dr. Craparo and Dr. Carlyle throughout this journey. I especially appreciate the time you dedicated to helping me grasp the more challenging concepts, even when it required multiple attempts, and graciously entertaining my latest ideas each week. Working with you has been a pleasure. To my classmates and friends, thanks for making the past two years an almost-always enjoyable experience. To my family, I am deeply thankful for your lifetime supply of love and support. Finally, thanks to my cats, Kona and Chubbs, for knowing the best place to lie down is on my keyboard while I'm using it.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1: Introduction

The Marine Corps planning concept for conflicts with near-peer adversaries in the Western Pacific, particularly the South China Sea, is called Expeditionary Advanced Base Operations (EABO). EABO focuses on littoral operations in contested environments, and is characterized by "employment of mobile, low-signature, operationally relevant, and relatively easy to maintain and sustain naval expeditionary forces from a series of austere, temporary locations ashore or inshore within a contested or potentially contested maritime area" (Headquarters, U.S. Marine Corps. 2021). A Marine littoral force conducting EABO is a key enabler to a naval campaign, supporting the Navy by conducting littoral sea denial operations, forward command, control, intelligence, surveillance, and reconnaissance (C2ISR), and supporting sea control operations, among other missions (Headquarters, U.S. Marine Corps. 2021). EABO requires rapid insertion of forces and equipment from their bases to initial operating positions (force closure), and agile logistics networks to support troops conducting EABO (Headquarters, U.S. Marine Corps. 2021). Marine Corps planners at Operations Analysis Directorate (OAD), Marine Corps Pacific, and III Marine Expeditionary Force (MEF) are intently analyzing how the Marine Corps can improve through force structure and acquisition decisions, in addition to conducting wargames to analyze various concepts of operation. These planners require a tool to assist them in analyzing force closure times and EABO sustainment using a variety of platforms, force structures, and plans.

The current tools used to solve this problem are manual labor by planners, the Strategic Marine Analytical Solving Heuristic (SMASH) heuristic algorithms, and the Path Enumeration Mixed Integer Program (PE-MIP) optimizer. Manually planning force closure and EABO sustainment requires deliberate planning by trained planners, but is often eschewed in favor of applying traditional rules of thumb with little basis in the current problem. SMASH is designed to find approximate solutions quickly when conventional methods are inefficient or fail to find an exact solution. Although the SMASH heuristic returns results in a satisfactory runtime, the quality of its solutions is unknown due to its heuristic nature and therefore requires further vetting before it can be used to inform important decisions (Roofner 2022).

The PE-MIP optimization tool returns solutions of known high quality, however its runtime is unacceptably slow, limiting the Marine Corps' ability to usefully apply the tool. This thesis seeks to improve the PE-MIP tool's runtime without sacrificing solution quality in order to provide Marine Corps planners with a useful tool with which to analyze EABO force closure and sustainment.

1.1 Research Questions

This thesis contributes to this research problem by seeking answers to the following questions:

1. Does column generation via route enumeration and route filtering result in a faster solution time without sacrificing solution quality?
2. Does employing a warm start result in a faster solution without sacrificing solution quality?

To answer the research questions, this thesis describes a new formulation incorporating enumerated connector routes and a warm start based on a trimmed network. Chapter 4 describes a series of experiments designed to determine how the new model formulation, Path and Route Enumeration Mixed Integer Program (PRE-MIP), compares to the old version, PE-MIP, and to determine those model parameter values that help the model solve fastest. All experimentation was conducted on a scenario of deploying two Marine Littoral Regiments (MLRs) from their home stations to initial operating positions. This scenario is explained in further detail in Chapter 4.

This research will provide OAD with a tool they can use to model force closure in operational planning, analysis of alternatives, and wargaming. This research also contributes to the body of work exploring how to expedite computationally complex mixed integer problem solution time while maintaining solution quality.

1.2 Organization of the Thesis

This thesis began with an overview of how this research problem supports the Marine Corps' future force design plans within EABO. Chapter 2 continues with a literature review exploring the history of military logistics optimization models, use of network theory in transportation planning, use of decision space enumeration to decrease solve times, and use of warm starts to decrease solve times. Chapter 3 sets forth the model formulation along with an explanation of changes from the previous model formulation and the logic for the changes. Chapter 4 describes the experiments conducted to explore the utility of the model changes and analyzes the experiment results. Finally, Chapter 5 lays forth insights gained from this research as well as proposed future work.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 2:
Literature Review

Delivering military forces and their equipment to potentially contested initial operating positions is a non-trivial problem to solve due to its complexity and the ever-fluctuating operational environment. However, there is a strong tradition in industry and the military of using optimization models based in network theory to plan logistics in such a way to optimize for chosen measurement, for example time to deliver all supplies from their origins to their destinations, using the available equipment and modeling any applicable physical constraints. We now provide an overview of selected works that utilize network optimization in logistics modeling, as well as a brief discussion of the use of warm starts in optimization modeling.

2.1 Network Optimization in Logistics Modeling

Leonhard Euler pioneered network theory in his 1736 paper "The Seven Bridges of Königsberg" (Estrada and Knight 2015). Applications of network theory to logistics problems progressed through the industrial age as analysts applied network theory to vehicular and railroad transportation as well as telecommunication networks. These applications drove further insights into network theory. In 1930, A.N. Tolsoi discovered that an optimal transportation problem solution's residual graph has no negative cost cycles while analyzing Soviet rail networks (Schrijver 2002). RAND scientists created the maximum flow, minimum capacity cut problem (Ford and Fulkerson 1956) and employed the problem to analyze Soviet railway interdiction in "Fundamentals of a Method for Evaluation of Rail Net Capacities" (Harris and Ross 1955). Chapter 3 includes a basic description of network theory as applied to the EABO problem.

The vehicle routing problem (VRP) established by Dantzig and Ramser in 1959 employs network theory to solve logistics optimization problems (Laporte 1992). The VRP takes a distributor's customer demand, vehicle fleet, and network with arc costs and employs a mixed integer program to return a solution which meets the customer demand while minimizing transportation costs. Due to the NP-hard complexity of the VRP, large-scale applications often employ both exact and heuristic algorithms to produce solutions (Liong et al. 2008). One such example is the United Parcel Service (UPS) On-Road Integrated Optimization and Navigation (ORION) driver route planning program, which won the 2016 Institute for Operations Research and the Management Sciences (INFORMS) Franz Edelman Award (Holland et al. 2017). As Holland describes, ORION demonstrates the utility of using network theory and variations on the VRP in large-scale instances to design optimal delivery driver routes within a constrained time-solving window and many side constraints. Holland also notes that in addition to a variation on the VRP, ORION employs clustering techniques on customer locations and heuristic algorithms to balance priorities in route length, cost, and customer demand.

Desrochers et al. (1992) use column generation to solve the linear program relaxation of the set partitioning formulation of a VRP variation, which generates an efficient lower bound supporting the successful optimal solution of a problem six times larger than any other reported at that time. Column generation adds additional decision variables to the problem formulation and ties these new variables to previously existing ones using constraints. As demonstrated by the authors, this technique can reduce the problem size and improve the solver's efficiency, especially for problems with sparse decision variable matrices.

Brown et al. (2013) employ the above concepts to great value in the Air Tasking and Efficiency Model, which assists airlift planners in maximizing flow through the command's airlift network. The authors represent the problem as a variation on the VRP, using air landing strip locations, possible flight routes, and data regarding the airplanes and cargo to maximize daily flow through the network. The authors enumerate all the possible airplane routes and add the routes to the model formulation using column generation. The model proved successful, helping the command improve its airlift operation efficiency such that the C-130 fleet was reduced by ten aircraft while the amount of cargo moved increased (Brown et al. 2013).

An additional application includes the Marine Expeditionary Unit (MEU) Amphibious Connector Scheduler (MACS), designed by Christafore (2017) to plan amphibious operation bulk fuel ship to shore distribution. The model meets fuel demands from shore nodes using available connectors through a combination of two linear programs and an heuristic assignment algorithm; one linear program satisfies each nodes' fuel demands, the heuristic algorithm assigns specific connectors a certain amount of fuel, and the second linear program schedules connector arrivals and departures. Major Christafore finds that that although the problem could be addressed using one large linear program, introducing the heuristic algorithm as a warm start solution to the second mixed integer program reduces the problem's computational complexity to a tractable level.

