NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

REAL-TIME SOLUTIONS OF ROBUST ENERGY-AWARE UNMANNED AERIAL VEHICLE ROUTING

by

Tyler W. Cotney

June 2023

Thesis Advisor:	Emily M. Craparo
Co-Advisor:	Vladimir N. Dobrokhodov
Second Reader:	Nicholas Ulmer

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 REAL-TIME SOLUTIONS OF ROBUST ENERGY-AWARE UNMANNED AERIAL VEHICLE ROUTING	**5. FUNDING NUMBERS**
6. AUTHOR(S) Tyler W. Cotney	
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 Marine Corps' force design priorities call for a small mobile force capable of conducting expeditionary advanced basing operations (EABO). Unmanned aerial vehicles (UAV) are the perfect platform to support this mission, but the Marine Corps will need an energy-aware routing system capable of solving a system of multiple depots and UAVs while considering weather forecasts. Due to limited internal energy available on UAVs, units must carefully consider energy consumption rates when routing assets. Weather variance can significantly impact energy requirements and must be accounted for to produce a robust energy-feasible solution. Works by Jatho, Haller, and Won, in 2020, 2021, and 2022, respectively, laid the groundwork for this thesis model by creating a final multiple depot vehicle routing problem (MDVRP) that accounts for an ensemble weather forecast. Large instances of the model can take days to solve and cannot support real-time operations. This thesis reduces the computation time of the final model by several orders of magnitude to support real-time use at a practical scale. We achieve this by implementing several warm start techniques to improve overall solve time. We also created additional data and an automated scenario builder to quickly compare complex sustainment networks and emerging UAV technologies.

14. SUBJECT TERMS unmanned aerial vehicle, UAV, traveling salesperson problem, TSP, multiple depot vehicle routing problem, MDVRP, expeditionary advanced basing operations, EABO	**15. NUMBER OF PAGES** 53
	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.

REAL-TIME SOLUTIONS OF ROBUST ENERGY-AWARE UNMANNED AERIAL VEHICLE ROUTING

Tyler W. Cotney
Major, United States Marine Corps
BS, United States Naval Academy, 2011

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

Vladimir N. Dobrokhodov
Co-Advisor

Nicholas Ulmer
Second Reader

W. Matthew Carlyle
Chair, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

The Marine Corps' force design priorities call for a small mobile force capable of conducting expeditionary advanced basing operations (EABO). Unmanned aerial vehicles (UAV) are the perfect platform to support this mission, but the Marine Corps will need an energy-aware routing system capable of solving a system of multiple depots and UAVs while considering weather forecasts. Due to limited internal energy available on UAVs, units must carefully consider energy consumption rates when routing assets. Weather variance can significantly impact energy requirements and must be accounted for to produce a robust energy-feasible solution. Works by Jatho, Haller, and Won, in 2020, 2021, and 2022, respectively, laid the groundwork for this thesis model by creating a final multiple depot vehicle routing problem (MDVRP) that accounts for an ensemble weather forecast. Large instances of the model can take days to solve and cannot support real-time operations. This thesis reduces the computation time of the final model by several orders of magnitude to support real-time use at a practical scale. We achieve this by implementing several warm start techniques to improve overall solve time. We also created additional data and an automated scenario builder to quickly compare complex sustainment networks and emerging UAV technologies.

THIS PAGE INTENTIONALLY LEFT BLANK

Table of Contents

THIS PAGE INTENTIONALLY LEFT BLANK

List of Figures

THIS PAGE INTENTIONALLY LEFT BLANK

List of Tables

THIS PAGE INTENTIONALLY LEFT BLANK

List of Acronyms and Abbreviations

NPS Naval Postgraduate School

BVP boundary value problem

mTSP multiple traveling salesperson problem

TSP traveling salesperson problem

PMP Pontryagin's maximum principle

UAV unmanned aerial vehicle

VRP vehicle routing problem

MDMTSP multiple depot multiple traveling salesperson problem

MDVRP multiple depot vehicle routing problem

MIP mixed integer program

IPM interior point method

MILP mixed integer linear program

MINLP mixed integer non-linear program

MOILP multi-objective integer linear program

CP constrained program

DP design point

Wh watt-hour

JSON JavaScript Object Notation

USMC United States Marine Corps

EABO	expeditionary advanced base operations
SIF	stand-in-forces
WEZ	weapon engagement zone
EAB	expeditionary advanced base

Executive Summary

Marine Corps force design priorities call for a smaller, more mobile force capable of operating in expeditionary advanced bases (EABs) within enemy weapon engagement zones (WEZs). A potentially effective means for logistically supporting these expeditionary advanced base operations (EABO) is with the use of UAVs; however, the Navy and Marine Corps do not currently have an effective and efficient means of routing a suite of UAVs. Most routing mechanisms do not account for weather when determining energy usage. Previous model formulations for our problem account for weather and weather uncertainty, but are too complex to solve in real time.

This thesis solves the model iteratively developed by Jatho (2020), Haller (2021), and Won (2022). Jatho laid the initial groundwork for the model by solving the optimal flight trajectory of a UAV between two nodes using a boundary value problem (BVP). Jatho then used this information to solve a simple vehicle routing problem (VRP) with optimal trajectories. Haller built upon Jathos work by reformulating the VRP as a multiple depot vehicle routing problem (MDVRP) and adding a priority system to demand nodes. Finally Won improved the formulation by including an ensemble weather forecast, developing a more robust solution for varying weather scenarios.

Our work improves the solve time of the final model Won (2022) developed. Although Won's model solves our scenario in approximately 20 minutes, more complex scenarios take multiple days to solve. Our work improves these solve times, making the final model useful in real time for day-to-day operations. We improve solve time by implementing a warm start to Won's model in two ways. We first calculate a worst-case energy cost matrix across all ensemble weather forecasts. We use this conservative energy cost matrix to then simplify Won's model and solve for an initial solution. We then use the initial solution as a warm start and are able to halve the overall solve time. For our second technique we decompose Won's model, iteratively solving it for subsets of the UAVs. We compile each of these solutions into a warm start to Won's full model. Using our second technique we solve the entire model instance, including the time to develop the warm start and solve the full model, in less than 30 seconds.

We also implement a scaling factor to reflect different UAV weights in the energy consumption calculation. Though we conduct our experiments using nominal energy requirements, future research can use this implementation for a variety of different payloads. In addition to the scaling factor we create a means to rapidly develop additional scenarios and data for further experimentation and research.

References

Haller JR (2021) Expanding optimization of energy efficient UAV routing in support of Marine Corps expeditionary advanced base operations with multiple supply depots. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/68328.

