NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

THESIS

OPTIMIZED ROUTING OF UNMANNED AERIAL SYSTEMS FOR THE INTERDICTION OF IMPROVISED EXPLOSIVE DEVICES

by

Daniel N. Reber

September 2007

Thesis Advisor: Johannes O. Royset
Second Reader: Robert F. Dell

Approved for public release; distribution is unlimited

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE				*Form Approved OMB No. 0704-0188*
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.				
1. AGENCY USE ONLY *(Leave blank)*	**2. REPORT DATE** September 2007		**3. REPORT TYPE AND DATES COVERED** Master's Thesis	
4. TITLE AND SUBTITLE Optimized Routing of Unmanned Aerial Systems for the Interdiction of Improvised Explosive Devices				**5. FUNDING NUMBERS**
6. AUTHOR(S) Major Daniel N. Reber, USMC				
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)

As of September 2007, improvised explosive devices (IED) account for 43% of U.S. casualties in Iraq – the largest single cause of death. One reason for their high rate of effectiveness is that they are extremely difficult to detect. This research develops a tool for selecting routes that will best employ unmanned aerial systems (UAS) for the purpose of detecting IED or related activity. We refer to this tool as IED Search Optimization Model (ISOM). ISOM – which uses prediction model results as an underpinning – accounts for factors such as winds, sensor sweep-width, and aircraft de-confliction. We formulate the problem as an Integer Program and optimally solve it to select the best routes. Initial evaluation of ISOM through field experiments with actual UAS suggest that the tool produces realistic routes which can be flown in the expected amount of time. Furthermore, these routes result in a 42% increase in the likelihood of achieving a detection opportunity over searching nodes in a random manner. ISOM could be implemented as a "reach-back" capability with an analyst providing daily routes for tactical operators.

14. SUBJECT TERMS IEDs, IED Detection, IED Interdiction, Vehicle Routing Problem, Integer Program			**15. NUMBER OF PAGES** 74
			16. PRICE CODE
17. SECURITY CLASSIFICATION OF REPORT Unclassified	**18. SECURITY CLASSIFICATION OF THIS PAGE** Unclassified	**19. SECURITY CLASSIFICATION OF ABSTRACT** Unclassified	**20. LIMITATION OF ABSTRACT** 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.

OPTIMIZED ROUTING OF UNMANNED AERIAL SYSTEMS FOR THE INTERDICTION OF IMPROVISED EXPLOSIVE DEVICES

Daniel N. Reber
Major, United States Marine Corps
B.S., Arizona State University, 1996

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN OPERATIONS RESEARCH

from the

NAVAL POSTGRADUATE SCHOOL
September 2007

Author: Daniel N. Reber

Approved by: Johannes O. Royset
Thesis Advisor

Robert F. Dell
Second Reader

James N. Eagle
Chairman, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

As of September 2007, improvised explosive devices (IED) account for 43% of U.S. casualties in Iraq – the largest single cause of death. One reason for their high rate of effectiveness is that they are extremely difficult to detect. This research develops a tool for selecting routes that will best employ unmanned aerial systems (UAS) for the purpose of detecting IED or related activity. We refer to this tool as IED Search Optimization Model (ISOM). ISOM – which uses prediction model results as an underpinning – accounts for factors such as winds, sensor sweep-width, and aircraft de-confliction. We formulate the problem as an Integer Program and optimally solve it to select the best routes. Initial evaluation of ISOM through field experiments with actual UAS suggest that the tool produces realistic routes which can be flown in the expected amount of time. Furthermore, these routes result in a 42% increase in the likelihood of achieving a detection opportunity over searching nodes in a random manner. ISOM could be implemented as a "reach-back" capability with an analyst providing daily routes for tactical operators.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

LIST OF ACRONYMS AND ABBREVIATIONS

AO	Area of Operations
CETSP	Close Enough Traveling Salesman Problem
CPLEX	C Programming Language Simplex Method
COIN	Counterinsurgency Doctrine
CONOPS	Concept of Operations
DoD	Department of Defense
GAMS	General Algebraic Modeling System
HSI	Hyperspectral Imaging
IED	Improvised Explosive Device(s)
IP	Integer Program
ISR	Intelligence Surveillance and Reconnaissance
JHUAPL	Johns Hopkins University Applied Physics Lab
JIEDDO	Joint IED Defeat Organization
JIEDD TF	Joint IED Defeat Task Force
MNC-I	Multi-National Corps – Iraq
MPOPTW	Multi-Player Orienteering Problem with Time-Windows
NGA	National Geospatial-intelligence Agency
OP	Orienteering Problem
OPTCR	Optimality Criteria
TSP	Traveling Salesman Problem
UAS	Unmanned Aerial System(s)
USSOCOM	United States Special Operations Command
VBA	Visual Basic for Applications
VMU-2	Marine Unmanned Aerial Vehicle Squadron 2
VRP	Vehicle Routing Problem
WCA	Wind Correction Angle
WTA	Wind to Track Angle

ACKNOWLEDGMENTS

First, I must thank my loving wife, Amy, whose patience and support throughout this process has been astonishing. I share this success with her and am truly grateful for all that she does for me. In addition, this work would not have been possible had it not been for the support and assistance from many individuals and many opportunities presented to me. My advisor, Professor Johannes Royset has been fantastic. His superb guidance and constant availability has made this process an exceptional learning experience. I would like to thank Mr. Jack Keane of the Johns Hopkins University Applied Physics Laboratory for provided me an opportunity to spend my experience tour with his group at the lab, and Mr. Larry Bulanda who served as my sponsor during that time. My time spent at JHUAPL was instrumental in developing this problem and I am grateful for it. I would also like to thank Dr. Dave Netzer who afforded me a chance to conduct field testing of my model in NPS-SOCCOM field experiments at Camp Roberts and his research assistant Marianna Verett making my experience there a success. This unique opportunity resulted in a great deal of learning which could otherwise never been achieved. Finally I would like to thank Professor Robert Dell for his interest and guidance as second reader, Professor Matthew Carlyle whose Math Programming Seminar was critical this work, Professor Gordon Bradley for his dedication to the IED problem and keen insight regarding it, and Professor Moshe Kress for his contributions throughout this process.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

This thesis develops a routing tool for Unmanned Aerial Systems (UAS) tasked with interdiction of Improvised Explosive Devices (IED) that we refer to as IED Search Optimization Model (ISOM). ISOM uses optimization models and algorithms to leverage recent developments in IED prediction as well as emerging UAS and sensor technology. Tactical level operators can use ISOM to determine routes that will best employ their UAS for the purpose of detecting IED or IED related activity.

ISOM receives output from an existing IED prediction model and uses it to establish relative values for searching various portions within a sector of operation. In the type of operations considered, there are a small number of UAS available, each with a search time determined by its fuel load. To model interdiction of IED with the aid of a prediction model, we discretize the space around roadways within the sector of operation into 200 meter by 200 meter square cells. We then use the prediction model to assign values that represent the likelihood of an IED event occurring in the respective cell the following day and treat these values as rewards which can be achieved by a UAS after it has searched the respective cells. Next, we reduce the number of cells to include only those above a threshold which will reduce the size of the problem without any significant reduction in solution quality. We then establish nodes at the center of each remaining cell that serve as a network of UAS waypoints with supplementary nodes added to represent UAS operating bases. Next, we discretize the search period into time-steps, and calculate travel times between each pair of nodes – rounded up to the next integer time-step. These travel times represent a "cost" incurred by a UAS in order to achieve the associated reward. The output of the model is a set of routes which focus UAS search efforts in the most likely areas of IED occurrence and result in the maximum number of detection opportunities within the search time allowed by the UAS endurance.

ISOM accounts for factors such as winds, sensor sweep-width, and aircraft de-confliction. Winds have a significant influence on UAS operations because they tend to operate at low speeds and can potentially be met with winds in excess of their capabilities. ISOM receives wind measurements and/or forecasts as input and calculates

groundspeeds based on wind speed, wind direction, desired course, and UAS airspeed. Travel times are then determined based on groundspeed and distance between nodes resulting in routes that are achievable in prevailing wind conditions. Sensor sweep-width establishes a limit to the coverage of cells adjacent to each other. ISOM incorporates rules for clustering nodes according to given sensor sweep-widths resulting in efficient coverage and fewer turns for the UAS. We ensure de-confliction of UAS operating within a limited block of altitude by ensuring that UAS maintain a minimum lateral distance between one another. In addition to addressing these factors, ISOM achieves a moderate restriction of the problem by identifying arcs whose removal result in little or no impact on the optimal value.

ISOM is a combination of an Excel spreadsheet designed to receive user inputs and perform preprocessing with the use of macros written in Visual Basic for Applications, and an integer programming model implemented in the General Algebraic Modeling System (GAMS) to find optimal or near-optimal solutions using commercially available optimization software. The model receives various input parameters including UAS airspeed and sweep-width, observed wind direction and speed, as well as criteria for restricting the integer programming model for the purpose of speeding the solution time.

We accomplish testing of ISOM at Camp Roberts, California during exercises conducted through the NPS-USSOCOM Cooperative Field Experimentation Program. In the absence of available real-world data including a history of IED events and related geographic features, we generate data to mimic prediction model output over the Camp Roberts area for these experiments. For a comparison with current methods, we present the same route planning scenario to a group of experienced UAS operators currently deployed to Iraq. While we await the results of this exercise at the time of this summary, preliminary examination reveals that the model produces on average a 75% reduction in problem size while obtaining solutions within 2% of the optimal value of a base case. It produces routes which achieve a 42% increase in the likelihood of achieving a detection opportunity over searching the nodes at random. ISOM could be utilized as a reach back capability with an analyst providing daily routes via the internet.

I. INTRODUCTION

A. BACKGROUND

As of September 8, 2007, improvised explosive devices (IED) have accounted for 1,609 of the 3,754 total confirmed U.S. fatalities in Iraq – 43% (iCasualties.org 2007). On June 27, 2005, Deputy Secretary of Defense Gordon England issued Department of Defense (DoD) Directive 2000.19, which established the Joint IED Defeat Task Force (JIEDD TF) and designated it as the focal point for all efforts in the DoD to defeat IEDs (DoD 2005). Later renamed Joint IED Defeat Organization (JIEDDO), the group has presented a three-part strategy to solving the problem: "Attack the Network," "Defeat the Device," and "Training the Force" (DoD 2006). The first part of this strategy deals with preventing the emplacement of IEDs by attacking enemy vulnerabilities at multiple points in the IED system. The second deals with defeating the device once it is emplaced and the third is to facilitate the establishment and growth of coalition and partner nation counter-IED capability by training the forces to implement the first two.

Beyond this strategy, the response to the IED problem has been further divided into five lanes: predict, prevent, detect, neutralize, and mitigate. Of these, detection has been the most challenging part of the IED problem and thus far Intelligence, Surveillance, and Reconnaissance (ISR) assets have been ineffective in this effort. In fact, most IED detections have been made visually by operators on the ground. According to Christine DeVries, a spokesperson for JIEDDO, "The best sensor we have for detecting an IED is an individual soldier's or Marine's eyes" (Chisholm 2005).

> IEDs are notorious for being difficult to detect and neutralize. IEDs can be packaged in myriad objects, such as burlap sacks, trash, toys, dead animal carcasses, buckets or cinder blocks. They can be attached to telephone poles, be placed in guardrails or buried under the road. IEDs can even be packaged to look like a concrete roadside curb.
>
> (Chisholm 2005)

A consequence of the previous statements is that visual detection does not usually occur until after one is already in range of the threat. Keeping in mind that IEDs in Iraq have been created with estimated kill zones of up to 125 meters and that coalition vehicles commonly travel at speeds greater than 20 mph in order to reduce exposure to other threats, consider the following example: Assume that a small IED has a kill radius of 50 meters, perception-reaction time is 3.0 seconds, and breaking distance of 14 meters. Then, a vehicle traveling at 20 mph would need to recognize the IED at a distance of 90 meters in order to remain clear of its kill zone. With litter and debris scattered across the roads of Iraq, relying on the human eye for IED detection makes for risky and tedious operations.

During the course of the conflict in Iraq, the U.S. military has taken several reactive measures to mitigate the IED threat including: up-armoring vehicles, fielding jammers to protect against radio controlled IEDs, and the ad-hoc use of airborne sensors to search for emplaced IEDs. As of June 2007, the military has spent more than $4 billion to devise tactics, armaments and technological means to defeat these bombs. One logical measure of effectiveness of these efforts is the rate of success of IEDs inflicting casualties over the course of time as depicted in Figure 1. As this data shows, the current rate of success of these devices has fallen to one sixth that of June 2003. Despite this fact, overall casualty rates due to IEDs have continued to increase as depicted in Figure 2. The reason for this is two-fold. First, the insurgency has simply increasing the volume of devices that they emplace in order to increase their overall effect. And second, they have adapted quickly by developing more effective IEDs and adjusting their tactics to mitigate ours.