2.2 Warm Starts in Optimization Modeling

It is well known that employing a warm start can improve a mixed integer problem's solution time. A minimization mixed integer program solution consists of two phases; phase one consists of solving the relaxed linear program to arrive at a feasible solution or determine that the problem is infeasible, while phase two consists of closing the gap between the upper and lower solution bounds to an acceptable amount. In a warm start, after the model is created but before phase one begins, some or all of the decision variables are given specified values. The warm start may be a feasible solution, completing phase one, or may not. The warm start may be generated by many methods, including heuristic algorithms not associated with the mixed integer program, a smaller instance of the mixed integer program, or a manual setting by subject matter experts. As stated in the book *An Optimization Primer*, warm starts can be particularly beneficial to finding a phase one solution when an approximation of the problem can be solved much faster (Royset and Wets 2022). The utility of a warm start depends on the time to arrive at and instantiate the warm start solution into the main model being less than the time the main model takes to arrive at an equivalent solution.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3: Model Formulation

This chapter describes the route-enumeration-based mixed integer program, PRE-MIP, including its inputs, outputs, and relationship with the prior model, PE-MIP. Of note, all components of PE-MIP are contained in PRE-MIP. Thus, this chapter focuses on the additional components introduced in PRE-MIP.

3.1 PRE-MIP Model History

In 2019, Lieutenant Colonel Nick Freeman developed the Schedule Mixed Integer Program (S-MIP) to assist Marine Corps planners with force closure and EABO sustainment planning by producing feasible connector and serial schedules that minimize serial delivery time (Mirsch 2022). S-MIP draws on network theory and the tradition of representing logistics distribution channels by a directed network. In a directed network, a node represents each physical location, arcs represent the connections between nodes, and each node and arc has a set of numerical values associated with it. These numerical values represent attributes such as cost, time to travel, capacity, supply, and demand (Estrada and Knight 2015). Directed networks easily apply to real-world logistics channels in any domain. Nodes represent hubs, whether it be depots, seaports, airports, or some other logistics hub. Arcs represent the method of getting from one node to another, whether it be road segments, a sea route, or an air flight plan. As in the real world, not all nodes are connected to each other by arcs. In S-MIP, serials are sets of personnel and equipment that must travel together through the network from their source node to initial operating position, while connectors are the surface vessels and aircraft that transport serials through the network. S-MIP allows the user to manually input serials or, if provided a list of all personnel and equipment with their origins and destinations, employs a heuristic algorithm to package serials. The network's nodes consist of the origins, destinations, and intermediary hubs while connectors carry these serials through the network. The specific sequence of arcs traveled by a serial is called a path, and the specific sequence of arcs traveled by a connector is called a route.

In 2021, Captain Forest Sentinella explored the value of employing a rolling horizon solve methodology, versus the original single solve methodology, in decreasing S-MIP's solution time and improving S-MIP's solution quality (Sentinella 2021). She found that the rolling horizon significantly improved S-MIP's run time, but produced lower quality solutions.

In 2022, Major Andrew Mirsch developed the PE-MIP, a reformulation of S-MIP drawing on the concept of column generation and decision space enumeration to try to decrease the model's runtime without sacrificing solution quality (Mirsch 2022). Major Mirsch uses breadth-first-search to enumerate all possible serial paths through the network from their source to destination nodes, then employs three successive filters to limit the decision space down to a user-specified number of possible paths per serial. This algorithm and subsequent analysis result in a reformulation of the model. This restriction of paths limits the model to solutions generated by the path enumeration algorithm, reducing the optimization problem's computational difficulty. Experimentation by Major Mirsch reveals that while PE-MIP does return solutions faster than the previous model, the solution times are still slower than desired. The research team posits that further use of column generation, namely enumeration of the connector routes, may further simplify the branch and bound process for the model, resulting in even faster solution times.

3.2 Route Enumeration

The task of enumerating paths is trivial compared to the task of enumerating routes. Unlike serials, connectors do not have specified origin or destination nodes and may have routes consisting of any combination of connecting arcs. Serials may be restricted to paths (sequences of arcs with no repeated nodes), however connectors must be allowed to travel walks (sequences of arcs that may contain repeated arcs) since connectors may carry multiple loads of serials between nodes. This complexity led the research team to employ stack-based depth-first search enumeration. Depth-first search is a method of enumerating all the routes from an origin to a destination node which prioritizes moving to the node up next in the most recently visited nodes' adjacency dictionary, as opposed to breadth-first search, which prioritizes exhausting the current node's adjacency dictionary prior to moving on. Depth-first search is well suited to this problem due to the network's directed cycles and the goal of moving serials through the network quickly. Due to the presence of directed cycles, the research team also specifies that all routes must be less than or equal to twice the longest path, as defined by the number of arcs in the path, to avoid an infinite loop. Stacked-based code, with its last-in-first-out nature, applies well to the depth-first search algorithm and is computationally more efficient than using dictionaries.

Figure 3.1 depicts the relationship between serial paths and connector routes on a small example network. In this example one serial and its path are shown in red. The serial is transported from its start node, A, to its destination node, C, by two different connectors with separate routes. Connector v_1 departs node A at time t_0, carrying serial s_1. At time t_1, connector v_1 and serial s_1 arrive at node B and serial s_1 disembarks. After unloading, connector v_1 continues on its route back to node A. At time t'_0, connector v_2 also departs node E and at time t'_1 arrives at node B. Due to the model's continuous-time nature, times t_0 and t'_0 and t_1 and t'_1 may or may not be equal. After a layover period to account for unloading and loading cargo, connector v_2 departs node B carrying serial s_1 at time t_2. At time t_3, connector v_2 and serial s_1 arrive at node C. Serial s_1 disembarks from connector v_2 and self-transports to its initial operating position, arriving at time t_4. At this point serial s_1's journey is complete and its contribution to the objective function is reflected by the time at which it arrives at its initial operating position, t_4.

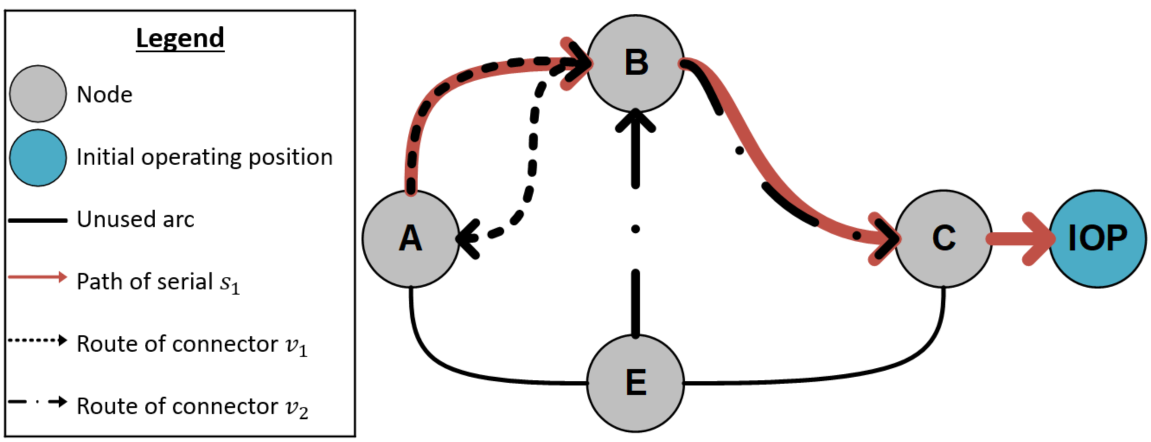


Figure 3.1. Example network depicting relationship between paths and routes.

In addition to enumerating the connector routes, we employ filtering to further limit the model's decision space and simplify the branch and bound process. The filter removes routes that do not contain a user-specified number of path arcs. Increasing the number of path arcs with which a route must overlap filters out more routes.

3.3 Warm Start

The research team identified during PRE-MIP development that the model struggled to find an initial feasible solution. In order to improve the time to find an initial feasible solution, we implement a user-specified parameter that allows the model to employ a warm start. In this case, we provide the model with routes which are direct copies of the paths and routes generated using route enumeration on a trimmed network, solve the warm start model, save the warm start solution, run the route enumeration and route filtering algorithm on the full network, instantiate the main model, specify the model to start with the warm start solution decision variable values, then solve the main model.

3.4 PRE-MIP Formulation

The route enumeration formulation adds a number of sets and indices, decision variables, and constraints not present in PE-MIP. Although route enumeration increases the total number of decision variables in the model, the additional constraints ensure that the decision space of feasible solutions is smaller.

The complete formulation of PRE-MIP follows. Red text indicates elements of the formulation which are not present in PE-MIP. In each solution, serials are assigned to a specific path while connectors are assigned to a specific route. The output values of greatest interest to the user are: $PATH_{s,p}$, which indicates the path each serial takes; $ROUTE_{v,r}$, which indicates the route each connector takes, and Z_s^{obj}, which indicates the delivery time for serial s, for those serials that are delivered.