Jatho A (2020) Optimizing energy efficient UAV routing in support of Marine Corps expeditionary advanced base operations. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/65554.

Won DD (2022) Robust energy-aware unmanned aerial vehicle routing using ensemble weather forecasts. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/70783.

Acknowledgments

I would like to thank both Professors Craparo and Dobrokhodov for their patience and mentorship through this process as well as all of the Operations Research faculty that taught the essential skills used throughout this thesis.

I would also like to thank my wife, Rhendi Cotney, for her unwavering support despite caring for our two children while I remained at my desk most days late into the night working on my "Thesis."

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 1: Introduction

The United States Marine Corps (USMC) is modernizing its force to support expeditionary advanced base operations (EABO) and the stand-in-forces (SIF) concept. Headquarters, U.S. Marine Corp defines EABO as “a form of expeditionary warfare that involves the employment of mobile, low signature, persistent, 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 in order to conduct sea denial, support sea control, or enable fleet sustainment” (2023, p. 1-2). As the USMC continues to explore EABO tactics, Berger (2022) has identified sustainment as the “pacing” function for operations.

Alexander et al. (2020) is quoted by Headquarters, U.S. Marine Corp (2023) and states “The ability to sustain lighter, faster, and more distributed operations is critical to maintaining the U.S. naval force’s competitive advantage.” The USMC has explored multiple concepts for the best way to provide this sustainment. Logisticians across the DOD logistics enterprise have proposed everything from small littoral logistics craft to foraging as a sustainment solution; however, Headquarters, U.S. Marine Corp (2023) has asked the enterprise to explore unmanned aerial vehicle (UAV) capabilities to fill this gap. Headquarters, US Marine Corps has specifically stated in their EABO tentative manual that “The use of uncrewed aerial delivery capabilities to reach remote maneuver units that may be operating in areas void of traditional infrastructure to provide delivery or retrograde of critical supplies may be necessary particularly to link those units to the sustainment aboard littoral surface connectors.”

The DOD is exploring several materiel solutions to support a UAV strategy for EABO sustainment; however, until the Marine Corps fields these solutions further research into the planning and execution of military UAV sustainment strategies is sparse. It is important that both planners and researchers continue to develop these strategies in concert with the fielding effort. This will ensure USMC tactical units can swiftly and efficiently implement more capable UAVs when they receive them.

This thesis continues the efforts of Jatho (2020), Haller (2021), and Won (2022) to develop an energy-efficient method to route a suite of UAVs across a sustainment network. We investigate several techniques to significantly improve how quickly we can solve Won's model. We also increase utility in our implementation from previous formulations by including mixed UAV types with respect to energy usage. As an effort to assist in further research and experimentation we have also created additional data and an automated scenario builder to quickly compare complex sustainment networks and emerging UAV technologies.

CHAPTER 2:
Background

This thesis builds upon prior work on last-mile logistics using UAVs, as well as related work in integer linear programming. We begin by providing an overview of three prior theses that laid the foundation for our work, then discuss the use of warm starts in integer linear programming. For a more comprehensive overview of other related topics such as last-mile logistics and the use of UAVs for logistics operations, please refer to the work of Jatho (2020), Haller (2021), and Won (2022).

2.1 Jatho: BVP and VRP

Jatho (2020) lays the initial framework for our work by formulating the UAV routing problem in a two-stage manner. His first stage utilizes a solution of a boundary value problem (BVP) using Pontryagin's maximum principle (PMP) to determine an optimal flight path given UAV characteristics and weather data. The output of his first layer is a matrix of energy costs for flying between connected nodes in a network. Jatho (2020) uses this energy cost matrix as input to solve a second layer mixed integer linear program (MILP) based on the vehicle routing problem (VRP) variation of the multiple traveling salesperson problem (mTSP). His solution provides an energy-feasible routing of a suite of UAVs traveling from and to a single depot node while maximizing the supplies delivered to a set number of demand nodes. His solution also accounts for both the routes taken between two nodes and the optimal flight paths that should be used on those routes. Jatho outlines the BVP, mTSP, and VRP and their formulations.

2.2 Haller: MDVRP

Haller (2021) extends Jatho's second stage to incorporate multiple UAV depots. Haller relies on the multiple depot multiple traveling salesperson problem (MDMTSP), and he specifically uses a variant of the MDMTSP known as the multiple depot vehicle routing problem (MDVRP). Haller also implements a priority system into his formulation allowing for demand nodes to be serviced based upon their urgency of need, as well as a depot selection model for use in the initial planning stages of an operation.

2.3 Won: Ensemble Weather Model

Won (2022) extends the work of Jatho and Haller to include a stochastic element to the energy cost matrix, as well as implementation of partial deliveries at demand nodes. Won's model determines a robust solution to the MDVRP that takes into account multiple weather forecasts and determines a solution with a higher chance of real-world feasibility over Haller's model. Won achieves partial deliveries in his formulation by breaking up demand nodes in his network into multiple nodes whose sum of demand is equal to the original demand node and whose cost to travel between these multiple nodes is set to zero. This formulation allows multiple UAVs to travel to the same node in order to satisfy each of the demands of the new split delivery nodes. Although Won's formulation adds significant robustness and utility to the overall problem, it also adds complexity. The added complexity of Won's formulation leads to computational times close to an hour for small scale scenarios and multiple days for more complex instances. High CPU time prompts one of the key focus areas of this research thesis.

2.4 Warm Starts in Integer Programming

Engau (2012) describes a warm start as an optimal or feasible solution from a less complex algorithm or formulation used as a starting point to reduce the solve time of the more complex formulation. Warm starts are widely used throughout the literature to improve the solve time of a multitude of optimization problems, including large scale mixed integer programs (MIPs). Gondzio (1998) demonstrates how warm starting an infeasible interior point method (IPM) can lead to a solution using a cutting plane scheme. Ralphs and Güzelsoy (2006) demonstrate improved computation time using warm starts on MILPs. Their findings show that with relatively little change in input data between a warm start and full scale problem instance, they can achieve significant gains in overall solution time. Filomena and Lejeune (2014) demonstrate the enhanced utility of warm starts in more complex formulations of mixed integer non-linear programs (MINLPs), specifically finding that using their warm start allows for an order of magnitude faster computation time. Pour et al. (2018) show a warm start approach that uses a blend of MIP and constrained program (CP) decreases the MIP master problem solve time from a few hours to a few seconds. Forget et al. (2022) demonstrates the utility of warm starting branch and bound algorithms on multi-objective integer linear programs (MOILPs).