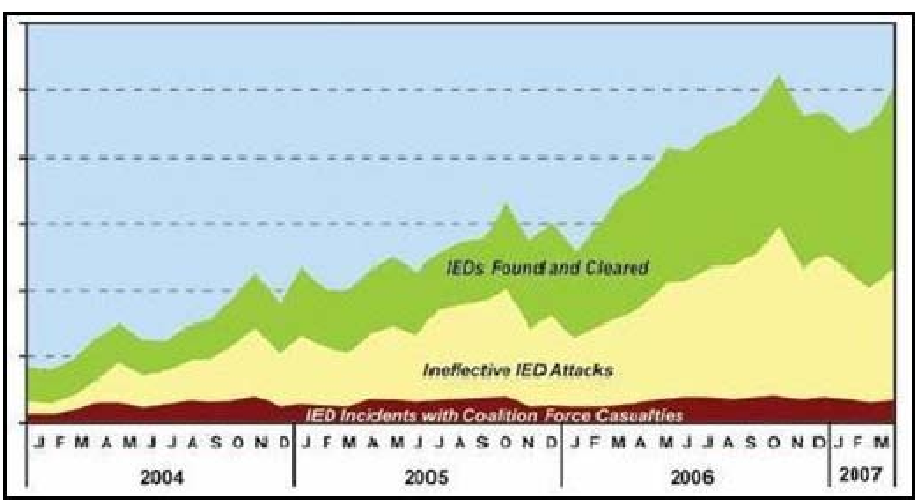


Figure 1. Iraq IED incident trends (From Schachtman 2007).

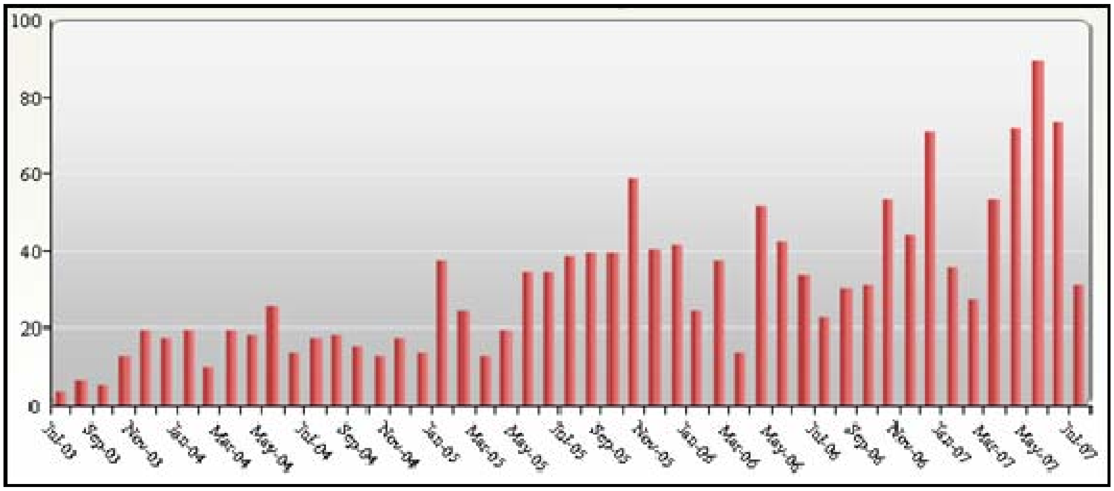


Figure 2. Record of U.S. fatalities in Iraq due to IEDs (From iCasualties.org 2007).

As a result of the continued effectiveness of these devices, there has been much debate in recent months over the effectiveness of the IED defeat effort including the use of UAS and other airborne assets to counter them. During a PBS Newshour interview by journalist Ray Suarez on June 21, 2007, General Montgomery Meigs, head of JIEDDO, is quoted as saying "We have improved in less than a year the UAS -- unmanned aerial vehicle -- profile in country. It's had a major impact on both going after networks and placers" (Suarez, 2007). In seeming contrast to this statement, General Ronald Keys, commander of the Air Force's Air Combat Command offered criticism of the use of airborne assets to find IEDs in an article published the same day. According to Keys, the number of IEDs found by UAS, surveillance aircraft or combat jets outfitted with advanced targeting pods per flight hour is very low. Keys elaborates by citing that airborne assets spend much time flying up and down streets that no coalition forces will be on for another 12 hours and also compares the area of operations (AO) in Iraq to a junkyard, with too many false positives (Fabey, 2007).

In light of the continuing IED problem, many experts have advocated that the only effective strategy to counter IEDs is one that focuses not on finding IEDs that are already placed but by attacking the network that produces the devices – finding out where the explosives come from and who the bomb makers are. With a recent change in leadership at Multi-National Corps – Iraq (MNC-I), a new emphasis has been placed on the use of counterinsurgency doctrine commonly referred to as "COIN." In a June 2007 memo, Lieutenant General Raymond T. Odierno provided counterinsurgency guidance with ten, mutually reinforcing principles that he expects to be "operationalized" in Iraq. One of these principles, which relates to the debate discussed above, is to look beyond the IED and to get the network that placed it. General Odierno elaborates by stating that commanders should map IED patterns and use friendly movements to trigger enemy action and coordinate reconnaissance assets to identify IED teams moving into position. Furthermore, he states that UAS should be used to trace enemy firing teams back to their caches and assembly areas (Odierno, 2007).

B. RESEARCH GOAL

Based on the casualty numbers cited earlier, we can see that IEDs have had a significant impact in the conflict in Iraq and there is no reason to expect that they will not continue to play a similar role in future conflicts against insurgencies. While we acknowledge that the COIN based plan for attacking the network behind these devices will likely have the largest impact in a counter IED strategy, we believe that a comprehensive approach is still necessary if we are to mitigate this threat to the maximum extent possible. Additionally, we question whether the lack of effectiveness of airborne sensors to date is due to an inherent inability, or if the methods of employing these assets have been inefficient and uncoordinated.

In order to minimize the effectiveness of the IED threat, we believe that a method for improving the probability of success of ISR assets in detecting IEDs and hence minimizing risk to troops should continue to be explored. And while there is skepticism regarding IED detection with airborne sensors, industry continues to pursue these technologies and develop new sensors. A concurrent effort to develop a concept of operations (CONOPS) for such a sensor not only makes sense from a standpoint of maximizing their effectiveness at initial fielding, but may also help prove their effectiveness in the first place. We believe that an effective surveillance strategy which focuses on the most likely locations of IED emplacement, closely coordinated with blue force activities will bear fruit not only in detecting IEDs already emplaced, but more importantly, catching insurgents in the act. Furthermore, although the counterinsurgency operations aimed at attacking the network rely heavily on human intelligence, each successful detection provides a window into the network behind it – particularly if insurgents are caught in the act. Therefore, we believe that such an effort is not at odds with the COIN guidance recently provided by MNC-I, but rather in-line with it.

This thesis develops a routing tool that leverages recent developments in IED defeat efforts and optimizes the employment of UAS and emerging sensor technology for the interdiction of IED. We define interdiction as those actions taken to confront and halt the activities leading to IED incidents and state our intent as to maximize the number of detection opportunities presented to UAS sensors including situations where insurgents are in the act of emplacing the IED as well as detection of IED which are already in place. Furthermore, we refer to the tool we develop for this purpose as IED Search Optimization Model (ISOM). We accept that vehicle borne IED and associated activities are not likely to be detected by UAS assets and rule them out as potential targets in our model. Some recent advancement that is fundamental to our work is the development of IED prediction models. Our analysis uses output from IED prediction models (developed as a guide to concentrate search efforts) and established vehicle routing models as a starting point in the formulation of a new model for the optimization of UAS routing specific to IED interdiction. Our study considers a heterogeneous fleet (2-3 types) of UAS and focuses on one effort per Air Tasking Order cycle – one sortie per UAS involved. Aspects specifically not addressed include dynamic re-tasking, and factors affecting aircraft availability such as maintenance and other logistical concerns.

C. LIMITATIONS

Ostensibly, the UAS that would be ideally suited for the types of operations we are studying are the larger UAS such as Pioneer and Predator. However, it is not practical to expect these types of UAS to be dedicated to a study such as ours so we consider the use of smaller UAS for a low level proof of concept. Fortunately, the Naval Postgraduate School and the United States Special Operations Command (USSOCOM) have been conducting a series of experiments known as the NPS-USSOCOM

Cooperative Field Experimentation Program. These exercises which are conducted periodically at Camp Roberts, California involve several small UAS and present an ideal opportunity to test ISOM which may then be scaled up to represent larger UAS.

One disadvantage to using smaller UAS is that they are not capable of carrying the most advanced sensors available. However, the intent of our study is to maximize the number of detection opportunities that occur based on route selection, not on sensor development. Therefore, we rationalize that, for the purpose of our study, the use of small UAS equipped with simple electro-optical sensors can be used to imitate larger UAS equipped with sophisticated sensors. Furthermore, as discussed in Chapter II, micro UAS have had some success in detecting IED and in fact, at the time of this writing, there are new types being deployed to Iraq for the specific purpose of identifying IEDs (Rosenberg, 2007). Depending on their effectiveness, it may be the case that these vehicles, routed optimally, may turn out to be the ideal instrument for IED interdiction.

D. ASSUMPTIONS

While our primary focus is to make efficient use of given assets we present an overview of emerging sensor technology in Chapter II. We do this in order to consider some sensors which may be ideally suited for IED detection in conjunction with our model. A key assumption then is that either (i) a sensor which is effective at identifying IEDs and designed for use with larger UAS will emerge or (ii) micro UAS will prove effective in identifying IEDs or IED activity. In any case we proceed as though our sensors have perfect detection capability and focus on presenting the maximum number of detection opportunities to them. Furthermore, we begin our study by assuming that IED are static events which last for indefinite lengths of time and develop a simple model to represent this case. After establishing this foundation we then extend the model to treat IED events as having time windows which are defined by blue force locations and activity.

E. STRUCTURE OF THESIS AND CHAPTER OUTLINE

This thesis is organized into five chapters including the Introduction. Chapter II provides a review of literature on IED defeat efforts and UAS employment as well as a thorough background on the field of Vehicle Routing Problems (VRP). We present the formulation of our Integer Program (IP) model in Chapter III after a discussion of development considerations. In Chapter IV, we present results and analysis based on several experiments conducted during two field exercises at Camp Roberts, California. Chapter V summarizes the research, presents the main findings and insights, and discusses the potential future work.

THIS PAGE INTENTIONALLY LEFT BLANK

II. LITERATURE REVIEW

A. IED DEFEAT EFFORTS

1. Sensors

Between 2003 and 2005, DoD spent more than $375 million on the IED defeat effort (Chisholm 2005) and in January 2006, JIEDDO was allocated a $3-billion budget to develop counter-IED technology (Levine 2006). With this vast anti-IED research and development effort underway, there has been significant progress in several areas of technology. One area of technological development that is of particular relevance to our study is in the field of airborne sensors; and while there is some skepticism of airborne sensors purported to be effective at detecting IEDs (Trimble 2006), industry has continued to pursue this effort. Some types of sensors that have been seriously considered for IED detection are *change detection, multispectral imaging* and *hyperspectral imaging* (HSI).

Change detection is a system which detects changes in an environment by comparing two, high-resolution photograph mosaics. The Marine Corps Warfighting Lab conducted a limited technical assessment of one change detection system known as Airborne Volumetric Change Detection System in May 2005. This assessment concluded that Airborne Volumetric Change Detection System was impractical for operational use in detecting IEDs due to excessive false alarm rates and lengthy processing time. (Funkhouser 2006)

Multispectral imaging is the science of taking multiple images at different (disjoint) parts of the spectrum. By selecting the proper wavelengths, many militarily important items such as camouflage, thermal emissions and hazardous waste can be detected (Pike and Aftergood 2006). HSI takes this concept a step further by creating a larger number of higher resolution images from a contiguous part of the spectrum. This creates a great increase in information, which allows the use of methods such as *bloodhounding* (i.e. detection of unique spectral images) and *anomaly detection* (i.e. detection of an object that is different from its surroundings including disturbed earth), as well as *change detection*.

> The US Army is preparing to assess the potential of [HSI] payloads for UAS, which could create a new class of surveillance technology and one that may be ideally suited for the task of detecting IEDs…" According to Bob D'Amico of BAE Systems: "The promise of hyperspectral [technology] has not really come until now... This is the beginning of it. We think the technology is ripe and it's ready to go, regardless of the reputation.
>
> (Trimble 2006)

An example of HSI is presented in Figure 3. The left image is a conventional three-color image (printed in grey scale) of the area. Mine widths are roughly equal to a pixel width and the mines are invisible on the full color image. In the HSI image (right half of the picture), mines are clearly depicted in white (McFee and Ripley 1997).

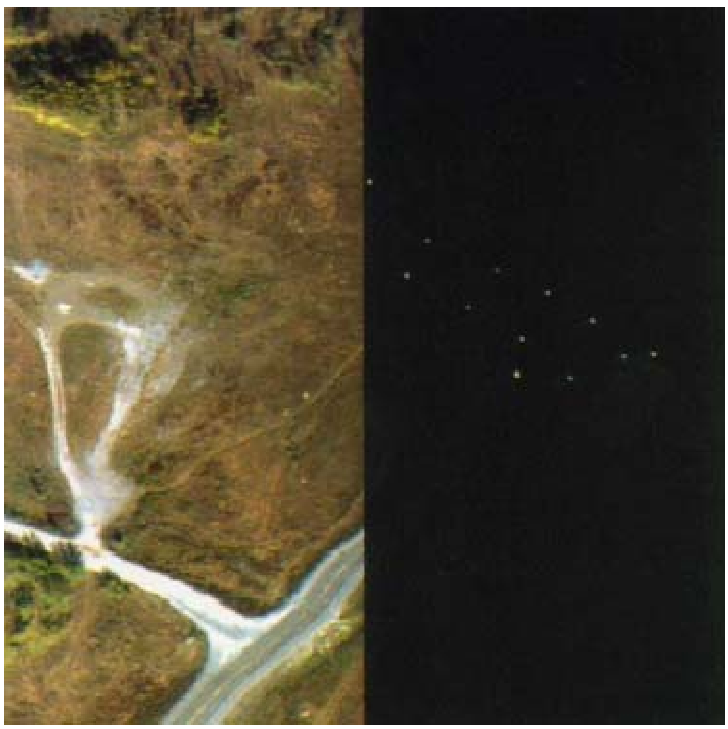


Figure 3. Surface-laid mine detections (From McFee and Ripley 1997).