3.4.1 Sets and Indices

$v \in CXRS$	connectors
$s \in EQUIP$	serials
$p \in PATHS$	paths for serials
$r \in ROUTES$	routes
$i, j \in NODES$	nodes/location
$(i, j) \in ARCS$	arcs
$k \in LEGS$	ordinal indices for connector leg order
$(v, k) \in CLEGS$	connector v executes at least k legs
$(v, i, j) \in TRIPS$	connector v can travel on arc (i, j)
$(v, k, i, j) \in S\_TRIPS$	connector transits
$(s, v, k, i, j) \in LOADS$	connector-serial transits
$(s, p) \in SPATHS$	serial s can travel path p
$(v, r) \in CROUTES$	connector v can travel route r
$(i, j, p) \in PARCS$	path p contains arc (i, j)
$(i, j, k) \in RARCS_r$	route r contains arc (i, j) on leg k

3.4.2 Data

h_v	usable deck area for connector v [sq ft]
v_s	value/priority weight for serial s
a_s	deck footprint size of serial s [sq ft]
$t_{v,i,j}$	time required for connector v to make transit (i, j) [days]
$LAYOVER$	time cost for connector to enter/leave nodes [days]
$maxhops_s$	maximum number of nodes visited by serial s on any path
$best\_possible\_time_{s,p}$	fastest transit time possible for serial s along path p, excluding layovers
$fastest\_time_s$	fastest transit time possible along any of the potential paths for serial s, excluding layovers
BIG	non-delivery penalty
e_s	terminal node for serial s

3.4.3 Decision Variables

$PATH_{s,p}$	= 1 if serial s is assigned to path p; else 0
$ROUTE_{v,r}$	= 1 if connector v is assigned to route r; else 0
P_s	= 1 if serial s not delivered to destination; else 0
$X_{v,k,i,j}$	= 1 if connector v makes transit (i, j) on leg k; else 0
$Y_{s,v,k,i,j}$	= 1 if serial s loaded for transit (v, k, i, j); else 0
$W_{v,k}$	time (≥ 0) at which connector v completes leg k [days]
$Z_{s,i}$	time (≥ 0) at which serial s arrives at node i [days]
Z_s^{obj}	objective penalty for serial s [days]

3.4.4 Formulation

$$\underset{W,X,Y,Z,P,PATH,R,ROUTE}{\text{minimize}} \quad z = \sum_{s \in EQUIP} v_s\left(\frac{Z_s^{obj}}{|EQUIP|} + P_s BIG\right) \tag{3.1a}$$

subject to

$$\sum_{p:(s,p)\in PATHS} PATH_{s,p} + P_s = 1 \ \forall\ s \in EQUIP, \tag{3.1b}$$

$$\sum_{r:(v,r)\in CROUTES} ROUTE_{v,r} = 1 \ \forall\ v \in CXRS, \tag{3.1c}$$

$$\sum_{(v,k)\in CLEGS} Y_{s,v,k,i,j} = \sum_{p:(i,j,p)\in PARCS} PATH_{s,p}, \tag{3.1d}$$

$\forall\ s \in EQUIP, (i,j) : \exists p : (s,p) \in SPATHS$ and $(i,j,p) \in PARCS$

$$X_{v,k,i,j} = \sum_{r:(i,j,k)\in RARCS_r,(v,r)\in CROUTES} ROUTE_{v,r} \ \forall\ (v,k,i,j), \tag{3.1e}$$

$$\sum_{s:(s,v,k,i,j)\in LOADS} a_s Y_{s,v,k,i,j} \leq h_v X_{v,k,i,j} \ \forall\ (v,k,i,j) \in S\_TRIPS, \tag{3.1f}$$

$$t_{v,i,j} X_{v,k,i,j} + W_{v,k-1} + LAYOVER \leq W_{v,k} \ \forall\ (v,k,i,j) \in S\_TRIPS : k > 1, \tag{3.1g}$$

$$W_{v,k} + (Y_{x,v,k,i,j} - 1)BIG + LAYOVER \leq Z_{s,i} \ \forall\ (s,v,k,i,j) \in LOADS, \tag{3.1h}$$

$$Z_{s,i} + t_{v,i,j} + (Y_{x,v,k,i,j} - 1)BIG + LAYOVER \leq W_{v,k} \ \forall\ (s,v,k,i,j) \in LOADS, \tag{3.1i}$$

$$Z_s^{obj} \geq Z_{s,e_s} \ \forall\ s, \tag{3.1j}$$

$$Z_s^{obj} \geq \sum_{p:(s,p)\in SPATHS} best\_possible\_time_{s,p} PATH_{s,p} \ \forall\ s, \tag{3.1k}$$

$$Z_s^{obj} \geq fastest\_time_s \ \forall\ s, \tag{3.1l}$$

$$W_{v,k}, Z_{s,i}, Z_s^{obj} \geq 0 \ \forall\ (v,k,s,i), \tag{3.1m}$$

$$X_{v,k,i,j}, Y_{s,v,k,i,j}, P_s, PATH_{s,p}, ROUTE_{v,r} \in \{0,1\} \ \forall\ (v,k,s,i,j,p) \tag{3.1n}$$

3.4.5 Discussion

The objective function (3.1a) motivates the model to deliver all possible serials by applying the BIG penalty value to any serials not delivered. Once delivery is accomplished, the model objective function motivates the model to minimize the time at which each serial is delivered.

Constraint 3.1b states that each serial s should either be assigned to one of its possible paths or marked as unassigned.

Constraint 3.1c states that each connector v must be assigned to one and only one of its possible routes.

Constraint 3.1d states for each arc (i, j) along the path to which serial s is assigned, the serial must be carried by some connector along that arc.

Constraint 3.1e states that if connector v travels arc (i, j) on leg k, then it must be assigned to a route that travels arc (i, j) on leg k.

Constraint 3.1f states that the total deck space of serials being delivered by vessel v on leg k from i to j cannot exceed h_v.

Constraint 3.1g states that the k leg-completion time of vessel v, $W_{v,k}$ is not less than the $k-1$ leg-completion time of that connector, plus the transit time of the kth leg (and associated layover).

Constraint 3.1h states that the arrival time of serial s at j, $Z_{s,j}$ is not less than the vessel v leg-transit (i, j, k) completion time $W_{v,k} + LAYOVER$, or zero for all trips (v, k, i, j) on which serial s does not arrive.

Constraint 3.1i states that the k leg-completion time of vessel v, $W_{v,k}$ is not less than the transit time $t_{v,i,j}$ plus the arrival time $Z_{s,i}$ for any cargo loaded from i to j, or zero for cargo not loaded.

Constraints 3.1j, 3.1k, and 3.1l establish valid lower bounds on the variable used to penalize delivery time in the objective function.

Constraints 3.1m and 3.1n define decision variable domains.

3.4.6 Comparing PRE-MIP and PE-MIP

PRE-MIP and PE-MIP share many features, with PRE-MIP's major departures being the route enumeration and route filtering algorithm, warm start, and the addition of sets and indices, data, decision variables, and constraints related to the route enumeration. No components of PE-MIP are eliminated, and both models share the same objective function. While PE-MIP solves by piecemeal arc-by-arc explorations for connectors, PRE-MIP consolidates these many decisions into one overarching decision per connector: which route to take. The sets $ROUTES$, $CROUTES$, and $RARCS_R$ and index r are new to PRE-MIP. PRE-MIP contains no new data. The decision variable $ROUTE_{v,r}$ is new to PRE-MIP. The constraints 3.1c and 3.1e are new to PRE-MIP.

3.5 PRE-MIP Input Data

PRE-MIP, like PE-MIP, accepts its input data from an Excel workbook. As described in detail by Mirsch (2022), the Excel workbook consists of six sheets: `WhatWhere`, `InputSerials`, `HowFar`, `HowFast`, `ConnectorData`, and `ExperimentDesign`. The `InputSerials` and `WhatWhere` sheets are mutually exclusive; users input data into `WhatWhere` to use the heuristic serial-packing algorithm or input data into `InputSerials` to manually specify the serials. Table 3.1 summarizes the input data by sheet. The `WhatWhere` and `InputSerials` sheets include information related to the equipment being transported, including the starting and ending location and area. The `HowFar` worksheet contains the directed network information, recording the distances between each node and which connectors can access each node. The `ConnectorData` worksheet contains information related to the connectors, including usable area and speed. The `HowFast` worksheet contains information regarding how much time is required to transit from a ship-to-shore site to an initial operating position. The `ExperimentDesign` worksheet contains information regarding each design point (DP), including the quantity of each connector type, various path filtering parameters, and the solver termination criteria.