This thesis uses two approaches to generate a warm start solution. First, we use the ensemble cost matrix from the model developed by Won (2022) to create a single matrix of worst-case energy costs for each pair of nodes, then solve the full problem (including all UAVs) with this single conservative cost matrix. We then use the solution from this simplified model as a warm start to the full problem with the original ensemble cost matrix. Second, we solve Won's model iteratively for each UAV type and each individual UAV and use the resulting composite solutions as a warm starts to the full problem.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 3: Warm Start of Energy-Optimal Routing

This chapter describes two methods for producing a warm start solution meant to improve the overall solve time of the model formulated by Won (2022). In both cases, we create a solution that is guaranteed to be feasible in the Won model, but is not necessarily optimal.

Our first approach involves replacing the ensemble cost matrix $C_{i,j,e}$ with a single conservative cost matrix $\hat{C}_{i,j}$. We construct this conservative cost matrix in such a way as to guarantee feasibility of the resulting UAV routes for each ensemble member e. We are then able to slightly modify Won's formulation by removing the ensemble set E entirely from the objective function and each respective constraint, or alternatively, setting the cardinality of E to one to create the warm start solution.

Our second approach solves for UAV routes sequentially rather than simultaneously. We perform multiple solves of the Won (2022) model, each considering a subset of UAVs, then compile the solutions into one solution to pass as a warm start to the full problem. While our first approach reduces the cardinality of the set E in order to improve the tractability of the warm start problem, this approach reduces the cardinality of the set of UAVs, M, to achieve the same goal.

3.1 Base Formulation

Won (2022) provides the basis of our formulation that we attempt to speed up throughout this thesis. This section provides the complete formulation of Won's model and our adjustments noted in red. This formulation is provided for the convenience of the reader as we reference the sets and indices for this formulation in future sections and chapters when we outline our 'warm start' efforts. We will detail our changes in the formulation but for further details on individual constraints and their functions see Won (2022).

We noticed that in all previous iterations of our formulation, the cost matrix $C_{i,j,k,e}$ included a k index to account for energy costs of different UAV types; however, past implementations of this formula assumed homogeneous UAVs for the cost matrix. Due to this assumption, previous theses did not implement a k index into the cost matrix. We will also note for clarity that previous implementations included all other k indices found throughout the formulation and only excluded the index from the cost matrix. Although we make the same assumption of homogeneity in our testing, we have also included an improvement in the formulation and implementation to facilitate follow on study using mixed UAVs for the cost matrix. We remove the k index from the energy cost matrix entirely and include a scaling factor, $Scale_k$, to account for the change in energy costs for UAVs that differ from our nominal UAV. We calculate the scaling factor by determining the ratio of the mass of the k UAV to the mass of nominal UAV used by Jatho (2020) in his BVP. McCormick (2011) shows that given a nominal weight and power, any additional power required for flight is proportional to the square of the increase in mass. We use this as our scaling factor k and apply it to the energy cost matrix throughout our formulation to account for varying energy costs on different types of UAVs.

$$Scale_k = \left(\frac{Mass_k}{Mass_{nominal}}\right)^2 \tag{3.1}$$

Sets and Indices:

$i, j \in N = \{1, \ldots, n\}$	Nodes
$k \in M = \{1, \ldots, m\}$	UAVs
$e \in E = \{1, \ldots, e\}$	Ensemble members
$P \subseteq N$	Depot nodes
$Q \subseteq N$	Demand nodes

Data:

$C_{ij\not{k}e}$ = Energy for nominal UAV to travel from node i to node j for ensemble member e [Wh]

d_j = Demand at node j [Pounds]

s_k = Capacity of UAV k [Pounds]

b_k = Starting depot of UAV k

$EnergyMax_k$ = Energy capacity of UAV k [Wh]

$Scale_k$ = Scaling factor for energy required for UAV k

ϵ = Penalty weight

γ = Penalty weight

R_j = Reward for delivering supplies to node j, based on demand priority

Decision Variables:

X_{ijk} = Binary variable representing whether or not UAV k travels from node i to node j

Z = Maximum energy consumption by any UAV

u_i = Dummy variable to prevent degenerate subtours

$$\max \sum_{i=1}^{n} \sum_{j:j\neq i} \sum_{k=1}^{m} X_{ijk} R_j - \gamma Z - \epsilon \sum_{i=1}^{n} \sum_{j:j\neq i} \sum_{k=1}^{m} \sum_{e=1}^{E} X_{ijk} Scale_k \frac{C_{ij\not{k}e}}{|E|} \tag{3.2}$$

$$Z \geq \sum_{i=1}^{n} \sum_{j:j\neq i} \sum_{k=1}^{m} Scale_k C_{ij\not{k}e} X_{ijk} \qquad \forall\, e \in E \tag{3.3}$$

$$\sum_{j\in Q} X_{b_k jk} \leq 1 \qquad \forall\, k \in \{1, \ldots m\} \tag{3.4}$$

$$\sum_{i\in Q} \sum_{j\in P} X_{ijk} \leq 1 \qquad \forall\, k \in \{1, \ldots m\} \tag{3.5}$$

$$\sum_{i=1}^{n} \sum_{k=1}^{m} X_{ijk} \leq 1 \qquad \forall\, j \in Q \tag{3.6}$$

$$\sum_{j=1}^{n} \sum_{k=1}^{m} X_{ijk} \leq 1 \qquad \forall\, i \in Q \tag{3.7}$$

$$\sum_{i\in N} X_{irk} = \sum_{j\in N} X_{rjk} \qquad \forall\, r \in Q; k \in \{1, \ldots m\} \tag{3.8}$$

$$\sum_{j\in Q} X_{b_k jk} = \sum_{i\in Q} \sum_{j\in P} X_{ijk} \qquad \forall\, k \in \{1, \ldots m\} \tag{3.9}$$

$$u_i - u_j + (n-m) * \sum_{k=1}^{m} X_{ijk} \leq n - m - 1 \qquad \forall\, i, j \in Q \tag{3.10}$$

$$\sum_{i=1}^{n} \sum_{j:j\neq i} d_j X_{ijk} \leq s_k \qquad \forall\, k \in \{1, \ldots m\} \tag{3.11}$$

$$\sum_{i=1}^{n} \sum_{j:j\neq i} Scale_k C_{ij\not{k}e} X_{ijk} \leq EnergyMax_k \qquad \forall\, e \in E; \forall\, k \in \{1, \ldots m\} \tag{3.12}$$

$$X_{ijk} \in \{0, 1\} \qquad \forall i, j \in \{1, ...n\} \tag{3.13}$$

$$u_i \in \mathbb{W} \qquad \forall i \in Q \tag{3.14}$$

$$Z > 0 \tag{3.15}$$

3.2 Conservative Cost Matrix Approach

We construct our conservative cost matrix $\hat{C}_{i,j}$ by utilizing the worst-case energy metric of each arc across all ensemble members.

$$\hat{C}_{i,j} = \max_{e \in E} C_{i,j,e} \tag{3.16}$$

The resulting cost matrix $\hat{C}$ can potentially be quite conservative. For example, suppose one ensemble member predicts light easterly winds, while another ensemble member predicts light westerly winds. The conservative cost matrix will utilize the highest cost for every arc, reflecting headwinds when traveling both east and west. While the resulting routes may be quite conservative, this guarantees energy-feasible UAV routes for each ensemble member when utilized in the full ensemble model from Won (2022).