One important factor influencing the effectiveness of these sensors in detecting IEDs is the manner in which they have been employed. While no formal CONOPS has been formulated for these types of sensors, during limited technical assessments they have been employed using an area search, sweeping over every square inch of a large area rather than focusing on key hot spots. This leads us to another important area of the

IED defeat effort – prediction – and we provide a more thorough discussion of CONOPS and limited technical assessments in Section B.

2. Prediction Models

At least two different models have been developed to predict the placement of IEDs in Iraq. Riese (2006) developed a tool known as *Threat Mapper*, a work in progress that provides a spatial predictive capability against IEDs and other threats. Now employed by the Johns Hopkins University Applied Physics Lab (JHUAPL), Riese did the bulk of this work while on active duty and assigned to U.S Strategic Command. An example of *Threat Mapper* output is presented in Figure 4. From this figure we can identify areas of various levels of threat which are depicted using color variation in a manner similar to a topographical map.

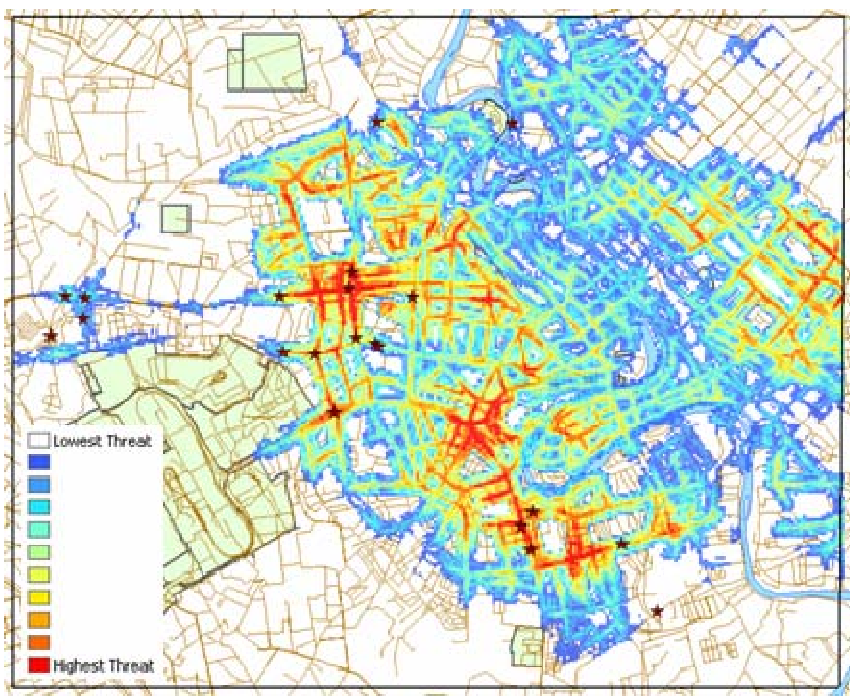


Figure 4. Example output from *Threat Mapper* (From Riese 2006).

Additionally, Lantz (2006) developed a spatio-temporal model to forecast likely locations of IED emplacement and has applied this model to real, classified IED data assembled from several diverse sources. These models discretize the area of interest into a grid of square cells and consider only cells within a given distance of known roads.

They then assign a value to each cell, which represents the likelihood of an IED attack in that cell based on various model inputs such as historical IED events, infrastructure and geography, and coalition force activity. The output is a map graphically depicting the likelihood of IEDs throughout the region on the next day. These models, while imperfect, have shown promise as decision aids for commanders in Iraq in planning convoys, patrols, and other ground operations (Lantz 2006 and Riese 2006).

B. UAS EMPLOYMENT

UAS are broken down into a tiered classification system which we depict in Figure 5. The primary missions of UAS in Operation IRAQI FREEDOM that fall in the Tier 1 and above categories are point surveillance, target following, area search, route reconnaissance, and IED detection (Owen, et al. 2005). Missions in Iraq flown by Pioneer, Scan Eagle, and Shadow UAS usually service a list of targets given to them by higher intelligence units. These target lists include various sites such as suspected insurgent safe houses, suspected weapons caches and mortar points of origin as well as direct support for raids, patrols, convoys and other operations. Based on the author's experience while serving with Marine Unmanned Aerial Vehicle Squadron 2 (VMU-2) and deployed to Al Taqaddum Airbase in Iraq during 2004 and again in 2005, routing for each mission is done manually by UAS operators and often these UAS respond to dynamic re-tasking for unpredicted events that occur during the mission: IED attacks, ambushes, patrols, downed aircraft, etc. Anytime that a UAS is not actively engaged servicing a target list or responding to dynamic re-tasking, is usually spent arbitrarily sweeping roads for any unusual activity or other signs which might lead to detection of IEDs.

Although both the Army and Marine Corps have conducted limited technical assessments of various Tier 1 and 2 type sensors for IED detection, no formal CONOPS have been developed for employment of such sensors. These assessments have generally used an area search approach, covering a large area with a "lawnmower search" to generate a graphic mosaic of the area for change detection comparison (Owen, et al. 2005).

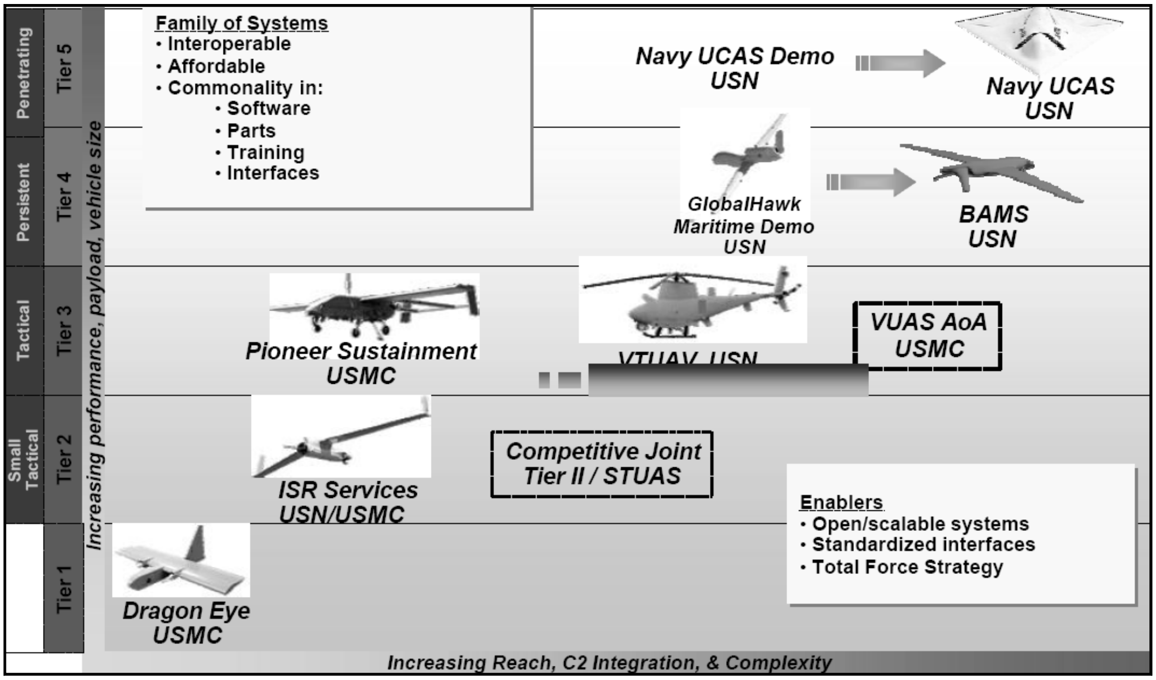


Figure 5. Naval UAS Family of Systems (From Kelly 2006).

Another possibility to consider is the use of small UAS which can weigh less than 30 pounds and are equipped with simple electro-optic and infrared cameras. One such vehicle produced by Honeywell Corporation known as the Micro Air Vehicle is currently being deployed to Iraq specifically for detection of IEDs. This circular vehicle, which is just 13 inches in diameter and weighs only 16 pounds, will be the first ducted-fan UAS to be used in combat operations. Small enough to be carried in a backpack, it operates like a helicopter and can fly down and hover at low altitudes to investigate possible threats. A UAS such as this, if proven capable of detecting IEDs, could certainly benefit from a well developed, optimization based, concept of operations (Rosenberg, 2007).

And while optimization models have been developed for various UAS missions including special operations (Kress and Royset 2007) and tactical reconnaissance missions in general (Moser 1990), the only optimization based method for employing UAS specifically for IED detection that the author is aware of is Jones (2006). The stated goal of Jones' thesis is "to develop models to determine the optimal allocation of sensor resources to search for IEDs along a specified road sector, where a road *sector* comprises the entire space to be searched." Jones proceeds by formulating an IP to maximize the expected number of segments "determined" with a finite number of looks *N* allocated *a priori* based on UAS availability. In this model, a segment is considered to be "determined" if there is strong evidence of either the presence or absence of an IED in the segment. Jones (2006) assumes independence between segments and recommends further consideration of the case where IED prior probabilities are correlated. An example given involves a spatial process that describes the prior belief on the presence or absence of IEDs in the road segments which is then used as a basis from which to allocate sensors.

C. VEHICLE ROUTING PROBLEMS

1. Background

The Vehicle Routing Problem (VRP) is a broad class of combinatorial optimization problems, which have been studied since the 1800's. A general description of these types of problems which is accepted by many researchers is provided by the following excerpt.

> The distribution of goods concerns the service, in a given time period, of a set of customers by a set of *vehicles*, which are located in one or more *depots*, are operated by a set of crews (*drivers*), and perform their movements by using an appropriate *road network*. In particular, the solution of a VRP calls for the determination of a set of *routes*, each performed by a single vehicle that starts and ends at its own depot, such that all the requirements of the customers are fulfilled, all the *operational constraints* are satisfied, and the global *transportation cost* is minimized.
>
> (Toth and Vigo 2002)

In its simplest form, this problem considers a fleet of vehicles of uniform capacity while more complex versions consider heterogeneous fleets of vehicles. A specific case of the VRP where only one vehicle is available at the depot and no additional operational constraints are imposed is commonly referred to as the Traveling Salesman Problem (TSP) (Toth and Vigo 2002). An extension of the TSP which is of particular relevance to our study is the Orienteering Problem (OP) in which the mandatory visit requirement is eliminated and a prize is associated with each customer. In this instance, the travel cost objective is stated as a constraint and the aim is to find a route that maximizes collected profit while ensuring that a predetermined maximum travel cost is not exceeded (Feillet, et al. 2005). It is important to note that some authors make no distinction between the OP and other named problems such at the Prize Collecting Traveling Salesman Problem and the Selective Traveling Salesman Problem while others note subtle differences. An additional constraint that can be added to any of these problems is time windows. In this case the visit to each customer must occur within a specified time window specific to that customer in order to collect a reward.

2. Multi-Player Orienteering Problem with Time-Windows

Moser (1990) develops a scheduling and routing tool for aerial reconnaissance vehicles. In his thesis, he considers the following characteristics:

- A set of targets N that includes a depot (designated as target number one).
- A nonnegative cost c_{ij} to travel between each pair of targets (i,j) is the distance between target i and target j.
- Associated with each target i is a nonnegative point value p_i (except for the depot which is assigned $p_1 = 0$).
- Each target has a service time, $s_{i,}$ which is the time required for a vehicle to service that target.
- The travel costs between targets can vary between vehicles but service times remain the same regardless of the vehicle servicing the target.
- Each target has a target window defined by the earliest (e_i) and latest (l_i) times that service may begin for that target.

Moser refers to this problem as the Multi-Player Orienteering Problem with Time-Windows (MPOPTW) that he formulates into an integer linear program and solves using both optimization as well as heuristic approaches. As a result of his model, Moser offers a planning tool capable of accepting various target and vehicle data as well as commander's guidance regarding target priority and produces a good solution to the routing and scheduling problem for these vehicles. Figure 6 depicts an example solution taken from Moser (1990). In this figure, each dot represents a potential ISR target (customer) having a specific reward and service time. Near optimal routes are depicted for two vehicles operating from a common depot which allows for the collection of a maximum reward while meeting a time constraint determined by each aircraft's fuel supply.