Table 3.1. Summary of model input data.

`WhatWhere`	`HowFar`	`HowFast`	`ConnectorData`	`ExperimentDesign`
Site	**Loc**	**From**	**Type**	**Design Point**
Type	**Access - Connector X**	**To**	**Square Footage**	**Connector X QTY**
Lat	**Access - Connector Y**	**Time**	**Broken Stow Factor**	**Connector Y QTY**
Long	**Access - Connector Z**		**Usable Square Footage**	**Connector Z QTY**
StS Site			**Speed - Sea State 1**	**SeaState**
StS Lat			**Speed - Sea State 2**	**FASTPATH**
StS Long			**Speed - Sea State 3**	**HOPS_COEF**
IOP			**Speed - Sea State 4**	**HOPS_ADD**
Source				**USE_ABSMINGAP**
Serial				Total Cap
Desc.				
TAMCN				MAXK
Length				VARLIM
Width				OVERLAP
Weight				mip_tolerances_mipgap
Stick Value				timelimit
Sqft				SERIALSIZE
				warmstart
				WS_PATH_NODES
				MIN_PATH_ARCS

Summary of input data specified via the Excel workbook, if not using user specified input serials. Bold text indicates the information is required to run the model. Adapted from (Mirsch 2022).

CHAPTER 4: Experimentation

This chapter describes the experiments performed in support of this thesis, including the design of each experiment, the software used, the scenario, results captured, and analysis techniques. This chapter also describes the metrics used to assess model performance, solution quality and solution time. For each DP, solution quality is measured in the objective function value, and solution time is measured in the total runtime. The experiments also capture additional results of interest such as last serial delivery time, warm start solution time, and model complexity.

4.1 Software

PRE-MIP is written in Python and created using the Pyomo package. All experiments employed the academic license of IBM's ILOG CPLEX Interactive Optimizer Studio as the solver, version 12.10.0.0.

Experiments ran using ConnectorFarmer, a script created by Stephen Upton of the Naval Postgraduate School (NPS) Simulation Experiments and Experimental Design (SEED) Center (Upton 2021). ConnectorFarmer is a data farming wrapper that runs PRE-MIP and extracts and exports the relevant results and run information. ConnectorFarmer can run multiple instances of a model concurrently on different cores of the same computer, an ideal design for this research due to the computationally expensive nature of the model. All experiments ran on the same computer, effectively standardizing hardware and software across all DPs. Due to utilizing multiple cores to run the model, a small amount of variation solution time is natural but insignificant to the results.

4.2 Two MLR Deployment Scenario

The research team developed PRE-MIP on a small-scale instance designed not to mimic a real-world problem, but verify the model's code and validate the model's design by testing its features. It was changed periodically throughout development to provide insight and stress-test specific portions of the program.

For experimentation, the research team utilized the same instance employed by Major Mirsch to develop and conduct testing on PE-MIP (Mirsch 2022). As described in Major Mirsch's thesis, the research sponsor, OAD, provided the model inputs, which represent a real-world analytic use-case for the program. The inputs represent the problem of deploying the 4th and 12th MLRs from their home stations to an initial operating position. An OAD working group tasked with analyzing the deployment of two MLRs provided the information regarding connector type, quantity, and characteristics. This problem includes over 170,000 square feet of equipment and 472 serials. The problem assumes four squadrons of KC-130Js, two squadrons of MV-22Bs, 27 Light Amphibious Warships, and two high-speed surface ships for a total of 117 connectors. The network consists of 15 total nodes and 35 arcs, of which 2 are source nodes and 2 are ship-to-shore nodes. The nodes, arcs, arc lengths, and connector accessibility data are obfuscated to maintain unclassified publication and do not represent any real-world plans. Physical proximity in the network graph does not correspond to real-world proximity between nodes. Figure 4.1 shows the network graph, with one path and two routes from a model solution depicted.

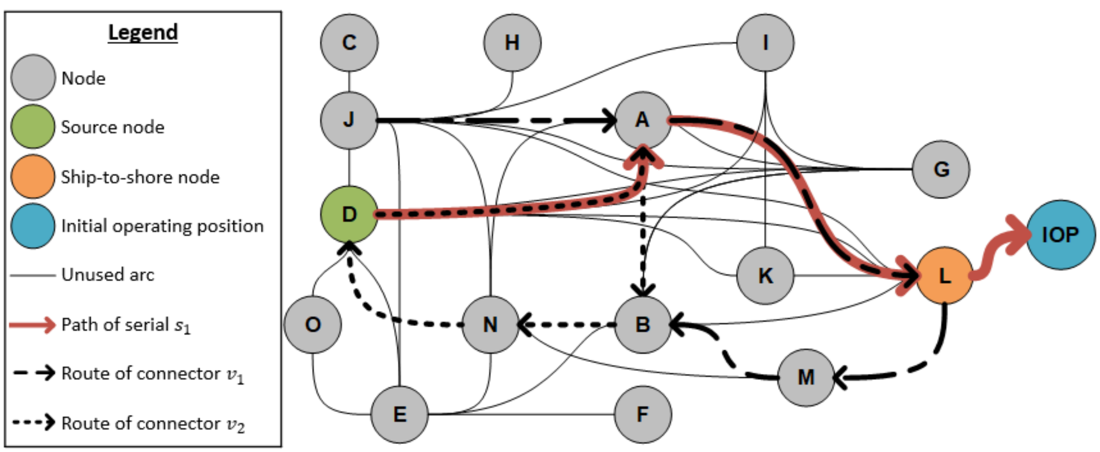


Figure 4.1. Two MLR deployment scenario network. The example serial travels the path of nodes D-A-L-IOP.

4.3 Experiments and Results

The research team conducted multiple experiments to answer the research questions. Some parameters are held constant across every experiment to facilitate quantitative comparison. Every experiment is conducted on the two MLR problem described in Section 4.2 using the same computer utilizing the same amount of resources and software. Each experiment changes only those parameters necessary to yield insight while maintaining constant values for all other parameters. Unless otherwise specified, every experiment uses the default parameters listed in Table 4.1. The default values PRE-MIP shares with PE-MIP are informed by Major Mirsch's research with different values for $MAXK$, $FASTPATHS$, $HOPS\_ADD$, $HOPS\_COEF$, and $USE\_ABSMINGAP$ (Mirsch 2022). Parameters new to PRE-MIP have default values informed by the research team's development experience.

Table 4.1. Model parameters with definitions and default values.

Parameter	**Value**	**Description**
$FASTPATHS$	1	number of paths available to each serial
$HOPS\_COEF$	1	$HOPS\_COEF$ and $HOPS\_ADD$ parameters combine to control the maximum allowable path length
$HOPS\_ADD$	0	see $HOPS\_COEF$ description
$MAXK$	5	maximum number of nodes on a path or route
$USE\_ABSMINGAP$	1	if 1, stop solver when all serials delivered than can be delivered; if 0, stop at specified relative gap
$mip\_tolerances\_mipgap$	.15	if $USE\_ABSMINGAP = 0$, specifies relative gap at which solver stops
$timelimit$	72 hrs	time at which solver stops working and reports best solution so far, if gap not already met
$warmstart$	True	whether warm start is used or not
WS_PATH_NODES	2	number of path-nodes a node must touch to be added to the trimmed warm start network
MIN_PATH_ARCS	1	number of path-nodes a route must touch

Model parameters with definitions and default values. Default values are shared by all experiments and DPs unless otherwise specified.

4.3.1 Warm Start Experiment

The first experiment explores what, if any, benefit employing a warm start provides to PRE-MIP, addressing the research question "does employing a warm start result in a faster solution without sacrificing solution quality?" The research team compares the time PRE-MIP takes to arrive at a solution of equal quality with and without the warm start, holding all other parameters equal. In this experiment comparing solution quality is superfluous because the main model is the same whether or not the warm start is used, so all DPs terminate upon reaching an 85% relative gap. The experiment includes a total of 12 design points, three without a warm start and nine with a warm start. See Table 4.2 for a complete list of the experiment's non-default parameters. The experiment varies the parameters $FASTPATHS$ and WS_PATH_NODES in order to capture model behavior under a variety of conditions.

Table 4.2. Warm start experiment design non-default parameter values.