To describe our implementation of the expression shown in Equation 3.16, we start by dissecting the format of our original ensemble cost matrix. The ensemble cost matrix is structured as a Python dictionary with three-element tuples as keys and energy costs in watt-hours (Wh) as values. Each tuple key corresponds to a source node, destination node, and weather ensemble member, respectively.

For each arc (i, j), we check the cost across each ensemble member. We determine the maximum cost of all the i, j ensembles and set this value as the cost for arc (i, j) in our new cost matrix. Below is an example of a three-member ensemble cost matrix for an arc between nodes 1 and 2.

Sets and Indices:
$e \in E = \{1, ..., e\}$, Ensemble members

$i, j \in N = \{1, ..., n\}$, Nodes

Data:

$C_{i,j,e}$ = Energy to travel from node i to node j for ensemble member e [Wh]

Pseudo-code:

$e \in E = \{1, 2, 3\}$

$C_{1,2,e} \leftarrow \{(1, 2, 1) : 1200, (1, 2, 2) : 2000, (1, 2, 3) : 400\}$

$\hat{C}_{1,2} \leftarrow \{0\}$

for $cost$ in $C_{1,2,e}(values)$ **do**

  if $cost > \hat{C}_{1,2}$ **then**

    $\hat{C}_{1,2} \leftarrow cost$

  end if

end for

We use the above example and pseudo-code to loop through the ensemble cost matrix three times. After the first loop we set the $NewCostMatrix_{1,2}$ to 1,200 since it is greater than 0; after the second loop we set the $NewCostMatrix_{1,2}$ to 2,000 since it is greater than 1,200; after the final loop we retain $NewCostMatrix_{1,2}$ at 2,000 since 2,000 is greater than 400. We are finally left an entry in our new cost value for arc $(1, 2)$ of 2,000, which was the highest energy for that segment among all ensembles. We then continue to execute this code for every segment in our fully connected network until we have completed our new cost matrix.

Once we have created our new cost matrix we then solve the Won (2022) formulation. From this simpler formulation we then extract the values of Z, u_i, and $X_{i,j,k}$ to pass into the full model as a warm start. Our extracted values are guaranteed to produce an energy-feasible, though not necessarily optimal, solution for the full model. We hypothesize that by providing an initial energy-feasible route to our optimizer to solve the full model, we are giving our optimizer a closer starting solution to work with, potentially speeding up the solve time.

3.3 Decomposition by UAVs

Our second technique for obtaining a warm start solution decomposes the problem by UAVs. The full model formulation by Won (2022) includes multiple UAV types and quantities. We decompose Won's formulation in two ways. In our first technique, we solve the model iteratively for each UAV type, compile our solution, and then pass it as a warm start. In our second technique, we solve the model iteratively for each individual UAV, compile our solution, then pass it as a warm start. Our first technique requires fewer sub-problems, but each problem is more complex than the numerous small subproblems we solve in our second technique. We expect that by using two separate decomposition techniques we can determine whether it is computationally faster to decompose our problem down to its simplest elements or faster to solve a minimal number of decomposed solutions.

3.3.1 Decomposition by UAV Type

Like Won (2022), we consider three types of UAVs: small, medium, and large capacity. It may be impossible to satisfy some demands using small or medium UAVs due to either the weight of the items demanded, or the distance from a depot. Therefore, when decomposing the problem by UAV type, we first solve for the optimal routes for the small UAVs. This allows the less capable UAVs to satisfy those demands they are able to satisfy, while leaving more difficult tasks to the more capable UAVs. After we have solved the model with this subset of UAVs, we extract the solutions for Z, the highest energy cost of any one UAV, and $X_{i,j,k}$, the route used by a UAV, so we may later compile the results. We also ensure that subsequent subproblems do not allow UAVs to visit nodes already visited by previous UAV types by setting the demand of each visited node to one plus the maximum capacity of our largest UAV. We then repeat this process for the medium UAVs and finally the large UAVs. Finally, we pass our compiled values for $X_{i,j,k}$ and Z into Won's model as fixed variables and quickly solve for u_i, then pass the resulting complete solution into the full model as a warm start. We provide pseudocode of our decomposition below, where $Model \leftarrow M, d$ denotes a solve of our optimization model with the set of UAVs M and demands d as input, and where $Model \leftarrow Init\_X_{i,j,k}, Init\_Z$ denotes a solve of our optimization model with fixed variables X and Z in order to obtain values for u.

Sets and Indices:

$k \in M = \{1, ..., m\}$ UAVs

$l \in L = \{1, ..., |L|\}$ Subsets of UAVs to solve over

$k \in U_l$ UAVs in subset l, where all sets U_l together constitute a partition of M

Decision Variables:

X_{ijk} Binary variable representing whether or not UAV k travels from node i to node j

Z Maximum energy consumption by any UAV

u_i Dummy variable to prevent degenerate subtours

Data:

d_j = Demand at node j [pounds]

s_k = Capacity of UAV k [pounds]

Pseudo-code:

$New\_d_j \leftarrow d_j$

$Init\_Z \leftarrow 0$

$Init\_X_{i,j,k} \leftarrow \{\}$

for l in L **do**

$Model \leftarrow U_l, New\_d_j$

$Init\_X_{i,j,k}.append(Model.X_{i,j,k}.extract\_values())$

if $Model.Z.extract\_values() > Init\_Z$ **then**

$Init\_Z \leftarrow Model.Z.extract\_values()$

end if

for (i, j, k) in $Init\_X_{i,j,k}.keys()$ **do**

if $Init\_X_{i,j,k}[i, j, k] > 0$ **then**

$New\_d_j[j] = \max_{k \in M} s_k + 1$

end if

end for

end for

$$Model \leftarrow Init\_X_{i,j,k}, Init\_Z$$
$$Init\_u_i \leftarrow Model.u_i.extract\_values()$$

Having obtained initial decision variable values $Init\_X_{i,j,k}$, $Init\_Z$, and $Init\_u_i$ we are then ready to warm start Won's full model.