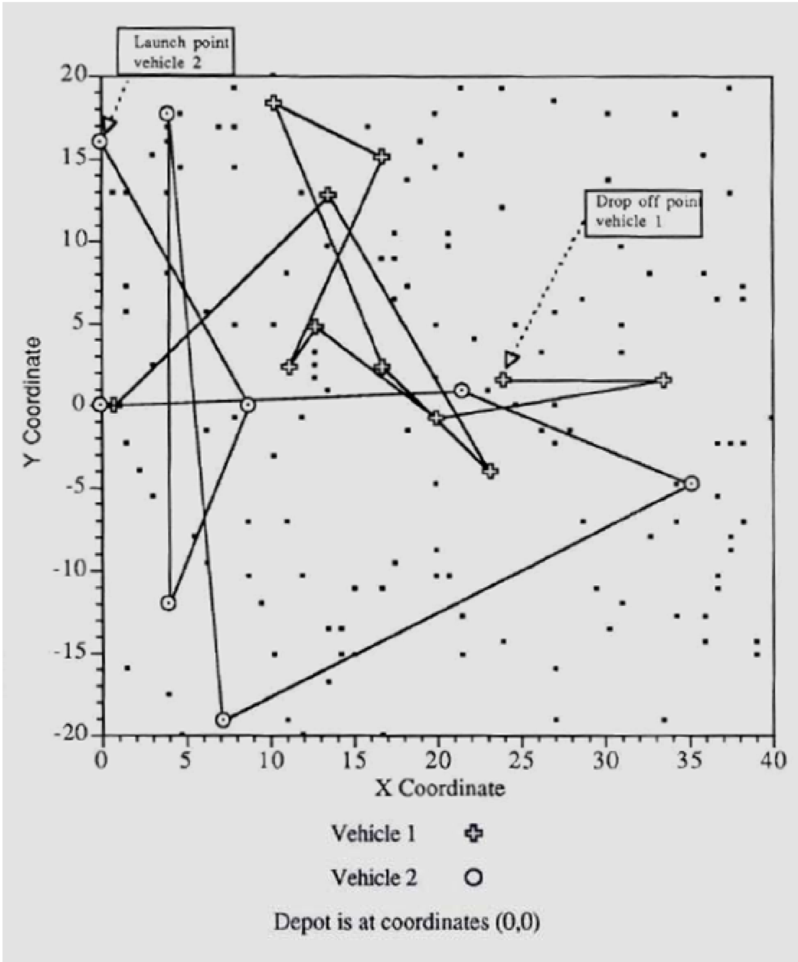


Figure 6. MPOPTW example problem (From Moser 1990).

3. Close Enough Traveling Salesman Problem

Golden and Wasil (2006) introduce the Close Enough Traveling Salesman Problem (CETSP). In this problem, a set of "supernodes" is created such that each customer is within a given distance of at least one supernode. The problem is then solved over this set of supernodes that allow the vehicles to get within a distance determined to be close enough to service each customer. A specific example given by Golden and Wasil is that of meter readers equipped with Radio Frequency Identification Technology.

Until recently, meter readers were required to visit each customer in order to manually read their meter. With Radio Frequency Identification Technology, readers are now only required to get within a given distance that allows them to remotely read each meter, allowing a CETSP approach. Figure 7 depicts an example solution taken from Golden and Wasil (2006).

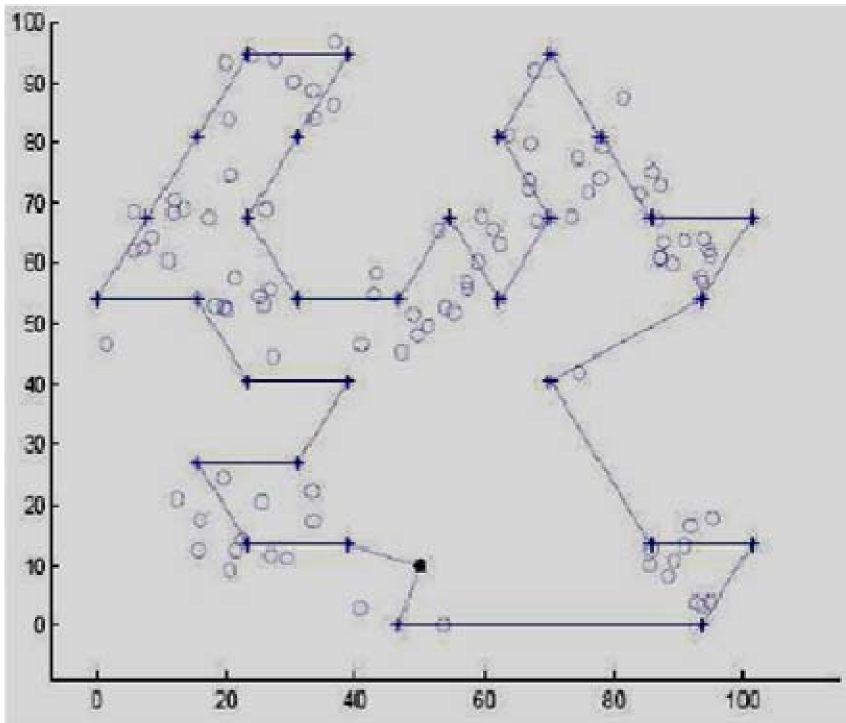


Figure 7. CETSP example solution (From Golden and Wasil 2006).

In this example, the small circles represent customers while the crosses represent supernodes. The route depicted allows the vehicle to get within a distance determined to be close enough to each customer and hence allows it to service every one.

D. LITERATURE REVIEW CONCLUSIONS

Based on our review of emerging UAS and sensor technology coupled with our appraisal of UAS employment, we reason that there is sufficient cause to pursue a model which would optimize the use of these assets as the basis of a structured concept of operations. Furthermore, our review of related VRPs revealed no model currently in existence which is ideally suited for the type of problem we consider. While Jones

(2006) offers a model which optimizes the allocation of assets on a grand scale by dividing them between sectors, our goal is to take this idea a step further by determining optimal routes within each sector.

III. MODEL DEVELOPMENT AND RESTRICTIONS

In this chapter, we consider the IED interdiction problem as a VRP and present a formulation. We begin by considering the simple case where IED behave as static targets which last for indefinite periods of time and proceed by using the MPOPTW model as a starting point in the formulation of a new model for the optimization of UAS routing – specific to IED interdiction – minus the time windows. Subsequently, we consider the complexity of our problem and several restrictions to it. Additionally, we address the impact of weather effects on UAS and introduce a method to account for winds in our model. We then present a preprocessing approach using Visual Basic for Applications (VBA) and Microsoft Excel which greatly reduces the size of our model by applying a moderate restriction to the problem, and implement the method of accounting for wind effects (Walkenbach, 2004). Next, we consider the case where multiple aircraft are operating in close proximity, and require positive separation. For this case, we present additional constraints which impose explicit de-confliction of aircraft. We finish by examining the more realistic case where blue force activity triggers IED events. These events present fleeting windows of opportunity and can occur only during periods of time determined by blue force locations. We consider the possibility of combining prediction model output with information regarding blue force activity to establish time windows for the nodes in our network and we present additional constraints to account for these.

A. MODEL DEVELOPMENT

We formulate the IED interdiction problem as an IP, in the following manner. Let U be a set of UAS dedicated to the IED interdiction effort for which there is a search period determined by aircraft endurance. The search area is discretized and reduced to a finite number I of suitably sized, square cells, each containing a portion of the road network within our AO. Cells between 50 and 800 meters square are recommended for use in the IED prediction model *Threat Mapper* (Riese 2006), while Lantz (2006) used 200-meter square cells in his model. Therefore, cells of size 200 meters square satisfy both models discussed and reasonably smaller than the sweep widths of sensors to be studied. These cells i can be treated as a network of nodes, each of which has been assigned a nonnegative, relative point value by a prediction model that represents its likelihood of containing an IED event on a given day. For the purpose of our model, we consider these assigned values as rewards p_i which are obtained when a UAS searches a respective node i; the total reward collected representing the expected number of IEDs detected. We add to this network additional nodes representing our UAS depots that are assigned $p_i = 0$. We let c_{iju} be the travel time between i and j for UAS u. As with the MPOPTW and other VRPs, we discretize the search period into T time-steps $(t \in \{1, 2, \ldots, T\})$ and set c_{iju} equal to the number of time-steps required – rounded up to the next integer value – for UAS u to travel from i to j.

One complication of our study is the decision of whether a cell may be searched multiple times, and if so how often and at what reward? The problem here is that vehicles permitted to search nodes multiple times may simply loop over the same nodes for extended periods picking up rewards which are not representative of the actual situation. Furthermore, the prediction models which describe our situation, assign values on a *daily* basis making rewards for multiple visits difficult to assign. Therefore, we begin by considering the simple case where IED are placed at night and we employ our sensor in the early morning, before the IED are to be used in an attack. The IED events in this case become static targets with no time windows associated and, assuming a perfect sensor, there is no value in searching a cell multiple times. An additional concern however, is that vehicles not permitted to revisit nodes in a network may be forced to use suboptimal paths to continue a search and should be permitted to revisit nodes for transit purposes only. In answer to these complications, we let a be a set of actions which can be performed by a UAS at each node – search or transit – and limit the number of times a node is searched to one. In Section E, we consider additional factors which potentially provide a basis from which to assign time windows to IED events and a way to account for them in our model. For the time being, we complete the formulation with a binary decision variable which represents whether or not UAS u performs action a at node i and travels from node i to node j during time-step t, given by $X_{i,j,t,u,a}$.

The formulation of our model is as follows.

Indices

i,j	Nodes, $i,j \in \{1, 2, ..., I\}$.
u	UAS, $u \in \{1, 2, ..., U\}$.
t	Time-steps, $t \in \{1, 2, ..., T\}$.
a	Action to be performed at a node (‘search’ or ‘transit’).

Sets

$F(i)$	Forward star of node i.
$R(i)$	Reverse star of node i.

Data

p_i	Probability of IED at node i.
d_u	Node which serves as depot of UAS u.
c_{iju}	Travel time from node i to j for UAS u, rounded up to the nearest time-step.

Binary Variables

$X_{i,j,t,u,a}$	1 if UAS u travels from node i to node j during time-step t, and performs action a at node i, 0 otherwise.

Mathematical Formulation

$$\max \sum_{i,j,t,u} p_i X_{i,j,t,u,'search'} \qquad \text{(Objective function)} \quad (1)$$

s.t.

$$\sum_{j \in F(d_u),a} X_{d_u,j,1,u,a} = 1 \quad \forall u \qquad \text{(Start at depot)} \quad (2)$$

$$\sum_{j \in R(i),t':\, t'=t-c_{jiu} \geq 1,a} X_{j,i,t',u,a} = \sum_{j \in F(i),a} X_{i,j,t,u,a} \quad \forall u, i \neq d_u, t \text{ s.t.} \max_{j \in R(i)} t - c_{jiu} \geq 1 \quad \text{(Balance of Flow)} \quad (3)$$

$$\sum_{j \in F(i),t,u} X_{i,j,t,u,'search'} \leq 1 \quad \forall i \qquad \text{(Prevent duplicate rewards)} \quad (4)$$

$$\sum_{i \in R(d_u),\ t\ :\ t+c_{i,d_u,u} \leq T,a} X_{i,d_u,t,u,a} = 1 \quad \forall u \qquad \text{(Return to depot)} \quad (5)$$

Equation (1) defines our objective function which represents the total reward collected during a given time horizon *T*. Constraint (2) ensures that each UAS begins its route from its depot. The strict equality of this constraint forces each UAS to begin a route at time-step 1. Constraint (3) is a balance of flow constraint, which ensures that if a UAS arrives at a node during any given time-step, it must depart that node on the same time-step. In constraint (4), we limit the number of times a reward is collected for each node to one; however, we permit UAS to transit previously visited nodes with no reward collection. Finally, constraint (5) ensures that each UAS returns to its depot no later than the last time-step. Note that we permit UAS to return home early with no penalty. This is an important aspect of the model because an optimal solution may result in the UAS arriving at the depot prior to the end of the search period with the next improved solution requiring more time than the search period permits.

B. RESTRICTIONS TO THE PROBLEM

In order to frame the size of our problem, we consider the number of arcs involved. We examine the limited case of a sector, which contains 100 nodes, searched by two UAS, which are capable of performing two actions at each node. For this network, we define an arc to be the straight line flight segment between two nodes. If an arc exists from each node to all other nodes – for each UAS and each action – the network then contains $I^2U \cdot 2 = 100^2 \cdot 2 \cdot 2 = 40,000$ potential arcs. This set of arcs is then duplicated for each time-step in the time horizon. For example, if T = 50, then this network would contain $I^2U \cdot 2 \cdot 50 = 2,000,000$ arcs. For our problem, this results in a significant amount of computing time for even the most efficient integer programming solvers available. With this in mind, we consider several restrictions to the problem in an attempt to produce solutions within a reasonable margin of the optimal solution in an acceptable period of computing time. Our analysis examines the use of the following restrictions and their impact on the optimal solution.

1. Node Clustering

We reason that a logical starting point with respect to restricting the problem is in dealing with nodes, which are located in close proximity to each other. Therefore, we refer to the first set of restrictions that we consider as *node clustering*. These cases present situations where it is sensible to trim certain arcs from the network and thereby reduce the number of possible solutions. We examine two cases and submit that, depending on the sweep-width of the UAS being modeled, we can apply rules with little or no impact on the optimal solution. The first case involves a sweep-width wide enough to cover nodes adjacent to one another, simply referred to as the large sweep-width case. The second addresses sweep-widths, which do not meet this criterion, and is consequently referred to as the *small* sweep-width case.

a. Large Sweep-width Case

For this case we apply the CETSP similar to the example given by Golden and Wasil (2006) and reassign reward values of nodes, which are in close proximity according to two sets of circumstances. For the situation where two nodes are uniquely adjacent to each other and set somewhat apart from other nodes in the network, we submit the rule depicted in Figure 8 which simply eliminates the node of lesser reward and adds its reward to the adjacent node. This clustering rule has the effect of creating one node of relatively high value and fewer arcs necessary to achieve this reward, which simplifies the decision of whether or not to include this node in an optimal solution. For a network of 100 nodes and 100^2 or 10,000 arcs per UAS, per time-step, we potentially eliminate up to 99 arcs going into the eliminated node as well as 99 arcs going out of it for a potential total reduction of 198 arcs (per UAS, per time-step) for each occurrence of this situation.