Parameter	Values
$MAXK$	4
$USE\_ABSMINGAP$	0
$mip\_tolerances\_mipgap$	.85
$warmstart$	True, False
$FASTPATHS$	1, 2, 4
WS_PATH_NODES	2, 3, 4

Non-default parameters for each DP. Combine for a total of 12 DPs, 3 without warm start and 9 with warm start. The WS_PATH_NODES parameter only applies to DPs employing the warm start.

This experiment's results emphatically demonstrate that the warm start provides significant benefit to PRE-MIP. All three DPs not employing the warm start took over 6.5 hours to reach the termination condition, while all nine DPs employing the warm start took less than 6.5 hours. The solution times for the warm start DPs ranged from 55 minutes to 6.1 hours, with a median time of 68 minutes, resulting in solution times up to 7.3 times faster. A Mann-Whitney nonparametric test on the mean solution time with the null hypothesis that using the warm start results in solution times less than not using the warm start generates a p-value of 0.005. The warm start DP parameters and solution times are listed in Table 4.3. Figure 4.2 shows the significant difference between solution times with and without the warm start. This experiment leads the research team to conclude that the warm start methodology is a valuable addition to PRE-MIP.

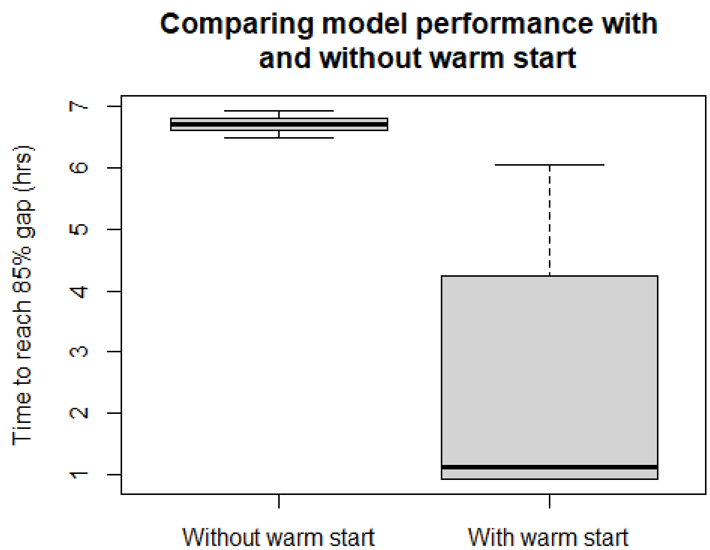


Figure 4.2. Warm start experiment results grouped by warm start parameter value. Notice that the "With warm start" solution times not do overlap at all with the "Without warm start" solution times, indicating that employing the warm start is always the best decision for this particular instance.

Table 4.3. Warm start experiment results.

FASTPATHS	WARMSTART	WS_PATH_NODES	Runtime (hrs)
1	True	2	6.06
1	True	3	0.92
1	True	4	2.90
2	True	2	4.74
2	True	3	0.92
2	True	4	0.92
4	True	2	6.06
4	True	3	0.92
4	True	4	0.92
1	False	n/a	6.71
2	False	n/a	6.70
4	False	n/a	6.73

Warm start experiment solution time in hours by DP. The DPs not employing the warm start perform the worst, while DPs employing the warm start and intermediate trimming of the warm start network (WS_PATH_NODES = 3 and 4) perform the best.

4.3.2 Route Filtering Experiment

This experiment aims to determine if employing a route filtering methodology provides benefit to PRE-MIP, in support of the research question "does column generation via route enumeration and route filtering result in a faster solution time without sacrificing solution quality?" The route filtering mechanism is the parameter MIN_PATH_ARCS, which the user specifies an integer value representing the minimum number of arcs each route must contain that coincide with the set of arcs that are on serial paths. The value for MIN_PATH_ARCS can range from 0 to $MAXK - 1$, with 0 representing the entire set of possible routes and $MAXK - 1$ representing the smallest set of routes, containing only path arcs. Similar to the warm start experiment, we record the time to arrive at a given solution quality (75% relative gap or absolute gap) when employing different levels of route filtering. Table 4.4 shows the full complement of non-default parameters. In addition to varying the value of the parameter MIN_PATH_ARCS, we also vary $FASTPATHS$ and WS_PATH_NODES in order to capture the performance of route filtering under a variety of conditions. In the model, $MIN\_PATH\_ARCS = 0$ represents no filtering. We run each DP with an absolute termination gap in order to explore the quality of a "fast-and-dirty" solution when employing the filter, though DPs with different termination criteria should not be compared to each other. In addition to entering integer values for the WS_PATH_NODES parameter, we also explore the case where the only routes provided to the warm start model are exact copies of paths; this case is reflected by WS_PATH_NODES = Path-Routes.

Table 4.4. Route filtering experiment design non-default parameter values.

Parameter	**Values**
$USE\_ABSMINGAP$	1, 0
$mip\_tolerances\_mipgap$	n/a, 75%
$MAXK$	4
$timelimit$	24 hrs
$FASTPATHS$	1, 2, 4
WS_PATH_NODES	2, 4, Path-Routes
MIN_PATH_ARCS	0, 1, 2

Non-default parameters for each DP. Combine using a full factorial design for a total of 27 DPs. Response variables of interest is the runtime required to reach the termination condition (75% relative gap or absolute gap).

The experiment results are shown in detail in Table 4.5 for DPs with a relative gap termination condition and Table 4.6 for DPs with an absolute gap termination condition.

The level of route filtering, as represented by the value of MIN_PATH_ARCS, does not appear to be associated with more desirable solution time, under both the relative and absolute gap termination conditions. Figure 4.3 shows the similar performance for all values of MIN_PATH_ARCS. Although not directly represented in the model's objective function, PRE-MIP users are also interested in the time at which the last serial arrives at its initial operating position. Figure 4.4 shows the similar performance with regard to last serial delivery time for all values of MIN_PATH_ARCS.

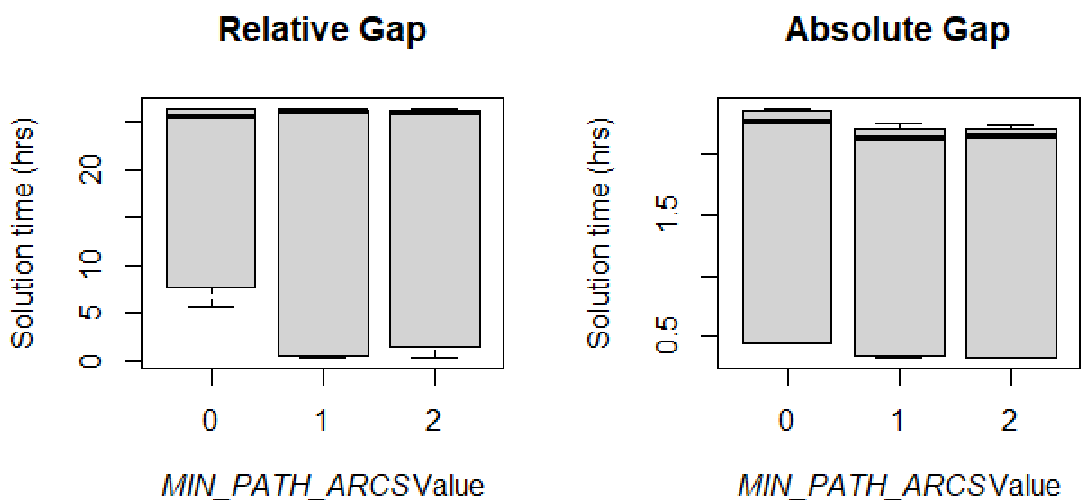


Figure 4.3. Comparison of runtime required to reach each gap termination condition for design points with varying values for MIN_PATH_ARCS.

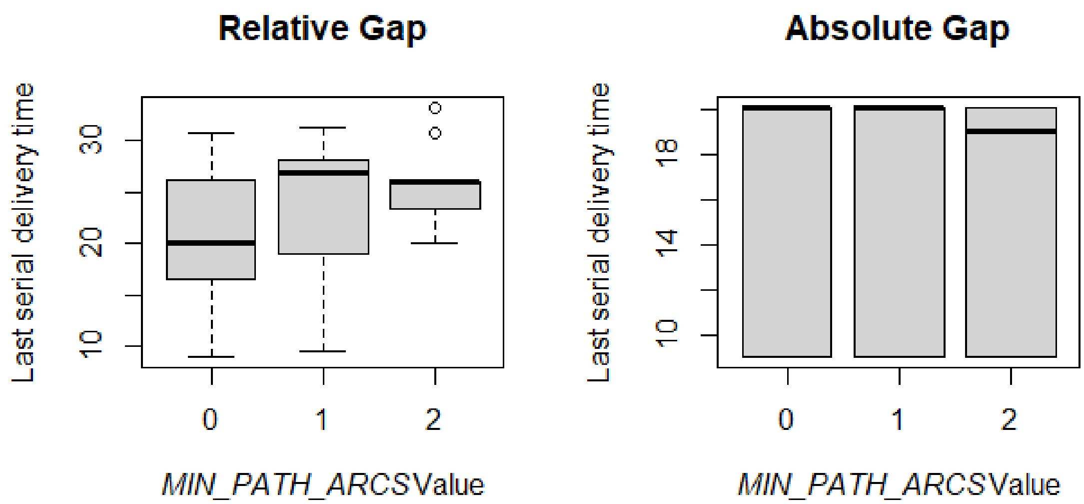


Figure 4.4. Comparison of last serial delivery times for each gap termination condition with varying values for MIN_PATH_ARCS.