3.3.2 Decomposition by Individual UAV

Although the three subproblems described in Section 3.3.1 are significantly simpler to solve than the full problem, they are nonetheless challenging, especially when many UAVs of a particular type are present. Therefore, we also decompose by individual UAVs, adding numerous additional subproblems to solve but significantly decreasing the complexity of each problem. We start by solving Won's model with our smallest individual UAV, iterating through every UAV from smallest to largest and compiling their solutions as described in Section 3.3.1. We use the same pseudocode as in Section 3.3.1, except that we now have $|L| = |M|$, and each subset l contains only a single UAV.

3.4 Scenario Builder

The full model implementation by Won (2022) and utilized in this thesis relies on three input files with multiple data inputs to run. Specifically, we extract our cost matrix using the work by Jatho (2020) in his first layer formulation given a PKL file with weather forecasts and the geography of our node network, as well as a PKL file with UAV characteristics. With these two files, we then create our scenario, the third required input file. We develop our scenario by deciding on quantities of UAV types and their starting locations, node demand quantities, node demand urgency, and depth of split deliveries and their demands at each node. Won created his scenarios manually, formatting JavaScript Object Notation (JSON) files to be used as input into the model. We have created an automated scenario builder that takes as inputs the number of nodes, number of depot nodes, and the quantity of large, medium, and small UAVs and outputs a scenario in the form of a formatted JSON file. The scenario builder also takes optional arguments to include split shipments on nodes and the number of split shipments for each node, as well as the ability for the user to determine overall demand or calculate demand automatically based on overall UAV capacity.

We created our scenario builder setting the default demand quantity to 80% of our overall UAV capacity. Distributing the demand was initially a challenge due to the large variance in capability among our different UAV types. For instance, our small UAVs can only carry 55 lbs of cargo where our medium UAVs can carry up to 5000 lbs. If we evenly distributed the demand across all nodes based upon 80% of our UAV capacity, our small UAVs would not have the capacity to service any nodes. Considering we typically have more small UAVs than any other type, this demand distribution would be unrealistic. We solve this problem by first characterizing our demand nodes as small, medium, and large. We then distribute across the small, medium, and large nodes appropriately to ensure our network is serviceable by our typical UAV makeup. Specifically, we assign 60%, 25%, and 15% of demand nodes small, medium, and large labels, respectively. Once we have labeled our nodes we then assign demand to each one. Our small, medium, and large nodes receive 1%, 24%, and 75% of the demand, respectively. By distributing the demand in this way we ensure that our small UAVs will most likely be able to service 60% of nodes; that medium UAVs will be able to efficiently service at least 25% of our nodes; and that our large UAVs have the option of satisfying large demand at 15% of our nodes.

Having created the demand distribution, the rest of the scenario builder is straight forward. We randomly assign UAVs to depot nodes and randomly assign priorities of need to demand nodes. Our final JSON file reads like a Python dictionary with keys for maximum range, list of UAVs, list of demands, list of depots, and a flag for split deliveries. If a user needs to specify a specific demand or priority for a node within the generated scenario, the user can easily open the JSON and change the characteristics of interest before loading it to the model.

THIS PAGE INTENTIONALLY LEFT BLANK

CHAPTER 4:
Experiments and Results

This chapter describes our computational experiments. We compare solve times using our two warm start techniques against that of the full model formulated by Won (2022), with no warm start. We use Won's infantry battalion scenario, which we outline in Section 4.1, for all testing. We test all models on the original weather data used by Won as well as a perturbed version of the weather data that reflects a great degree of variance and uncertainty. We conduct all tests using Pyomo (version 6.4.2) in Python (version 3.9) with the CPLEX solver (version 22.1.0.0). Our Windows system hardware consists of a four-core 1.8 GHz Intel Core i-7 processor and 16 GB of RAM.

4.1 Scenario

We extract our scenario for testing from Won (2022). The scenario mimics a possible sustainment lay down to support a USMC infantry battalion operating across a small island chain. Figure 4.1 is a visual depiction of our node network.

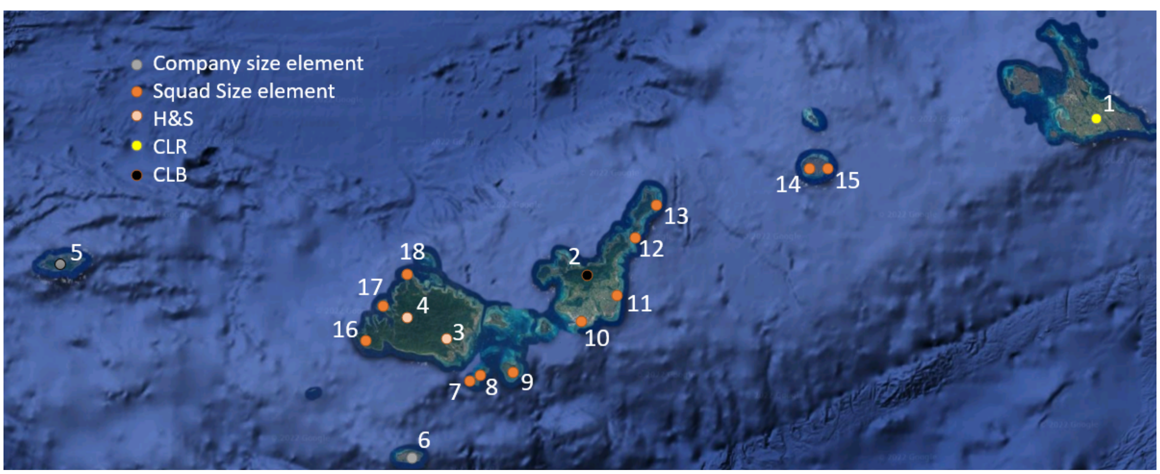


Figure 4.1. Force layout for scenario resupplying infantry battalion. Source: Won (2022) Figure 4.3.

Won (2022) details our node network, demand distribution, and UAV makeup in Table 4.1. Our overall demand in this scenario is 27,013 lbs across 18 nodes with split deliveries enabled. We define split deliveries in this case as multiple demands that can be filled by multiple UAVs at the same node. We set our delivery capacity to 45,330 lbs split between one large UAV, five medium UAVs, and six small UAVs. This scenario results in a MDVRP model with 24,295 decision variables and 2,394 constraints. For termination criteria, we primarily utilize a 50% relative optimality gap for decomposition warm start instances, a 16% relative optimality gap for instances that use a conservative cost matrix, and a 10% relative optimality gap for all full instances (with or without a warm start). We comment on any deviations of these settings within their respective sections.