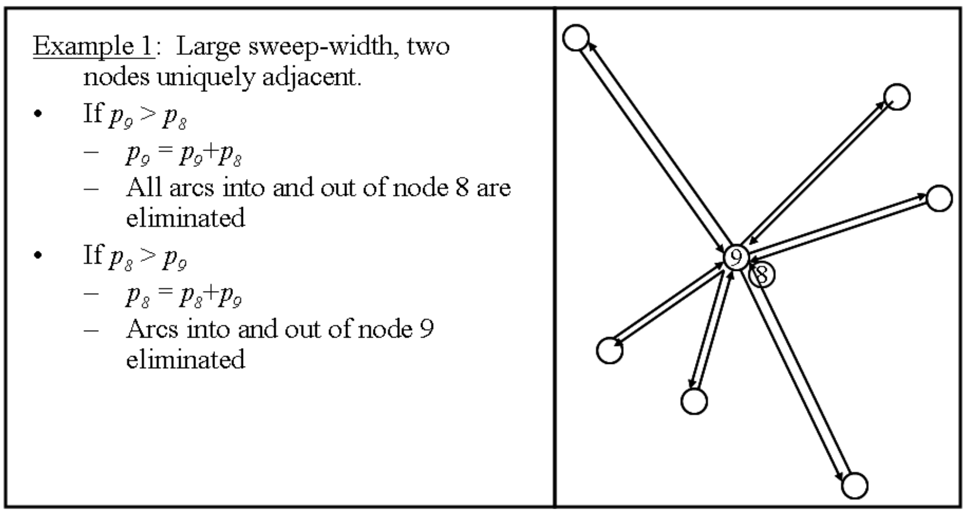


Figure 8. Node clustering for UAS with *large* sweep-width (Example 1).

A more likely occurrence, given that the nodes in our sector correspond to a road network, is the situation where several nodes are closely aligned in a row. For this circumstance, we submit the rule depicted in Figure 9. In this case we eliminate nodes and reassign rewards similar to the previous case; however we do this on the basis of position rather than reward value. We eliminate the node on each end and reassign the reward to the adjacent node. Any node which falls between these new end nodes is also eliminated with its reward divided between the new end nodes. For a 100 arc network similar to the one previously discussed, the situation in Figure 9 presents an opportunity to eliminate three nodes for a total of up to $3 \cdot 2 \cdot 99$ or 594 potential arcs (per UAS, per time-step) for the particular situation depicted.

Example 2: Large sweep-width, nodes closely aligned in a row.

- Arcs into and out of nodes 4, 8 and 10 are eliminated.
- Reward of node 4 added to node 7.
- Reward of node 10 added to node 9.
- Reward of node 8 divided between nodes 7 and 9.

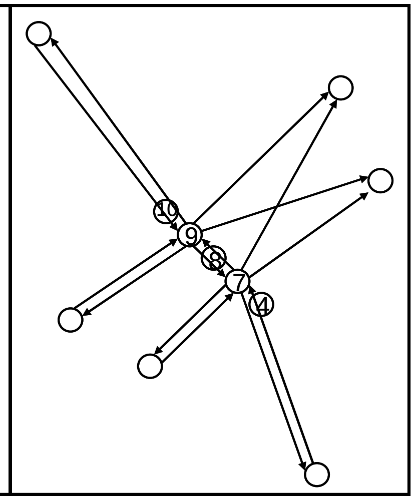


Figure 9. Node clustering for UAS with *large* sweep-width (Example 2).

b. Small Sweep-width Case

For the case where sweep-width is not large enough to sufficiently cover adjacent nodes, we consider only the situation of several nodes in a row and submit the rule depicted in Figure 10. We reason that in most cases where a row of nodes exist, it is beneficial to enter at one end and fly along the row picking up all the rewards before altering course. This clustering rule has the effect of creating one high value node at the center of the row which simplifies the decision whether of not to enter the row. By applying this rule, we would eliminate $2 \cdot 2 \cdot 99$ or 396 arcs (per UAS, per time-step) for the particular situation depicted.

Example 3: Small sweep-width, nodes closely aligned in a row.

- Arcs into and out of nodes 7 and 9 are eliminated.
- Rewards from 7 and 9 added to node 8.
- Node 8 is accessible only through nodes 4 and 10.

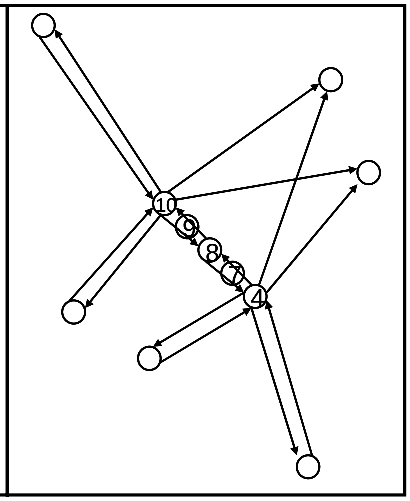


Figure 10. Node clustering for UAS with *small* sweep-width (Example 3).

We present a brief description of a spreadsheet model which we use for preprocessing in Section D. In order to implement the node clustering rules presented here, we create a macro in this preprocessing model as follows. We read in the given value for sweep-width as well as waypoint number, row indices, column indices and rewards. We then eliminate nodes and reassign rewards depending on the sweep-width of the UAS modeled according to these rules.

2. Arc Filtering

In addition to node clustering, we consider a second set of restrictions which we refer to as *arc filtering*. An important aspect of our model formulation is that UAS achieve rewards as they depart searched nodes. After clustering the nodes as discussed, we find that there are still many cases in our network where longer arcs strictly over-fly shorter arcs, yet do not account for their reward. These are circumstances which allow us to significantly reduce the network with no effect on the optimal solution. An example of this situation is depicted in Figure 11.

Example 1:

- $c_{3,7,u} + c_{7,8,u} = c_{3,8,u}$
- Arc (3,8) over-flies node 7 but does not account for its reward.
- Elimination of Arc (3,8) does not effect the optimal solution.

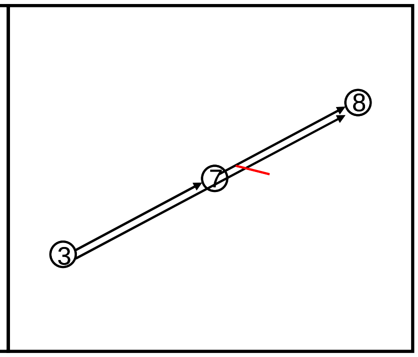


Figure 11. Arc filtering (Example 1).

Less obvious, yet still advantageous situations are those where the center node is only slightly offset from the arc connecting the other two. Depending on the geometry of the situation, the center node may lie within the sweep-width of a UAS traveling between the two outside nodes. Superficially, it may seem advantageous to eliminate the shorter arcs in favor of the long arc and somehow account for the reward of the center node. However, further contemplation reveals that elimination of the short arcs will restrict a significant number of possible paths involving nodes outside the three nodes considered. Furthermore, we reason that there are additional situations where the center node lays outside the sweep-width yet the difference between the combined length of the two short arcs and the long arc is insignificant. Consequently, we establish an additional *arc filtering* restriction by eliminating arcs based on the change in distance, δ, between the two possible paths in these situations, which we present in Figure 12.

Example 2:

- If $c_{3,5,u}+c_{5,9,u} < (1+\delta)\, c_{3,9,u}$, where δ represents an arc filtering criteria.
  - Consider elimination of arc (*3,9*) as a reasonable restriction of the problem.

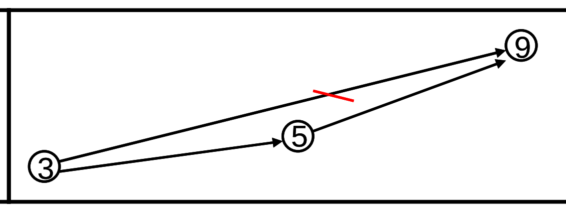


Figure 12. Arc filtering (Example 2).

3. Arc Length Limit

Lastly, we rationalize that it is unlikely an optimal route will contain excessively long arcs. For this reason we consider a third type of restriction which simply limits the length of arcs in our network. Our analysis in Chapter IV examines various potential arc length limits, their resulting computing times and effect on the solutions obtained.

C. WEATHER EFFECTS

One factor that comes into play in flight planning which can have a profound impact on UAS is the effect of winds. With many UAS operating at speeds less than 30 mph, it is not unlikely that they are met with winds in excess of their capabilities. For this reason, it is essential that we factor the effects of winds into the travel cost data for our model to ensure that it produces routes which are viable, let alone optimal. In order to do this, we turn to a set of equations that is used in flight planning and navigation system software, which define a concept known as the "Wind Triangle." We use these equations; depicted in Figure 13, to calculate UAS groundspeed for each arc in our network based on either forecast or observed winds.

- Wind Dir = Input wind + 180
  (Winds are given as "from")
- WTA = Desired Course – Wind Dir
- sin(WCA) = windspeed * sin(WTA)/Airspeed
- WCA = arcsin(sin(WCA))
- Groundspeed = airspeed * cos(WCA) + windspeed * cos(WTA)

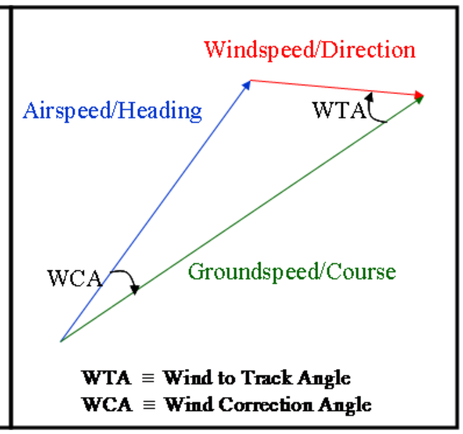


Figure 13. The Wind Triangle.

We then define the travel cost c_{iju} in our model as the distance between nodes i and j, divided by the groundspeed achieved between nodes i and j, multiplied by 60 seconds and again by the number of time-steps desired per minute. Because our model is an IP, we round these values up to the next integer for a close approximation of each arc in our network. Acknowledging that the cumulative effect of these approximations has an appreciable impact on total route time, we consider yet one more factor which has not been accounted for in our model – extra time incurred with turns. A simple flight planning technique is to connect waypoints by straight lines and determine travel times based on groundspeed and wind as we have previously discussed. Although some aircraft perform turns better than others, none are able to turn instantaneously and therefore add some varying distance to their route during each turn as depicted in Figure 14. We reason that unused travel time which results from the rounding error balances the extra time required for turns and ultimately produce routes which are close to the desired time horizon with c_{iju} values that are a reasonable approximation of reality. We present verification of this reasoning in Chapter IV.

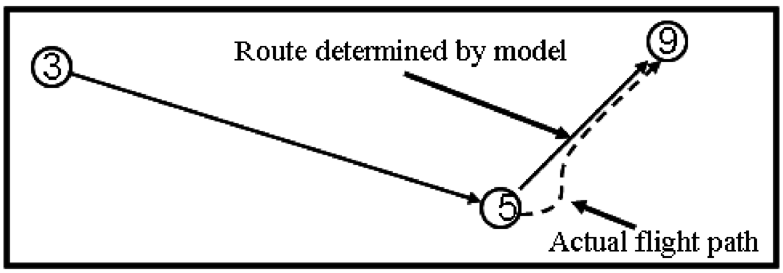
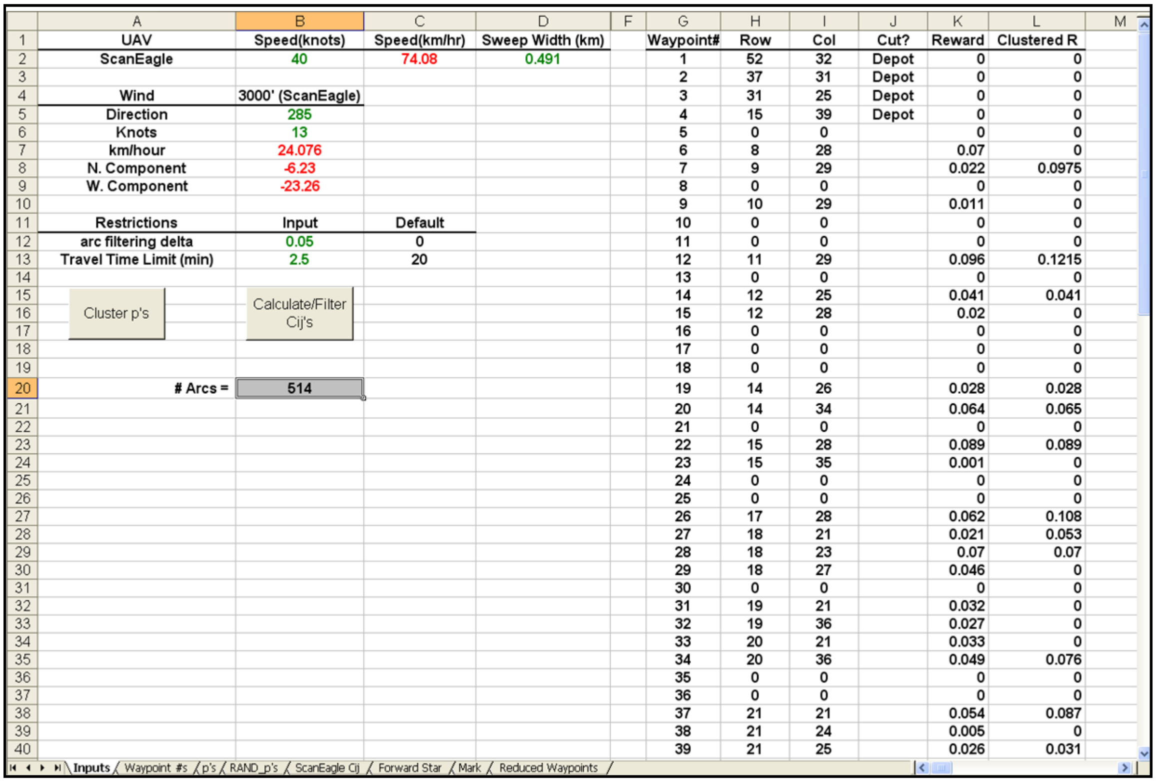


Figure 14. Additional time incurred by turns.