The experiment yields an unexpected insight into the warm start route generation algorithm. The warm start experiment in Section 4.3.1 uses values of 0, 2, and 4 for WS_PATH_NODES, decreasing the trimmed network size with increasing values of WS_PATH_NODES. This experiment explores what happens when the warm start model is provided the most restrictive set of routes possible, routes that are merely direct copies of the paths. This case is represented by the WS_PATH_NODES value of Path-Routes. As Figure 4.5 shows, the experiment results reveal that restricting the warm start model to the smallest possible set of routes results in a much quicker solution time than any other warm start route generation methodology for both the gap termination conditions. Figure 4.6 shows that the Path-Routes warm start route generation methodology also results in significantly more desirable last serial delivery times for both gap termination conditions.

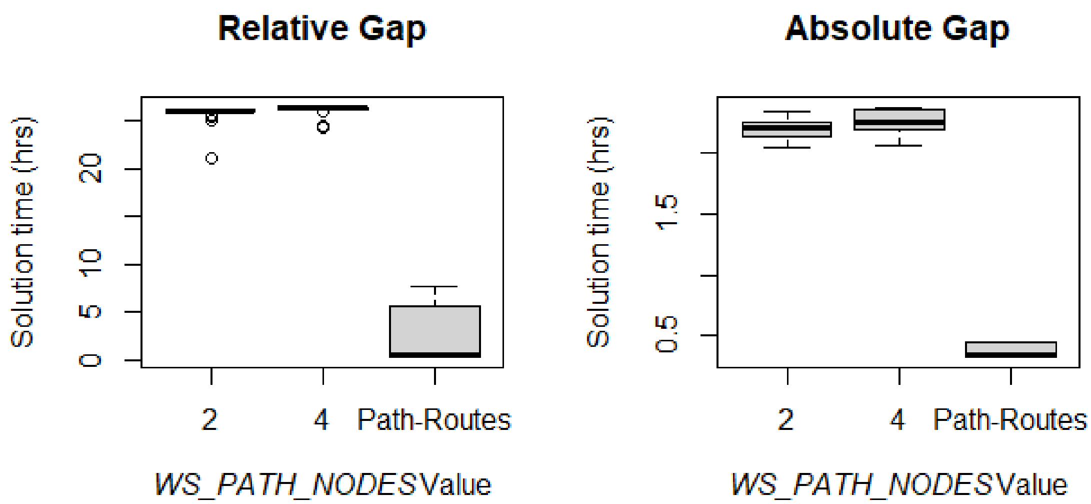


Figure 4.5. Comparison of runtime required to achieve relative gap termination condition for design points with varying values for *WS_PATH_NODES*.

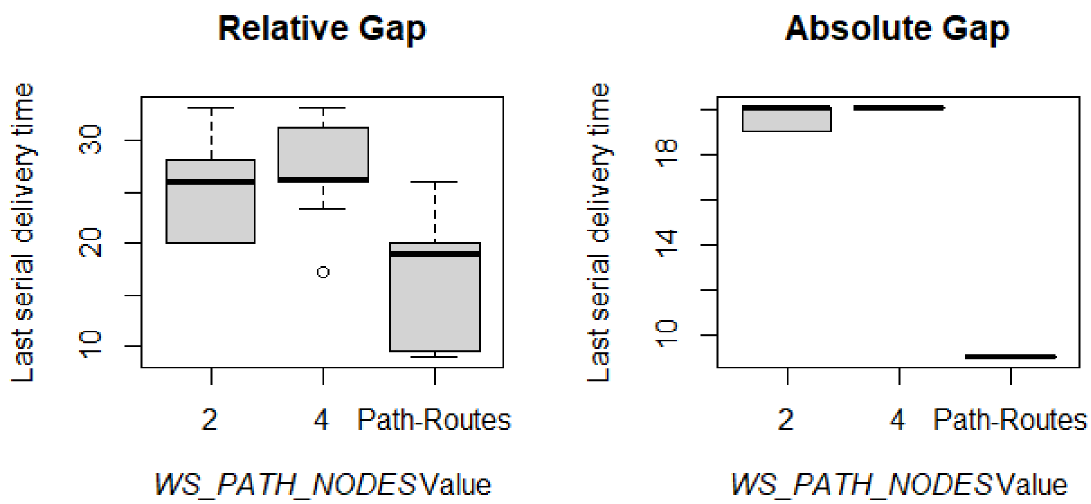


Figure 4.6. Comparison of last serial delivery times, in days, with varying values for WS_PATH_NODES. Notice that design points with Path-Routes tend to return a more desirable last serial delivery time.

The experiment results also indicate a strong relationship between the warm start runtime and the overall model runtime, regardless of the termination condition. Fast warm start runtimes are highly correlated with fast main model runtimes, with the lower 95% confidence bound on the correlation coefficient for both absolute and relative termination criteria being 0.97. Figure 4.7 shows the distinct relationship between various warm start runtimes and the main model runtime.

This experiment leads the research team to conclude that it is beneficial to run the warm start model with the WS_PATH_ARCS parameter set to only use Path-Routes. Although the value of MIN_PATH_ARCS did not appear to impact performance, the research team concludes to keep the parameter set to 1 as the default value because a route that does not include any serial path arcs is useless. Finally, the experiment suggests that improvements to the model which have desirable warm start runtimes are also advantageous to the main model, and vice versa.

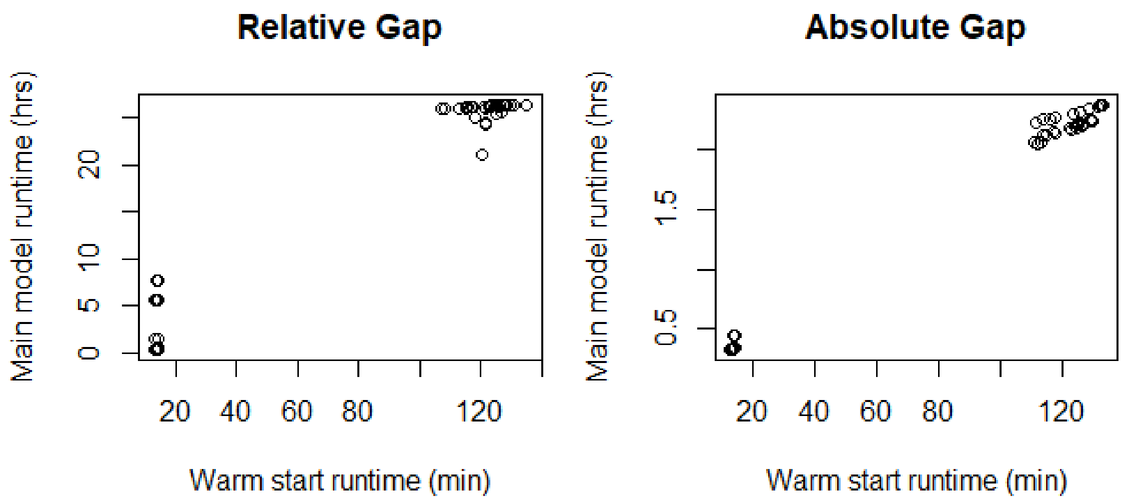


Figure 4.7. Relationship between warm start and main model runtimes for both termination conditions.

Table 4.5. Route filtering experiment results for all design points with relative gap termination condition.