Table 4.1. Demand node and UAV characteristics for infantry battalion scenario. Source: Won (2022) Table 4.5.

Demand Node	Demand Type and Amount	Total Demand
5	(1) 1,500 U, (2) 1,500 U, (3) 1,000 U, (4) 1,000 U (5) 1,900 U, (6)1,670 U	8,570
6	(1) 1,750 U, (2) 3,500 U, (3) 3,000 U, (4) 4,080 U	12,330
7	(1) 2,500 U, (2) 500 P, (3) 40 R	3,040
8	(1) 500 P, (2) 250 R, (3) 10 U, (4) 5 U	765
9	(1) 750 U, (2) 100 P, (3) 125 R	975
10	(1) 55 R, (2) 30 U, (3) 30 U	115
11	(1) 400 U, (2) 300 R, (3) 10 U	710
12	(1) 30 P, (2) 5 U	35
13	(1) 40 U, (2) 20 R	60
14	(1) 55 U	55
15	(1) 10 U, (2) 20 P	30
16	(1) 60 U, (2) 35 P, (3) 10 P	105
17	(1) 70 P, (2) 30 U	100
18	(1) 90 R, (2) 10 U, (3) 23 R	123

Demand priorities are denoted as: U = Urgent, P = Priority, and R = Routine.

UAV	Payload Capacity (lbs)	Energy (Wh)	Starting Depot
1	20,000	10,929,756	1
2	5,000	1,593,919	2
3	5,000	1,593,919	2
4	5,000	1,593,919	2
5	5,000	1,593,919	2
6	5,000	1,593,919	2
7	55	546,488	3
8	55	546,488	3
9	55	546,488	3
10	55	546,488	4
11	55	546,488	4
12	55	546,488	4

4.2 Conservative Cost Matrix

A warm start approach using a conservative cost matrix yields a considerable decrease in overall solve time. We solve the full problem instance in 1,320 seconds; Won (2022) achieves similar results on his system. Using a conservative cost matrix, we obtain an initial solution in approximately 11 minutes. When we use this initial solution to warm start the full problem, the full problem solves to a 10% relative optimality gap in 5–10 seconds. Table 4.2 displays our solution to the full problem when we use the conservative cost matrix warm start. When we compare the final solution to the full model using the conservative cost matrix warm start to Won's solution in Table 4.3, we see very similar results. Both solutions meet just over 88% of the overall demand, though the conservative cost matrix solution uses approximately 32% more energy. Overall Won's solution results in a 1% better objective value but takes twice as long to solve. Should we encounter scenarios where more than 10 ensemble members are present, we may find additional benefits in solve time by using this method.

As previously discussed, a conservative cost matrix may result in quite conservative flight paths, which may have inferior objective values. To explore the possibility of reducing our level of conservatism, we tune our conservative cost matrix by scaling the energy costs down by 1–10% and iteratively solve our warm start model with these scaled matrices. We then utilize the solution that has the best objective value while remaining feasible in the full model. The scaling method is successful in marginally speeding up the solve time of different conservative cost matrix iterations and results in a better objective value for some instances. These improvements, however, do not decrease the overall solve time as the iterations still take a third to half the amount of time that solving the original full problem takes. This means that after solving two warm start instances, we have lost the speed benefit from using a warm start (for this particular scenario). We also find that tuning this scaling parameter is instance-specific and may not yield energy-feasible routes for all weather scenarios.

Table 4.2. Flight statistics for conservative cost matrix warm start solution.

UAV	Flight Path (Nodes)	Total Resupply (lbs)	Percentage of Capacity Resupplied	Energy Expenditure (Wh)	Percentage of Energy Expended
One	1→5.1→5.4→5.3→ 5.5→6.2→6.3→6→ 6.1→7.2→8.3→9→ 14→15.1→15→1	19,280	96.40%	10,905,509	99.78%
Two	2→11.2→13→ 10.1→2	80	1.60%	1,531,881	96.11%
Three	2→10→8.1→7→ 7.1→8→9.1→ 8.2→3	3,415	68.30%	1,395,271	87.54%
Four	2→16.1→4	35	0.70%	1,289,252	80.89%
Five	2→17→17.1→ 16→16.2→4	170	3.40%	1,382,667	86.75%
Six	2→11.1→12→11→ 10.2→12.1→2	765	15.30%	1,358,007	85.20%
Ten	4→18.1→4	10	18.18%	373,248	68.30%
Eleven	4→18.2→4	23	41.82%	373,248	68.30%

Table 4.3. Flight statistics for infantry battalion scenario with no warm start. Source: Won (2022).

UAV	Flight Path (Nodes)	Total Resupply (lbs)	Percentage of Capacity Resupplied	Energy Expenditure (Wh)	Percentage of Energy Expended
One	1→15.1→14→7.2→ 8→8.2→8.1→16→ 5.1→5→5.2→5.5→ 5.4→5.3→18→6.1→ 6→6.3→7.1→9.1→ 11.1→11.2→12.1→12→ 13.1→13→2	19,930	99.65%	10,406,848	95.20%
Two	2→7→8.3→ 9.2→9→3	3,380	67.60%	1,219,109	76.5%
Four	2→16.2→16.1→ 17.1→17→4	145	2.90%	1,389,295	87.20%
Six	2→11→10.1→ 10→10.2→2	515	10.30%	617,387	38.70%
Twelve	4→18.2→18.1→4	33	60%	373,248	68.30%

4.3 UAV Decomposition by Type

On our first attempt to decompose the problem by UAV type, we encountered a solution time of over five hours when solving for medium UAV with a 50% optimality gap. Thus, we increased the optimality gap for the medium UAV solve to 100%, leaving the optimality gap at 50% for the small and large UAV runs. With these settings, we solve the decomposition instances in a total of 278 seconds, and the full problem with the warm start in an additional 863, giving us an overall solve time of 1,141 seconds (approximately 19 minutes). While this is a modest improvement over the computation time with no warm start, some trial and error was required to determine that the optimality gap for the medium UAV instance should be raised, and this modification may not be required or effective on other problem instances. Furthermore, modifying the optimality gap for medium UAVs resulted in an inferior solution being passed to the full solve, which in turn meant that the full solve required significantly more time to get below its 10% optimality gap. Table 4.4 details our solution from the decomposition by type method. Despite the additional time to solve over the conservative cost matrix method, we do end up with a better objective value. It is also interesting that this solution does not utilize any of the small UAVs, though this is perhaps not surprising considering the large disparity in their supply capacity when compared to the medium and large UAVs.