D. PREPROCESSING WITH EXCEL AND VBA

In order to provide a user interface with a means for changing input parameters to create data sets for various situations, we set forth to construct a spreadsheet model which carries out node clustering and arc filtering as depicted in Figure 15. We begin by establishing input blocks for the following adjustable parameters and arc filtering criteria: UAS airspeed, UAS sweep-width, wind speed, wind direction, arc travel time limit, and arc filtering δ. Additionally, we assign columns of the input sheet to represent various waypoint information including number, row index, column index and reward. We then write macros to perform node clustering, calculation of travel times, and arc filtering. While the macro for node clustering is straightforward as discussed in Section B, we describe the macros for calculation of travel times and arc filtering as follows.



	A	B	C	D	G	H	I	J	K	L
1	UAV	Speed(knots)	Speed(km/hr)	Sweep Width (km)	Waypoint#	Row	Col	Cut?	Reward	Clustered R
2	ScanEagle	40	74.08	0.491	1	52	32	Depot	0	0
3					2	37	31	Depot	0	0
4	Wind	3000' (ScanEagle)			3	31	25	Depot	0	0
5	Direction	285			4	15	39	Depot	0	0
6	Knots	13			5	0	0		0	0
7	km/hour	24.076			6	8	28		0.07	0
8	N. Component	-6.23			7	9	29		0.022	0.0975
9	W. Component	-23.26			8	0	0		0	0
10					9	10	29		0.011	0
11	Restrictions	Input	Default		10	0	0		0	0
12	arc filtering delta	0.05	0		11	0	0		0	0
13	Travel Time Limit (min)	2.5	20		12	11	29		0.096	0.1215
14					13	0	0		0	0
15					14	12	25		0.041	0.041
16	Cluster p's	Calculate/Filter Cij's			15	12	28		0.02	0
17					16	0	0		0	0
18					17	0	0		0	0
19					18	0	0		0	0
20	# Arcs =	514			19	14	26		0.028	0.028
21					20	14	34		0.064	0.065
22					21	0	0		0	0
23					22	15	28		0.089	0.089
24					23	15	35		0.001	0
25					24	0	0		0	0
26					25	0	0		0	0
27					26	17	28		0.062	0.108
28					27	18	21		0.021	0.053
29					28	18	23		0.07	0.07
30					29	18	27		0.046	0
31					30	0	0		0	0
32					31	19	21		0.032	0
33					32	19	36		0.027	0
34					33	20	21		0.033	0
35					34	20	36		0.049	0.076
36					35	0	0		0	0
37					36	0	0		0	0
38					37	21	21		0.054	0.087
39					38	21	24		0.005	0
40					39	21	25		0.026	0.031

Figure 15. Excel/VBA Preprocessing Model.

1. Travel Times

In order to create an array of travel times between each pair of nodes we create a macro in the following manner. We begin by reading in given values for UAV speed, wind speed, wind direction and travel time limit as well as the results from our node clustering macro. Using the row and column indices, which represent 200 meter increments of distance, we compute latitudinal and longitudinal distances between nodes *i* and *j*. Of course the values of these component vectors take on positive or negative orientations which represent directions north, south, east and west. By combining these component vectors together we find the straight line direction from node *i* to node *j* in degrees; this resultant vector is known in aviation terms as the "desired course" or "track." Within the same loop of code we compute the straight line distance between these nodes according to the Pythagorean Theorem – a close approximation to the actual great circle distances between these points of relatively close proximity. Next, using the *wind triangle* equations presented in Figure 13, we compute the wind to track angle

(WTA), the *sin* of the wind correction angle or *sin*(*WCA*), the wind correction angle (WCA), and finally the groundspeed that will be achieved when UAS u travels from node i to j. We finish by computing the travel time c_{iju} by dividing the distance by the groundspeed and then multiplying by 60 seconds and again by the number of time-steps desired per minute. An important distinction to be made is that with the wind effect factored in, the travel time from node i to node j will likely be different than from j to i.

Aside from using this macro to calculate travel times, we also use it to eliminate a number of arcs according to the following rules. First, by reading in the results of our node clustering macro, we can eliminate all arcs to and from nodes with zero reward. By performing this step at the beginning of the travel time macro, we eliminate numerous unnecessary calculations and achieve a reduction of computing time. Second, we reason that there is no use for arcs which go from a node to itself, and therefore eliminate all arcs having distances of zero. The exceptions to this rule are depot nodes which are assigned arcs returning to them with a travel time of one time-step. This construct permits a UAS to return to its respective depot prior to the last time-step and loiter there until the end of the search period with no penalty. Finally, we impose the arc length limit restriction proposed in Section B by eliminating all arcs with travel times in excess of the travel time limit provided as input. Note that this restriction is not applied to the arc lengths as measured in distances but to the actual travel times which are the true cost that they demand as a result of headwinds or tailwinds.

The product of this macro is an Excel sheet that contains an array of travel times for all possible combinations of nodes. Arcs which have been selected for elimination are assigned a travel time in excess of the cardinality of time-steps, say 99, which allows them to be identified and ignored later by an optimization solver.

2. Arc Filtering

After the node clustering and travel time macros have been completed, a third macro is used to apply the arc filtering rules previously discussed. We begin by establishing an additional sheet in the model which will receive reduced network in forward star format. Next, we read in an input value which we call δ that represents the user's criteria for filtering arcs according to the rule presented in Figure 12. We then iterate through the array created by the travel time macro, creating the forward star of all nodes which have not been eliminated. Subsequently, we sort the forward star according to travel time in descending order. We then iterate through the list of arcs, beginning with the longest, and perform a comparison between each arc and other possible paths. With each arc, we establish nodes *i* and *j*, then iterate through the list of nodes *k* in the network for a comparison of the travel time c_{iju} with the summed travel time $c_{iku} + c_{kju}$.

We perform this loop of comparisons in two separate routines and eliminate arcs according to the following criteria. For the first routine, if $c_{iku} + c_{kju} < 1.005 \bullet c_{iju}$, then node *k* is determined to be within a negligible distance of the path from *i* to *j* and thus the arc from *i* to *j* is eliminated in accordance with the example in Figure 12. In the second routine, if $c_{iku} + c_{kju} < (1+\delta) \bullet c_{iju}$, then the arc from *i* to *j* is removed imposing a modest restriction to the problem. One important implication of these routines is that they result in a compounding effect that we present in Figure 16. While the elimination of each arc results in a negligible change, subsequent arc eliminations can make the new shortest path between *i* and *j* significantly longer than the original shortest path.

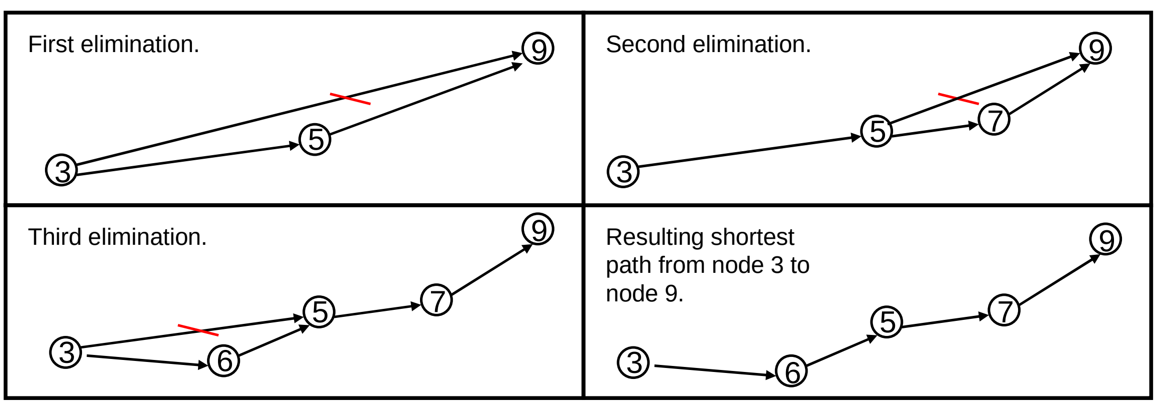


Figure 16. Compounding affect of arc filtering.

However, this effect is kept in check by the arc length limit restriction which has been previously imposed by the travel time macro. In order to quantify the limit of this effect, we present the following example. Assume we are operating a UAS with a cruise speed of 40 knots (or 78.1 km/hr) in a prevailing wind of 15 knots (or 27.8 km/hr), and let α represent the longest possible arc in the network. Then for a travel time limit of say 3 minutes, α would represent the longest possible distance covered with a direct tail wind in that amount of time. Therefore, in this case $\alpha = \left(\frac{74.1+27.8}{60}\right) \bullet 3 \approx 5.1 \text{ km}$. Furthermore, let β represent the shortest arc possible which is determined by the size of our 200 meter nodes equating to 0.2 km. The ratio of α and β then represent the extreme case or the maximum number of times that δ can be compounded and the maximum total additional distance that can be induced is:

$$\left(\frac{\alpha}{\beta}\right) \bullet \delta = \left(\frac{5.1}{0.2}\right) \bullet 0.1 = 2.55 \text{ km}$$

The maximum percent change in this case would be 2.55/5.1 = 50% increase in travel time required for the shortest path from node i to node j. Considering that the resulting path due to arc eliminations would have the benefit of accumulating rewards associated with α/β = 5.1/0.2 ≈ 25 additional nodes, it is reasonable to believe that this is a modest restriction of the problem.

E. AIRCRAFT DE-CONFLICTION

In some cases, particularly with small UAS, multiple aircraft are assigned to work in a given sector within a limited altitude block. For these situations it is indispensable that we ensure positive separation of aircraft throughout the search period. In order to address this challenge we present the following formulation to be added to our model.

Additional Index

g	Group of UAS to be de-conflicted.

Additional Sets

U_g	Vehicles $u \in U$ in group g. *All vehicles in a group are de-conflicted.*
$A(i,j)$	Set of arcs which cross arc (i,j).
$D(t)$	Set of time-steps to close to t for adequate separation of UAS.

Additional Constraints

$$\sum_{(i',j')\in A(i,j),t'\in D(t),u\in U_g,a} X_{i',j',t',u,a} \leq 1 \; \forall i,j,t,g \qquad \text{(Deconfliction)} \quad (6)$$

These additional constraints permit routes to cross one another, yet ensure that UAS maintain a minimum lateral distance between one another.

F. TIME WINDOWS

Up to this point we have limited our model to a case where IED events have an unlimited window of opportunity for detection. Reality is that the opportunity for identifying an impending IED attack begins when the emplacers arrive at the scene and ends when the IED is detonated. Ostensibly, these incidents occur at random times and last for random lengths of time providing few clues as to when and where they might take place. The truth is that they are specifically triggered by blue force activity which provides an insight into the times and locations that they might occur. In fact, when we consider that the presence of blue forces is a prerequisite for a casualty inflicting IED incident, we reason that locations of blue forces over the course of time limits the chances that they can occur. With this in mind, if we are able to merge information about planned blue force activity with information gained from IED prediction models, we reason that time windows for visits to certain nodes can be established for the model.

Consider the case where it is known in advance that a route clearance team will sweep a particular route some time before we expect a convoy to travel the route. If these operations take place at their planned times, then for each particular cell that contains a portion of the route in question we have a window of opportunity for insurgents to emplace an IED and target the coming convoy. This window begins at the time the route clearance team is outside of visual range of the cell and lasts until the convoy arrives. This window of opportunity for the insurgents translates to a time window that the cell should be a candidate for search. Therefore, assuming that accurate information of blue force activity is available; we propose the following additions to our model.

Additional Set

$W(i)$ Time-steps t that are within time window for node i

Additional Constraints

$$\sum_{j \in F(i), t \notin W(i), u} X_{i,j,t,u,'search'} = 0 \quad \forall i \qquad \text{(Prevent searches outside time window)} \quad (7)$$

While we acknowledge that blue force operations do not always go as planned, we believe this type of coordination between UAS efforts and blue force activity does stand to bear fruit not only in preventing IED casualties but also by catching insurgents in the act, revealing critical intelligence for disrupting the network behind the device. Furthermore, the addition of the constraints proposed is a significant restriction of our original problem which would have an expected benefit of reducing computing time and/or permitting the relaxation of restrictions previously implemented.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. COMPUTATIONAL STUDY

This chapter presents results obtained through testing ISOM in field experiments with the use of actual UAS equipment and experienced personnel. We begin our discussion by developing a scenario which we use to study ISOM in two NPS-SOCCOM field exercises at Camp Roberts. Using this scenario as a basis, we then examine implementation of our model in the General Algebraic Modeling System (GAMS) (GAMS, 2007) and discuss considerations we took into account in order to find solutions reasonably close to optimal while reducing computing time to a practical level. Finally, we present results of the field exercises and discuss learning points that we garner from these events.