MIN_ PATH_ ARCS_	WS_ PATH_ NODES	FAST-PATHS	Warm start time (min)	OBJ Value	Runtime (hrs)	Time of last delivery (days)
0	2	1	118	3.0	25	20
0	2	2	127	3.0	26	20
0	2	4	118	3.1	26	20
0	4	1	125	3.2	26	31
0	4	2	129	5.5	26	26
0	4	4	128	5.5	26	26
0	Path-Routes	1	14	3.0	5.7	17
0	Path-Routes	2	13	3.0	5.6	17
0	Path-Routes	4	14	3.0	5.7	17
1	2	1	123	3.4	26	28
1	2	2	108	3.6	26	28
1	2	4	107	3.6	26	28
1	4	1	127	3.7	26	31
1	4	2	118	3.7	26	26
1	4	4	118	4.4	26	26
1	Path-Routes	1	14	2.8	0.5	10
1	Path-Routes	2	14	3.0	0.5	19
1	Path-Routes	4	14	3.0	0.5	19
2	2	1	124	3.1	26	25
2	2	2	115	3.4	26	26
2	2	4	115	3.9	26	26
2	4	1	131	4.1	26	31
2	4	2	122	3.1	26	23
2	4	4	135	3.7	26	33
2	Path-Routes	1	14	3.0	1.6	26
2	Path-Routes	2	14	3.0	0.4	20
2	Path-Routes	4	13	3.0	1.6	26

Route filtering experiment results for all design points with relative gap termination condition.

Table 4.6. Route filtering experiment results for all design points with absolute gap termination condition.

MIN_ PATH_ ARCS_	WS_ PATH_ NODES	FAST-PATHS	Warm start time (min)	OBJ Value	Runtime (hrs)	Time of last delivery (days)
0	2	1	112	6.6	2.2	20.1
0	2	2	114	6.6	2.3	20.1
0	2	4	129	9.7	2.3	19
0	4	1	133	9.2	2.4	20.1
0	4	2	132	9.2	2.4	20.1
0	4	4	124	10.0	2.3	20.1
0	Path-Routes	1	14	3.2	0.4	9
0	Path-Routes	2	14	3.2	0.4	9
0	Path-Routes	4	14	3.2	0.4	9
1	2	1	117	6.6	2.2	20.1
1	2	2	125	9.7	2.2	19
1	2	4	124	9.7	2.2	19
1	4	1	122	10.0	2.2	20.1
1	4	2	129	9.2	2.3	20.1
1	4	4	124	10.0	2.2	20.1
1	Path-Routes	1	13	3.2	0.3	9
1	Path-Routes	2	14	3.2	0.3	9
1	Path-Routes	4	14	3.2	0.3	9
2	2	1	112	6.6	2.1	20.1
2	2	2	127	9.7	2.2	19
2	2	4	126	9.7	2.2	19
2	4	1	130	9.2	2.2	20.1
2	4	2	123	10.0	2.2	20.1
2	4	4	130	9.2	2.2	20.1
2	Path-Routes	1	14	3.2	0.3	9
2	Path-Routes	2	14	3.2	0.3	9
2	Path-Routes	4	14	3.2	0.3	9

Route filtering experiment results for all design points with absolute gap termination condition.

4.3.3 PRE-MIP vs PE-MIP Experiment

This experiment aims to determine whether PRE-MIP is indeed faster than PE-MIP, addressing the research question "does route enumeration and column generation result in a faster solution without sacrificing solution quality?"

The experiment includes three DPs. Table 4.7 shows the full complement of non-default parameters. DP 1 utilizes model parameters Major Mirsch found to work best with PE-MIP (Mirsch 2022) and focuses on quickly returning the first solution that delivers the maximum number of deliverable serials, controlled by the absolute gap termination condition. DP 2 increases select model parameters to create a larger and more complex instance of the model. DP 3 utilizes most of the same model parameters as DP 1, but focuses finding the best solution quality each model can achieve in a 72 hour runtime.

Table 4.7. PRE-MIP versus PE-MIP experiment design non-default parameter values.

Design Point	**1**	**2**	**3**
$USE\_ABSMINGAP$	1	1	0
$mip\_tolerances\_mipgap$	n/a	n/a	0.3
$FASTPATHS$	1	2	1
$HOPS\_COEF$	1	2	1
$HOPS\_ADD$	0	1	0

Non-default parameters for each DP. Each DP is run using both PE-MIP and PRE-MIP. The MIN_PATH_ARCS parameter only applies to PRE-MIP, the value 0 indicates that all routes are accepted an no route filtering takes place in this experiment.

The research team ran all three design points on multiple versions of the model: PE-MIP, to provide a benchmark for comparison; PRE-MIP, using parameters informed by the development process; and PE-MIP using a warm start provided by PRE-MIP. As a result of the warm start and route filtering experiments, PRE-MIP uses a warm start generated with WS_PATH_NODES = Path-Routes and MIN_PATH_ARCS = 1.

The experiment provides mixed results regarding the research question. Table 4.8 summarizes the results of interest for each DP and model. In DP 1, both models employing a PRE-MIP warm start return a solution twice as fast as PE-MIP, demonstrating that route enumeration significantly improves the time required to return a feasible solution. In addition to a faster runtime, the PRE-MIP warm start also returns a solution of significantly better quality than PE-MIP. In DP 2, no PRE-MIP model returns a feasible solution in the available 72 hours, while PE-MIP returns a feasible solution in about 20 hours. In DP 3, PE-MIP returns the best solution quality in the 72 hour runtime. However, all models demonstrate very little movement in objective function value after about 24 hours runtime. The lower bound on the objective value increases by no more than 0.1351 during the experiment for any model, and PRE-MIP shows no increase in the lower bound throughout the entire runtime. Figure 4.8 shows each model's objective function value and lower bound over the runtime. Notice that the PRE-MIP warm start models begin with much better solution quality, but that PE-MIP surpasses them at about the 10-hour runtime point.

Table 4.8. PE-MIP vs PRE-MIP experiment results.

DP	Metric	PE-MIP	PE-MIP w/PRE-MIP warm start	PRE-MIP
DP 1	**Time**	61 min	29 min	35 min
	OBJ	10	3.4	3.4
	Gap	92%	87%	78%
DP 2	**Time**	21 hrs	72 hrs	72 hrs
	OBJ	14	n/a	n/a
	Gap	95%	n/a	n/a
DP 3	**Time**	72 hrs	72 hrs	72 hrs
	OBJ	2.37	2.69	2.83
	Gap	62%	67%	73%

Summary of PE-MIP vs PRE-MIP experiment results for each DP. Notice that both models employing the PRE-MIP warm start finish DP 1 in about half the time as PE-MIP, but PE-MIP finds a feasible solution for DP 2 while both models employing the PRE-MIP warm start fail to find a feasible solution.

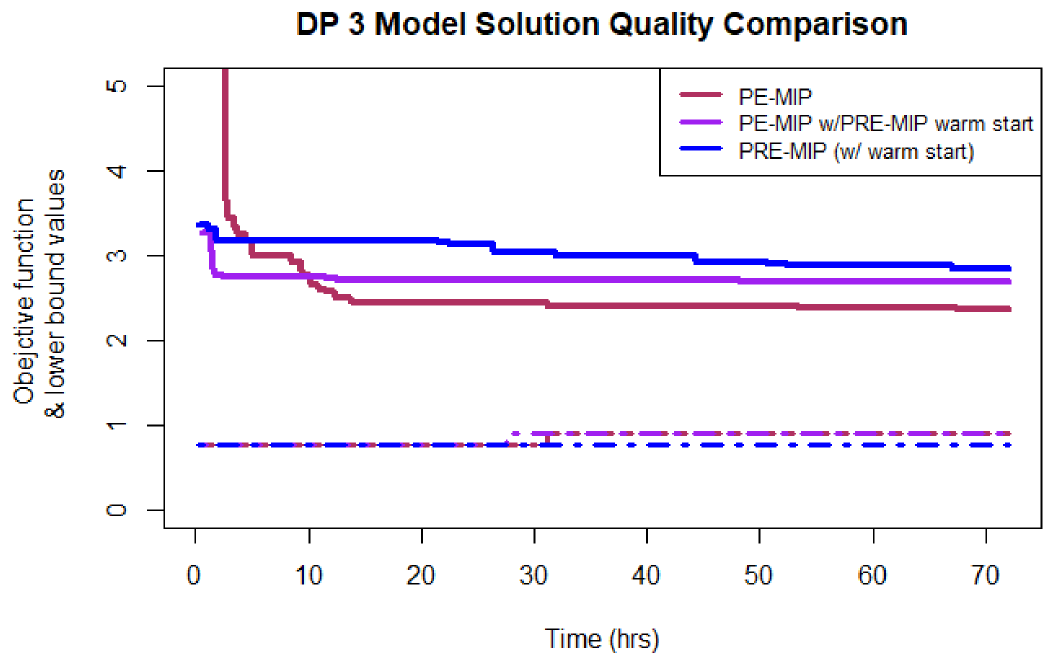


Figure 4.8. DP 3 objective function value and lower bound comparison between PE-MIP, PE-MIP with PRE-MIP warm start, and PRE-MIP over 72 hours. Notice that the lower bounds move very little, PE-MIP achieves the best objective function value, and PRE-MIP attains a feasible solution much faster than PE-MIP.