Table 4.4. Flight path statistics for decomposition by UAV type.

UAV	Flight Path (Nodes)	Total Resupply (lbs)	Percentage of Capacity Resupplied	Energy Expenditure (Wh)	Percentage of Energy Expended
One	1→14→15→15.1→ 12→16.1→6.3→6.2→ 6→16→6.1→5.5→ 5→5.4→5.2→5.3→ 17.1→18.2→18→ 18.1→17→8.3→3	19,838	99.19%	10,410,142	95.25%
Two	2→16.2→4	10	0.20%	1,289,252	80.89%
Three	2→11→10.2→7.1→ 7→7.2→8.2→3	3,480	69.60%	1,240,866	77.85%
Four	2→9.2→3	125	2.50%	843,090	52.89%
Five	2→12.1→13.1→13→ 10→11.2→11.1→2	430	8.60%	1,544,167	96.88%
Six	2→9.1→9→8.1→ 8→10.1→2	1,630	32.60%	1,509,245	94.69%

4.4 Decomposition by Individual UAVs

When we solve the full problem decomposed by individual UAVs we see the most significant improvement in solution time. It takes approximately 20 seconds (total) to solve for the individual UAVs' solutions and compile these into a complete warm start solution, and an additional two seconds to solve the full model using that warm start. Table 4.5 details the solution we obtain from the full problem with a warm start from the individual UAVs decomposition. In this case, we see notable differences between the full model results in Table 4.3 and the individual decomposition results in Table 4.5. We are now using nine UAVs compared to five, while servicing 96.3% of the overall demand. Although we use 7.6% more energy, the additional demand delivered results in a 3.6% improvement in our objective value. Considering the remarkable decrease in overall solution time and increase in objective value, decomposing by UAV for this type of problem is highly recommended.

Table 4.5. Flight path statistics for decomposition by individual UAV.

UAV	Flight Path (Nodes)	Total Resupply (lbs)	Percentage of Capacity Resupplied	Energy Expenditure (Wh)	Percentage of Energy Expended
One	1→15.1→15→14→ 6→6.1→5.4→5.3→ 5→5.1→5.5→6.3→ 6.2→3	19,985	99.93%	8,315,653	76.08%
Three	1→8.1→3	250	5.00%	806,604	50.61%
Four	1→18→17→16→4	220	4.40%	1,419,707	89.07%
Five	1→13.1→13→ 11.2→11.1→10→2	425	8.50%	1,358,785	85.25%
Six	2→12→12.1→11→ 10.1→10.2→9→ 9.2→9.1→8→ 7.1→7→3	4,970	99.40%	1,440,156	90.35%
Nine	3→8.2→8.3→ 7.2→3	55	100.00%	392,540	71.83%
Ten	4→18.2→4	23	41.82%	373,248	68.30%
Eleven	4→16.2→16.1→4	45	81.82%	453,253	82.94%
Twelve	4→17.1→18.1→4	40	72.73%	514,186	94.09%

4.5 Overall Comparison

Table 4.6 provides an overview of our results. The conservative cost matrix is an effective procedure for speeding up the solution time of the full model, but it significantly underperforms when compared to the technique of decomposing by individual UAVs. We are surprised by relative computation time of the decomposition by UAV type and decomposition by individual UAV solves, though we expect this may be an artifact of this particular problem instance. Decomposition by individual UAVs is the best warm start method by a large margin. This technique decreases solve time by two orders of magnitude while yielding the best objective value for this instance. Table 4.7 demonstrates that the full decomposition solution immediately gives CPLEX a solution within its set optimality gap, while the conservative cost matrix method also produces a very good starting point. Again, further experimentation is necessary to determine whether this pattern holds for other problem instances.

Table 4.6. Base case and warm start solution comparisons.

Method	Nodes Serviced	Total Energy Used (KWh)	Total Supplied (lbs)	Demand Met (%)	Computation Time (sec)
Base Case (no warm start)	39/41	14,005,887	24,003	88.86%	1,320
Conservative Cost Matrix	36/41	18,609,083	23,778	88.02%	660
Decomposition by UAV Type	40/41	16,836,762	25,513	94.45%	1,141
Decomposition by Individual UAVs	40/41	15,074,132	26,013	96.30%	22

Table 4.7. Base case and warm start objective comparisons.

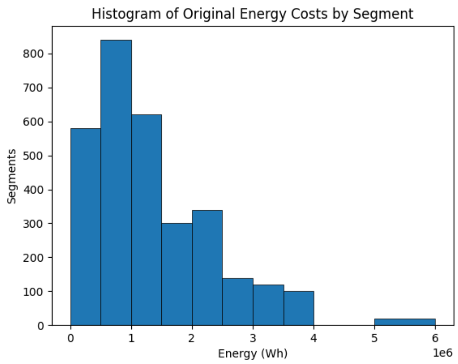
Method	Starting Opt Gap	Final Opt Gap	Starting Obj Value	Final Obj Value
Base Case (no warm start)	287.02%	7.19%	3.92E+10	1.42E+11
Conservative Cost Matrix	12.45%	8.36%	1.35E+11	1.40E+11
Decomposition by UAV Type	15.50%	3.47%	1.31E+11	1.47E+11
Decomposition by Individual UAVs	3.47%	3.47%	1.47E+11	1.47E+11

4.6 Varying Weather Data

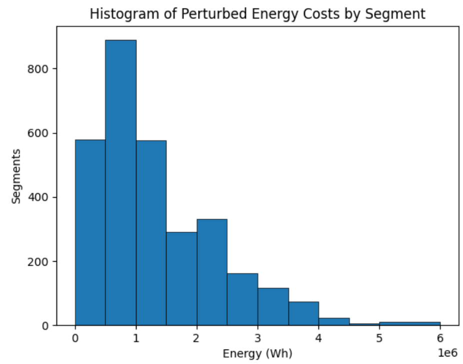
Having compared the various warm start methodologies using the same weather data as Won (2022), we now perturb the data to test how our warm start techniques react to weather forecasts with a higher variance. The initial data used has very low variance of energy costs over flight segments for different ensemble members. The maximum standard deviation for any segment across all ten ensemble members is 3.72E-9. Although we would not typically expect large variance between ensemble members, this value is low. Due to the low variance, we test ensembles with greater variance by simulating our own.