A. SCENARIO DEVELOPMENT

After establishing the Camp Roberts training area as the site for our experimentation, we discretize the space within the AO that contains all National Geospatial-Intelligence Agency (NGA) identified roads into a grid of 200 meter by 200 meter cells. We accomplish this by laying a spreadsheet of square cells over a map of the area generated with the use of FalconView flight planning software (FalconView, 2007). In the absence of real world data for our area of study, including a history of IED events, associated infrastructure and related geographic features, it was not feasible to implement an actual prediction model for this region. Therefore, data is randomly generated to mimic prediction model output over the Camp Roberts area for the purposes of these experiments. Next, the set of waypoints is reduced to include only the upper 30% of reward values in order to reduce the size of the problem for an initial test case. This resulted in a set of 80 waypoints including four established as UAS depots.

Three types of UAS are available for the experiments: Scan Eagle, Buster, and Raven which operate at 40, 35 and 25 knots respectively. In order to test various aspects of our model, it is decided that two types of experiments would be conducted concurrently. Scan Eagle, which is operated at a higher altitude and has a relatively large sweep-width would be used to test the large sweep-width case of the model, and would be free of the de-confliction constraint proposed in Chapter III. Buster and Raven, which have smaller sweep-widths and operate within a narrow block of altitude required de-confliction which presents a realistic set of circumstances for testing the multiple UAS case discussed in Chapter III. In order to conduct several experiments in the time allotted, the search periods are limited to 25 minutes in length and broken down into 50, 30-second time-steps which resulted in a problem size of $I^2U \cdot 2 \cdot 50 =$ $80^2 \cdot 2 \cdot 50$ = 1,280,000 arcs before restrictions are applied.

With the resulting problem requiring a significant amount of computing time, we consider means to reduce its size. In order to avoid a complete duplication of the arc set for transit purposes, we reason that a limited number of transit nodes achieve our objective of ensuring freedom of movement. For that reason, we select one node for each one kilometer square block within our sector to act as a transit node. Furthermore, we limit transit arcs departing each transit node to only those other transit nodes which are leading in the direction of the depots and within a limited distance of the originating node. The resulting network then contains 80^2 search arcs, plus approximately $17 \cdot 3$ transit arcs for a total of 6451 arcs for each UAS and each time-step and hence for the two UAS, 50 time-step case previously discussed, we now have $6451 \cdot 2 \cdot 50$ = 645,100 arcs.

While the most important aspect of the experiments is to study the mechanics of modeling UAS flight, in order to add an element of appeal for our operators, a group of participants is formed in the role of a red team which would act as IED emplacers. Keeping in mind that our study does not concern development or testing of sensor technology, the red team is equipped with a number of large, conspicuously colored tarps which would serve as notional IED detection opportunities. In order to place these detection opportunities in a random fashion – according to the likelihood values of our cells, we compared the column of rewards in our Excel preprocessing model with a column of random numbers. We then scaled the column of rewards to a level that produced the desired number of instances where the random number was larger than the reward value. Next, we actuated the random number generator to create eight scenarios, each having four or five IED emplacement sites. The red team leader then drew one of these scenarios at random for each experiment and placed the IED accordingly.

B. IMPLEMENTATION IN GAMS

With the experiment scenario established, we implement our Integer Program model in GAMS and found optimal or near-optimal solutions using the CPLEX solver (ILOG, 2007). The following results were obtained by running the model on a Dell Precision PWS690 Intel® Xeon™ CPU 3.37GHz processor, with 3.00 GB of RAM.

1. Use of CPLEX Options

In order to reduce computing time we consider the use of CPLEX options which can be used to enhance the solver by tailoring it for a particular problem. To examine the effect of CPLEX options, we consider the case where a sector to be searched by a single UAS operating from depot #1 has been reduced to contain 77 nodes – including the depot. For this case, we used the following input parameters. UAS airspeed = 40 knots, sweep-width = 0.5, wind from 285° at 13 knots, $\delta = 0.1$, travel time limit = 1.75 minutes and an optimality tolerance of 0.05. The resulting network contains 514 search arcs plus $17 \cdot 3 = 51$ transit arcs for each time-step resulting in a total of 28,250.

The first option we consider is 'branch direction' (*brdir*), an option which is used to set the priority order for branching at each node in the branch and bound tree. For *brdir*, the default setting of '0' lets the algorithm decide the branch direction, while '-1' sets the down-branch as priority and '1' sets the up-branch as priority. A second option considered is the 'pricing strategy for dual simplex method' (*dpriind*). This option permits the selection between six different dual, steepest-edge pricing selections which can be particularly efficient compared to primal steepest-edge pricing. Finally, a third option we investigate is the 'cuts' option which permits selection between five different aggressiveness levels of CPLEX cut generation. We run the model with each of the options selected exclusively, and for each setting available for each option. We then select a limited number of combinations of options and settings and present the results in Table 1.

Table 1. Results from various CPLEX option selections.

CPLEX Option Selections/Settings			Computing Time
brdir	***dpriind***	***cuts***	**[min:sec]**
0	--	--	8:12
1	--	--	**5:37**
-1	--	--	9:23
--	0	--	8:07
--	1	--	28:16
--	**2**	**--**	**4:54**
--	3	--	14:30
--	4	--	**6:04**
--	5	--	**6:07**
--	--	-1	7:35
--	--	0	8:06
--	--	1	**5:10**
--	--	2	9:30
--	--	3	26:38
1	2	--	7:17
-1	2	--	**6:40**
--	2	1	9:11
1	--	1	10:59

Model Inputs: UAS Speed – 40 knots, Wind – 285 / 13, Arc Filtering δ – 0.1, Travel Time Limit – 1.75 min, Sensor Sweep-width – 0.5 (Resulting in 514 search arcs per time-step), Optimality tolerance – 0.05. The number of transit arcs (per time-step) was 51.

Noting that the run depicted in the first row of Table 1 represents the equivalent of not using any options, this run establishes our baseline computing time of 8:12. Comparing the results in the table, we see a wide range of computing times and identify six configurations that appear to be competitive which are depicted in bold print. We make a closer examination of these six options as well as the default case by creating four different problems from our basic scenario with varying wind inputs to the preprocessing model. We then ran the GAMS model with each of the six CPLEX option configurations and for each of the four problems created, and present the results of these runs in Table 2. From these results we see that while the solution obtained for each problem is very close, regardless of the configuration, the required computing times vary considerably. We also see that the configuration of having the exclusive option *brdir* = 1 resulted in the best performance for two out of the four problems presented and displayed consistent performance throughout this experiment. For this reason we choose this configuration for the remaining of our analysis.

Table 2. A closer examination of six CPLEX option choices.

CPLEX Option Selections/Settings					Computing Time	
Winds	# Search Arcs	*brdir*	*dpriind*	*cuts*	[min:sec]	IP Solution
037 / 15	419	--	--	--	4:46	2.227
		1	--	--	4:27	2.227
		--	2	--	3:57	2.227
		--	4	--	**3:44**	**2.227**
		--	5	--	4:51	2.227
		--	--	1	4:24	2.227
		-1	2	--	3:58	2.227
135 / 15	437	--	--	--	2:40	1.939
		1	--	--	**1:58**	**1.939**
		--	2	--	2:18	1.939
		--	4	--	2:52	1.939
		--	5	--	2:03	1.939
		--	--	1	2:06	1.939
		-1	2	--	2:16	1.908
215 / 15	497	--	--	--	3:24	2.337
		1	--	--	3:02	2.339
		--	2	--	**2:56**	**2.339**
		--	4	--	3:30	2.339
		--	5	--	5:03	2.315
		--	--	1	3:52	2.339
		-1	2	--	3:50	2.339
360 / 0	575	--	--	--	12:38	2.729
		1	--	--	**11:34**	**2.729**
		--	2	--	11:40	2.729
		--	4	--	16:02	2.712
		--	5	--	16:20	2.716
		--	--	1	19:30	2.694
		-1	2	--	16:38	2.705

Model Inputs: Same as for Table 1 with the exception of wind. The number of transit arcs for this network was 51 (per time-step).

2. Effect of Restrictions on Optimal Value

With CPLEX tuned for our problem, we now proceed by examining the effects of restrictions presented in Chapter III. We begin by establishing a base case from which to judge the quality of solutions obtained with restrictions imposed based on the following considerations. First, we reason that while node clustering will likely have some measurable affect on the optimal value, it is a necessary step in order to produce routes that are mechanically feasible with a reasonable number of turns. Furthermore, we reason that the elimination of arcs which strictly coincide with shorter arcs does not apply a restriction to the problem but simplifies the solution by removing arcs which would not be part of an optimal solution. After applying these steps, our base case for the single UAS operating in a 77 node network, with a 50 time-step search period results in 1997 search arcs plus 51 transit arcs for a total of 2048·50 = 102,400 arcs.

With a base case established, we create three different problems by applying wind inputs of 360° at 0 knots, 275° at 15 knots and 315° at 12 knots to our preprocessing model. For each wind input, we establish a solution with the base case inputs and with optimality criteria (OPTCR) of 1%. We then run the model with various increasing levels of filtering affects, and for OPTCR of 1% and 5% and present results in Table 3.

Table 3. Affect of restrictions on optimality.

Winds	Travel Time Limit	δ	# Search Arcs	OPTCR	Computing Time [hr:min:sec]	IP Solution
360 / 0	--	--	1155	0.01	5:09:49	2.763
	--	--	1155	0.05	3:48:37	2.763
	4	--	1117	0.01	4:34:32	2.763
	4	--	1117	0.05	4:00:16	2.742
	2	--	735	0.01	1:32:13	2.763
	2	--	735	0.05	1:18:49	2.754
	--	0.05	883	0.01	3:51:01	2.763
	--	0.05	883	0.05	2:46:07	2.763
	4	0.05	841	0.05	0:56:33	2.763
	2	0.05	555	0.05	0:37:05	2.726
275 / 15	--	--	1100	0.01	1:07:17	2.497
	--	--	1100	0.05	0:48:48	2.497
	4	--	1019	0.01	1:09:46	2.497
	4	--	1019	0.05	0:40:40	2.497
	2	--	677	0.01	0:50:37	2.497
	2	--	677	0.05	0:37:45	2.493
	--	0.05	868	0.01	0:39:04	2.497
	--	0.05	868	0.05	0:26:19	2.497
	4	0.05	791	0.05	0:29:56	2.493
	2	0.05	552	0.05	0:20:43	2.478
315 / 12	--	--	1129	0.01	20:57	2.430
	--	--	1129	0.05	18:23	2.430
	4	--	1073	0.01	13:48	2.430
	4	--	1073	0.05	11:41	2.430
	2	--	711	0.01	36:31	2.390
	2	--	711	0.05	21:43	2.390
	--	0.05	891	0.01	10:34	2.408
	--	0.05	891	0.05	09:06	2.408
	4	0.05	833	0.05	11:17	2.408
	2	0.05	565	0.05	05:10	2.390

Model inputs were adjusted to create four scenarios of varied complexity which is indicated by the resulting number of arcs. CPLEX option brdir = 1. The number of transit arcs per time-step was 51.

From these results we see significant reductions in computing times with each increase in restriction. In fact the reduction in computing time for each of the three problems with the most restrictive inputs (travel time limit of 2 minutes and δ = 0.05) coupled with the widest optimality tolerance (0.05) can be computed to be 88%, 69% and 75% respectively for an average reduction of 77%. In comparison with the base case, the first problem presented in Table 3 is formed by applying the restrictions in a notional situation of calm wind. The most restricted case for this problem resulted in 555 search arcs which implies a network size of (555+51)·50 = 30,300 arcs total and a reduction of 102,400-30,300 = 90,100 or 74.8%. Furthermore, for each of the three problems the solution obtained with the most restricted configuration and optimality tolerance of 0.05 was within 2% of the corresponding base case with optimality tolerance of 0.01. This implies that all of these solutions are within 3% of the optimal solution – a nominal reduction considering the resulting reduction in computing time.

C. FIELD EXPERIMENT RESULTS

Field experiments were conducted during exercises at Camp Roberts in February and again in May of 2007. The purpose of the experiments was to learn about the mechanics of modeling UAS flight and to make corresponding adjustments to the model. A screenshot taken from the tactical operations center which shows an example of three routes generated by the model for two concurrent experiments is depicted in Figure 17. Here, the routes for Scan Eagle, Buster, and Raven are shown in yellow, green, and blue respectively. Buster and Raven were operated in concert with one anther and were de-conflicted while Scan Eagle was treated as a separate experiment.

The most important insights that were gained from the exercises at Camp Roberts related to the mechanics of modeling UAS operations – specifically in dealing with winds and with the extra time required by turns. In addition to these factors, we recall the error incurred by rounding travel times to the next integer time step, and consider its impact on the resulting route completion times. Our initial thoughts were that the rounding error and the turn factor would cancel each other to some extent but may require some additional adjustment. For this reason we tested various speed factor inputs ranging from

70% to 100% of actual UAS speed. Additionally, we ran experiments with and without wind correction inputs – based on wind data collected on site by a weather balloon – to see how effective the model was in producing routes that represent reality. That is, whether the routes produced by the model were achievable by UAS in prevailing wind conditions in the time allowed.