This experiment leads the research team to multiple conclusions. We acknowledge that PE-MIP continues to achieve the best objective function value under long runtime horizons, and as the results from DP 2 demonstrate, is the most adept at finding a feasible solution in more complex problems. We theorize that symmetry in the model may be causing each model's struggle to close the gap in DP 3. Many different feasible solutions result in the same objective function value and result in CPLEX's branch and bound algorithm exploring redundant portions of the decision space which takes up valuable time and does not change the lower bound or objective function value. Second, although PRE-MIP and PE-MIP with the PRE-MIP warm start are outperformed by PE-MIP over long runtime horizons, both models achieved an objective function value within 0.5 of PE-MIP's best objective function value in only 1.5 hours. This compromise between solution speed and solution quality may be attractive to the model users. Similarly, the PRE-MIP warm start improves the model's ability to return a "fast and dirty" solution in both solution quality and solution speed as demonstrated by DP 1. Finally, we conclude that beginning the model with a Path-Routes based warm start should be permanently added to the model due to its ability to aid in returning a feasible solution. We make no claim at this time that route enumeration and column generation should be added to the main model, as PRE-MIP did not outperform PE-MIP in returning a better solution under long runtime horizons.

CHAPTER 5: Conclusions and Future Work

This thesis utilized a route enumeration reformulation and a warm start as well as experiments to gain insights into the following research questions:

1. Does route enumeration and column generation result in a faster solution without sacrificing solution quality?
2. Does employing a warm start result in a faster solution without sacrificing solution quality?

To these ends, the research team designed the PRE-MIP model, which added route enumeration and route filtering to the existing PE-MIP model through column generation. The research team also implemented a warm start methodology and restricting routes to exact copies of the serial paths in both PRE-MIP and PE-MIP. The experiments described in Chapter 4 tune PRE-MIP parameters and provide insight into the research questions. This chapter draws conclusions based in analysis from the research team's experiments with PRE-MIP regarding PRE-MIP's value in providing OAD with a tool they can use to model force closure in operational planning, analysis of alternatives, and wargaming. The chapter concludes by suggesting potential avenues of analysis which may improve the model's utility.

5.1 PRE-MIP Parameters

Both the warm start experiment and PRE-MIP vs PE-MIP experiments demonstrate that employing a warm start is very helpful in improving the runtime required to return a feasible solution, so the research team concludes that $warmstart = True$ is the best value for the model in this instance. The route filtering experiment further demonstrates that employing the maximum restriction on the route generation algorithm results in the fastest warm start solution in this instance. Therefore, the research team concludes that the warm start consisting of WS_PATH_NODES = Path-Routes is a valuable addition to the model, although we keep both $warmstart$ and WS_PATH_NODES as parameter values for user entry to account for instances in which $warmstat = False$ or a different value for

WS_PATH_NODES yield better results. The route filtering experiment does not reveal an optimal value for the parameter MIN_PATH_ARCS. In the absence of an empirically-determined optimal value, the research team concludes that $MIN\_PATH\_ARCS = 1$ is the best default value for PRE-MIP, given that this retains all routes which have the ability to transport any serials through the network toward their destination.

5.2 PRE-MIP vs PE-MIP

The experiments do not return results which emphatically support adopting PRE-MIP in place of PE-MIP. As shown in DP 3 of the PRE-MIP vs PE-MIP experiment, PE-MIP returns the best objective value over long runtime horizons for the Two MLR instance. However, PRE-MIP returns a significantly better solution in half the time when solving each model to the absolute gap termination condition, due to PRE-MIP's warm start. Therefore, the research team concludes that a warm start using routes based on exact copies of serial paths ought to be implemented moving forward.

5.3 Future Work

As DP 3 of the PRE-MIP vs PE-MIP experiment shows, both PRE-MIP and PE-MIP make very similar progress in closing their relative gaps after about 24 hours runtime. As $MAXK$ increases in value with respect to the number of nodes in serial paths, more legs of connectors' routes carry no serials and therefore do not impact the objective function value. The research team observed during development that in DP 3 of the PRE-MIP vs PE-MIP experiment that many of the routes were functionally equivalent, in that many routes may be assigned to a connector without changing the objective function value. We theorize that symmetry in the model may be causing each model's struggle to close the gap in DP 3. Many different feasible solutions result in the same objective function value and result in CPLEX's branch and bound algorithm exploring redundant portions of the decision space. This takes valuable time and does not change the lower bound or objective function value. Future work may explore adding a tie-breaker to the objective function to reduce this symmetry. We propose that a tie-breaker which accounts for the last serial delivery time or rewards not using a connector at all may help reduce symmetry and improve the speed at which the model closes the relative gap.

As DP 2 of the PRE-MIP vs PE-MIP experiment shows, both models struggle to find a feasible solution in extremely complex instances. As mentioned in Chapter 1, the SMASH heuristic algorithm is also designed to solve force closure problems, and is able to solve instances similar to the Two MLR instance in seconds. However, this attractive solution time comes at the expense of solution quality, as SMASH's heuristic nature precludes analysis regarding its solution quality. Using the SMASH solution as a warm start to PRE-MIP or PE-MIP may yield valuable improvements in the time required to find a feasible solution in large and complex problem instances.

THIS PAGE INTENTIONALLY LEFT BLANK

List of References

Brown G, Carlyle M, Dell R, Brau J (2013) Optimizing intratheater military airlift in Iraq and Afghanistan. *Military Operations Research* 18(3), https://www.mors.org/Publications/MOR-Journal.

Christafore RM (2017) Generating ship-to-shore bulk fuel delivery schedules for the marine expeditionary unit. Master's thesis, Graduate School Operations Research Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/55582.

Desrochers M, Desrosiers J, Solomon M (1992) A new optimization algorithm for the vehicle routing problem with time windows. *Operations Research* 40(2):342–354, http://www.jstor.org/stable/171457.

Estrada E, Knight P (2015) Introduction to Network Theory, Euler and the Konigsberg Bridge, *A First Course in Network Theory* (Oxford University Press, New York).

Ford LR, Fulkerson DR (1956) Maximal flow through a network. *Canadian Journal of Mathematics* 8, https://www.cambridge.org/core/journals/canadian-journal-of-mathematics/article/maximal-flow-through-a-network.

Harris T, Ross F (1955) Fundamentals of a method for evaluating rail net capacities. Technical Report RM-1573, RAND Corporation, Santa Monica, CA, https://apps.dtic.mil/sti/pdfs/AD0093458.pdf, unclassified 1999.

Headquarters, US Marine Corps (2021) Tentative manual for expeditionary advanced base operations, https://www.mcwl.marines.mil/TMEABO/.

Holland C, Levis J, Nuggehalli R, Santilli B, Winters J (2017) UPS optimizes delivery routes. *INFORMS Journal on Applied Analytics* 47(1).

Laporte G (1992) The vehicle routing problem: An overview of exact and approximate algorithms. *European Journal of Operational Research* 59.

Liong CY, Ismail WR, Omar K, Zirour M (2008) Vehicle routing problem: models and solutions. *Journal of Quality Measurement and Analysis* 4(1):205–218, https://core.ac.uk/download/pdf/23174078.pdf.

Mirsch AM (2022) A path enumeration reformulation of the schedule mixed integer program supporting expeditionary advanced base operations. Master's thesis, Graduate School Operations Research Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/70755.

Roofner TW (2022) Smash workbook and example scenario data demonstration. Personal communication.

Royset JO, Wets RJB (2022) *An Optimization Primer*, 1 ed. Springer Series in Operations Research and Financial Engineering (Springer Cham), https://doi.org/10.1007/978-3-030-76275-9.

Schrijver A (2002) On the history of the transportation and maximum flow problems. *Mathematical Programming* 91, https://link.springer.com/article/10.1007%2Fs101070100259.

Sentinella FQ (2021) Using data farming and optimization to enable analysis of concepts of employment for surface connectors. Master's thesis, Graduate School Operations Research Naval Postgraduate School, Monterey, CA, https://hdl.handle.net/10945/67811.

Upton S (2021) *ConnectorFarmer User's Manual version 0.14*, unpublished user's manual, SEED Center for Data Farming, Monterey CA.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California