Choosing a fixed seed of 888 to support reproduction of our results, we create our simulated data by calculating the mean energy requirement for each flight segment across all ten of its ensemble members in the original data. We then pull a random number generated from a normal distribution with the mean equal to the flight segment mean and the standard deviation equal to 10% of the flight segment mean. With this new data we now have flight segments that vary significantly more in energy requirements across ensemble members. We also ensure all segments have positive energy requirements across the network before using them in our models.

Figure 4.2 displays the distribution of energy costs among all segments and ensemble members from both the original and simulated data. As the figure indicates, we replicate the energy distribution quite well despite our increase in variance among ensemble members. Table 4.8 shows the optimality gaps and solve times of our varying solve methods when using the new perturbed data. We observe very similar results as our original data with some marginal improvement in solve time. We also see that our decomposition by UAV type still struggles when compared to our individual UAV decomposition method.



(a) energy by segment from original weather data.



(b) energy by segment from perturbed weather data.

Figure 4.2. Histograms of energy costs for original and perturbed data.

Table 4.8. Perturbed data objective and solution time comparison.

Method	Starting Opt Gap	Final Opt Gap	Starting Obj Value	Final Obj Value	Computation Time (sec)
Base Case (no warm start)	227.47%	7.97%	5.71E+10	1.73E+11	1,216
Conservative Cost Matrix	3.47%	3.47%	1.79E+11	1.79E+11	462
Decomposition by UAV Type	20.16%	7.19%	1.64E+11	1.84E+11	971
Decomposition by Individual UAVs	3.47%	3.47%	1.85E+11	1.85E+11	25

CHAPTER 5:
Conclusions and Future Work

Throughout this thesis we show improved solve time of a complex MDVRP using various warm start techniques. We demonstrate that by using warm starts we develop through either solving a conservative solution to a problem instance or decomposing and solving smaller pieces of a larger instance, we significantly improve computation time. We have specifically improved computation time from 1,320 seconds for the initial model to 660 seconds using our conservative cost matrix as a warm start and to 22 seconds using our decomposition techniques as a warm start. We can use these techniques to speed up the solve time of more complex scenarios that take too long to solve in support of real-world operations with previous formulations. With these improvements in hand future research may seek to relax various limitations and assumptions to the model to support real world use cases while limiting solve time and enabling day to day operations.

5.1 Load-Dependent Energy Costs

Our current model assumes that the starting weight of a UAV will be its constant weight for the duration of the route, and this starting weight is determined in a conservative manner by assuming the UAV is fully loaded. The model does not take into account the potential energy savings of a UAV as it lightens its load while servicing demand across the network. We recommend future research adjust or reformulate the model to recalculate the energy costs of a UAV each time it services a demand.

5.2 Time-Dependent Energy Costs

The current cost matrix for each (i, j) segment is calculated for a single discrete period in time across the entire network. In reality, the weather changes and thus the energy cost matrix changes as time advances for any single flight segment. We recommend future work account for the time horizon by keeping track of how long a UAV takes to fly its route and calculating the new cost to fly between segments based upon the time that flight occurs.

5.3 Intermediate Supply Depots

The current problem formulation starts each UAV at a depot node carrying its maximum supply capacity at the start of a route. All nodes visited along a route are being serviced and thus remove supply from the UAV until the route terminates at a depot node. There may be utility in future research reformulating the problem to allow staging of additional supply or depot nodes along a route that a UAV could visit to replenish its stores and energy as it continues to service demand nodes. There also may be additional gains by integrating this area of research with the depot selection model developed by Haller (2021).

List of References

Alexander I, Elle E, Mitchell R (2020) Logistics is a shaping factor. *Proceedings* 146(11) (November).

Berger (2022) Force design 2030 annual update, https://www.marines.mil/Portals/1/Docs/Force_Design_2030_Annual_Update_May_2022.pdf.

Engau A (2012) Recent progress in interior-point methods: Cutting-plane algorithms and warm starts. Anjos MF, Lasserre JB, eds., *Handbook on Semidefinite, Conic and Polynomial Optimization* (Springer, New York, NY, USA), 471–498, https://doi.org/10.1007/978-1-4614-0769-0.

Filomena TP, Lejeune MA (2014) Warm-start heuristic for stochastic portfolio optimization with fixed and proportional transaction costs. *Journal of Optimization Theory and Applications* 161(1) (June), https://doi.org/10.1007/s10957-013-0348-y.

Forget N, Gadegaard SL, Nielsen LR (2022) Warm-starting lower bound set computations for branch-and-bound algorithms for multi objective integer linear programs. *European Journal of Operational Research* 302(3) (November), https://doi.org/10.1016/j.ejor.2022.01.047.

Gondzio J (1998) Warm start of the primal-dual method applied in the cutting-plane scheme. *Mathematical Programming* 83(1-3) (January), https://doi.org/10.1007/BF02680554.

Haller JR (2021) Expanding optimization of energy efficient UAV routing in support of Marine Corps expeditionary advanced base operations with multiple supply depots. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/68328.

Headquarters, U.S. Marine Corps (2023) *Tentative Manual for Expeditionary Advanced Base Operations 2nd Edition*. TM EABO. Washington, DC, USA, https://www.marines.mil/Portals/1/Docs/230509-Tentative-Manual-For-Expeditionary-Advanced-Base-Operations-2nd-Edition.pdf?ver=05KvG8wWlhI7uE0amD5uYg%3d%3d.

Jatho A (2020) Optimizing energy efficient UAV routing in support of Marine Corps expeditionary advanced base operations. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/65554.

McCormick BW (2011) *Introduction to flight testing and applied aerodynamics* (American Institute of Aeronautics and Astronautics, Reston, Va.), Proquest.

Pour SM, Drake JH, Ejlertsen LS, Rasmussen KM, Burke EK (2018) A hybrid constraint programming/mixed integer programming framework for the preventive signaling maintenance crew scheduling problem. *European Journal of Operational Research* 269(1) (August), https://doi.org/10.1016/j.ejor.2017.08.033.

Ralphs T, Güzelsoy M (2006) Duality and warm starting in integer programming. *The proceedings of the 2006 NSF design, service, and manufacturing grantees and research conference*, https://coral.ise.lehigh.edu/~ted/files/papers/DMII06.pdf.

Won DD (2022) Robust energy-aware unmanned aerial vehicle routing using ensemble weather forecasts. Master's thesis, Operations Research Department, NPS, Monterey, CA, USA, https://calhoun.nps.edu/handle/10945/70783.

Initial Distribution List

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California