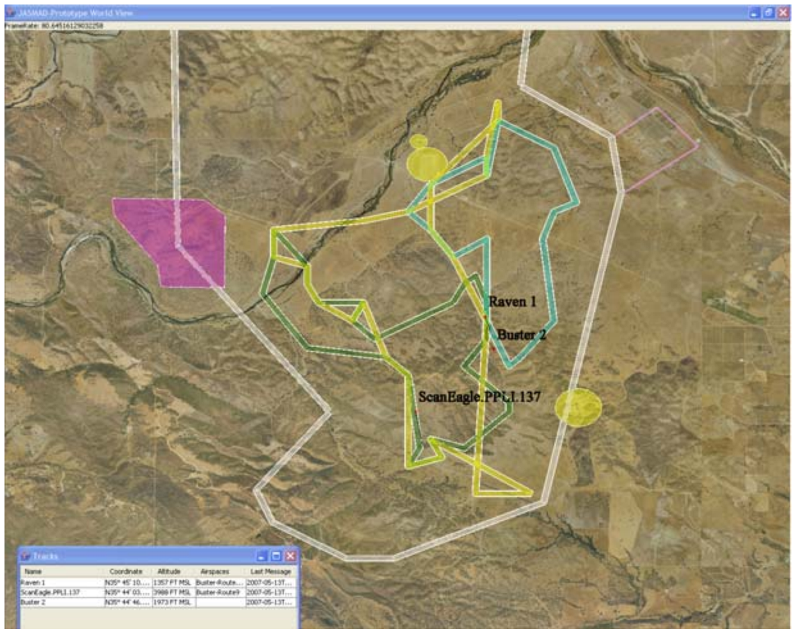


Figure 17. Routes flown during an experiment at Camp Roberts.

Results from nine, tentatively 25-minute experiments involving Buster and Raven aircraft we depict in Table 4. One observation of note is that runs we complete with a 100% speed factor result in completion times which are generally close to the targeted 25 minute time horizon while lower speed factors usually resulted in route completion times which were short of the time horizon. The only exception to this finding is experiment #3 where Raven completed its route at 28 minutes with 90% speed factor applied and no wind correction. Additionally, experiments conducted without wind correction seem to result in a wider variance of completion times for each of the speed factors applied while the wind corrected runs seem to have a steady progression in completion times corresponding with increasing speed input factors. While there is not enough data to make a conclusive argument, we believe the results do suggest that a speed input factor of

100% coupled with our wind correction method produce routes with associated total time required that is close to the targeted time horizon set by the experiment length and that the rounding error is negated by the extra time required by turns.

Table 4. Field experiment results: wind correction vs. various speed factors.

Experiment #	Speed Reduction Factor	Wind Corrected	UAS	Actual Time
1	70%	NO	Buster	13 min
			Raven	14 min
2	90%	NO	Buster	23 min
			Raven	24 min
3	90%	NO	Buster	24 min
			Raven	28 min
4	90%	NO	Buster	20 min
			Raven	*(no data)*
5	100%	NO	Buster	24 min
			Raven	*(no data)*
6	100%	NO	Buster	29 min
			Raven	*(no data)*
7	85%	YES	Buster	22 min
			Raven	19 min
8	90%	YES	Buster	20 min
			Raven	21 min
9	100%	YES	Buster	24 min
			Raven	23 min

D. MODEL SUCCESS IN DETECTION OPPORTUNITIES

In order to examine ISOM's ability to produce an increase in detection opportunities we generate four problems by arbitrarily selecting wind inputs from various directions and speeds. Notional routes are then computed for the Scan Eagle UAS using a travel time limit of 2 minutes, parameter $\delta = 0.05$ and an optimality tolerance of 0.05. We compare the resulting routes of each of these wind inputs with the eight IED scenarios used in the exercise at Camp Roberts. A compilation of the detection opportunities which would occur given these circumstances we present in Table 5. For each route we list the wind input, resulting number of arcs (per time-step), resulting number of nodes searched (out of 79 possible), the number of IED detection opportunities achieved (out of the number possible), and a total number of detection opportunities over all eight scenarios. The result of this example is that we have an expected 78% of detection opportunities achieved while searching an expected 55% of the nodes in the reduced network.

Table 5. Model success rate as measured by detection opportunities.

Route	Wind	# Search Arcs	#Nodes Searched	Detection Opportunities/Scenario: 1	2	3	4	5	6	7	8	Total
1	035 / 14	478	42 / 79	4 / 5	3 / 4	2 / 4	2 / 4	4 / 4	3 / 4	4 / 4	2 / 4	24 / 33
2	135 / 12	516	41 / 79	4 / 5	3 / 4	3 / 4	2 / 4	4 / 4	3 / 4	3 / 4	2 / 4	24 / 33
3	235 / 10	580	45 / 79	4 / 5	3 / 4	3 / 4	4 / 4	4 / 4	4 / 4	4 / 4	2 / 4	28 / 33
4	310 / 12	568	45 / 79	5 / 5	4 / 4	3 / 4	2 / 4	4 / 4	4 / 4	4 / 4	1 / 4	27 / 33
		Avg #Cells Searched:	**43.25 / 79**								**Grand Total:**	**103 / 132**
		% Searched:	**55%**								**Success Rate:**	**78%**

V. CONCLUSIONS AND FUTURE WORK

A. CONCLUSIONS

This thesis develops a vehicle routing tool for Unmanned Aerial Systems (UAS) tasked with interdiction of Improvised Explosive Devices (IED) which we refer to as ISOM. ISOM uses optimization models and algorithms to leverage recent developments in IED prediction as well as emerging UAS and sensor technology. Tactical level operators can use ISOM to determine routes that best employ their UAS for the purpose of detecting IED or IED related activity.

ISOM receives output from an existing IED prediction model and uses it as a means to establish relative values for searching various portions within a sector of operation. To model interdiction of IED with the aid of a prediction model, we discretize the space around roadways within the sector of operation into portions and assign values to represent the likelihood of an IED event occurring in that cell the following day. These values are then treated as rewards which can be achieved by a UAS after it has searched the respective cells. The output of the tool is a set of routes which focus UAS search efforts in the most likely areas of IED occurrence and result in the maximum number of detection opportunities within the search time allowed by the UAS endurance.

1. Measures of Success

In order to measure the effectiveness of ISOM compared to manual selection of routes (after the node clustering step has been accomplished), the scenario used at Camp Roberts was presented to a group of experienced UAS operators deployed to Iraq who were asked to provide their solutions to this problem had they been given the same tasking. Due to technical challenges that have arisen with the operators deployed to Iraq, results from this group are still awaited at the time of this writing. However, preliminary examination reveals that ISOM produces viable routes which result in efficient use of given assets. Field testing of ISOM suggests that it models the mechanics of UAS flight quite well. Routes selected by ISOM display a reasonable number of turns, and result in mission times which are close to the intended search period. This means they will make effective use of sortie time while not extending a UAS beyond its fuel capacity. Furthermore, routes do not cross back over themselves to the extent that they seem unrealistic or inefficient. Additionally, reasoning that a random search of 55% of the nodes in a network would be expected to achieve 55% of the detection opportunities possible, the results depicted in Table 5 suggest that ISOM achieves a (78-55)/55 = 42% increase in the likelihood of achieving a detection opportunity over searching the nodes in a random manner.

2. Operational Considerations

A critical aspect for employment of ISOM is the necessity of preplanned coordination. In order to capitalize on the successes achieved by ISOM there must be a means to close the loop when IED or IED emplacing activity is detected. The establishment of an IED radio net is the first necessary step that should be taken. When detection is made, the location should be broadcast over a secure net that all blue force units monitor while they operate in areas where IED are likely. Radio relay would be accomplished by airborne platforms such as EA-6Bs, EC-130s and other aircraft that routinely fly missions over the AO. Secondly, a quick reaction force should be assigned within each sector of the AO in order to respond to IED emplacement sightings. When such sightings occur, the UAS should maintain sight of insurgents and possibly follow them to safe houses or weapons caches. The respective quick reaction force should be alerted in a timely manner so that it can respond appropriately in order to arrest insurgents, disarm the device and acquire important intelligence regarding the source of the IED.

B. FUTURE WORK

1. Dynamic Updates

Placing a UAS at the right place at the right time requires current, accurate information on blue force locations. In order to achieve the most up to date information possible, ISOM should be capable of providing dynamic updates of UAS routes. Additionally, while we have presented formulation to account for time windows as a means to coordinate with blue force ground activity, we have not tested this aspect of

ISOM. Ideally, the final result of this effort will be as follows. Routes are pre-selected by ISOM based on forecast wind, planned blue force activity and IED prediction model results. As the UAS flies a pre-selected route, ISOM receives additional and updated inputs including a forecast UAS position and fuel state, current blue force locations and directions of travel, and prevailing winds. The UAS then receives a dynamic update as it reaches the forecast location and fuel state, and flies from that position direct to the first waypoint on a route with refined time windows.

2. Turn Angle Penalties

An additional aspect of ISOM that should be examined is the effect of turn radius on finding IED. Aside of the extra time that turns require, they also have the effect of placing the UAS slightly off its intended course as depicted in Figure 14. Since this effect increases with the angle of the turn, penalties should be considered for high angle turns. This could be achieved with the introduction of an additional index on the binary variable to represent the node visited prior to i and could possibly have some positive effects on ISOM's results including smoother routes and shorter computing times.

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF REFERENCES

Department of Defense (2005). Directive 2000.19 Joint IED Defeat Task Force (JIEDDTF).

Department of Defense (2006). Directive 2000.19E Joint IED Defeat Organization (JIEDDO).

Chisholm, P. (2005). “Clearing the Roads.” Special Operations Technology Online, Vol. 3, Retrieved 21 November, 2006, from http://www.special-operations-technology.com/article.cfm?DocID=1129.

Fabey, M. (2007). “UAVs, Other Aircraft Being Misused in IED Fight, ACC Chief Says.” Aerospace Daily and Defense Report, Vol. 222, No. 58, p. 1.

FalconView Homepage. Retrieved September 2, 2007, from http://www.falconview.org/.

Feillet, D., Dejax, P. and Gendreau, M. (2005). “Traveling Salesman Problems with Profits.” Transportation Science, Vol. 39, No. 2, pp. 188-205.

Funkhouser, D.A. (2006). Unmanned Aerial Systems Division Head, Marine Corps Warfighting Lab. Private communication Dec 11.

GAMS Homepage. Retrieved August 26, 2007, from GAMS Homepage Web site: http://www.gams.com/.

Golden, B.L. and Wasil, E. (2006). University of Maryland, Robert H. Smith School of Business. “A Guided Tour of Several New and Interesting Routing Problems.” Retrieved September 8, 2007, from http://www.rhsmith.umd.edu/faculty/bgolden/recent_presentations.htm.

iCasualties.org Retrieved September 8, 2007, from Iraq Coalition Casualty Count Web site: /oif/IED.aspx (5 January 2007).

ILOG Optimization. Retrieved September 2, 2007, from ILOG.com Web site: http://www.ilog.com/products/optimization/archive.cfm.

Jones, J.S. (2006). “Modeling Detection Strategies to Battle Improvised Explosive Devices.” Master’s thesis, Naval Postgraduate School, Monterey, CA.

Kelly (2006). Defense Technical Information Center Homepage. Retrieved September 9, 2007, from DTIC Search Web site: http://www.dtic.mil/ndia/2006expwarfare/kelly.pdf.

Kress, M. and Royset, J.O. (2007). “Decision Aid for Deployment and Employment of UAVs in Special Operations Missions.” Military Operations Research, to appear.

Lantz, R. W. (2006). “A Data Mining Approach to Forecasting IED Placement in Space and Time.” Master’s thesis, Naval Postgraduate School, Monterey, CA.

Levine, H. (2006). “Improvised Response: Combating IEDs Will Require an Agile Game of Action and Reaction.” Bulletin of the Atomic Scientists, Vol. 62, No. 4, pp. 22-23.

McFee, J.E. and Ripley, H.T. (1997). “Detection of Buried Landmines Using a Casi Hyperspectral Imager.” Retrieved September 8, 2007, from Smithsonian/NASA ADS Physics Abstract Service Web site: http://adsabs.harvard.edu/abs/1997SPIE.3079..738M.

Moser, H.D. (1990). “Scheduling and Routing Tactical Aerial Reconnaissance Vehicles.” Master’s thesis, Naval Postgraduate School, Monterey, CA.

Odierno, R.T. (2007). “Counterinsurgency Guidance.” Retrieved September 8, 2007, from Small Wars Journal Online Web site: http://www.smallwarsjournal.com/blog/2007/06/hq-mnci-counterinsurgency-guid/.

Owen, P., Martin, R. and Carriger, T. (2005). “Shadowing IEDs.” Unmanned Systems, Vol. 23, No. 5, pp. 13-16.

Pike, J. and Aftergood, S. (2006). “Hyperspectral Imaging.” Federation of American Scientists. Retrieved September 8, 2007, from FAS Web site: www.fas.org/irp/imint/hyper.htm.

Riese, S.R. (2006). “Templating an Adaptive Threat: Spatial Forecasting in Operations Enduring Freedom and Iraqi Freedom.” Engineer Magazine, January-March, pp. 42-43.

Rosenberg, B. (2007). “Honeywell Micro-UAV Soon on IED Patrol in Iraq.” Retrieved September 8, 2007, from DefenseNews.com Web site: http://defensenews.com/story.php?F=2842081&C.

Schachtman, N. (2007). “IED Defeat Group: How Effective?” Retrieved September 8, 2007, from Wired Blog Network Web site: http://blog.wired.com/defense/2007/06/ied_defeat_grou.html.

Suarez, R. (2007). “Military Grapples with Onslaught of Homemade Bombs in Iraq.” Retrieved September 8, 2007 from The Online News Hour Web site: www.pbs.org/newshour/bb/middle_east/jan-june07/ied_06-21.html.

Toth, P. and Vigo, D. (2002). “The Vehicle Routing Problem.” Philadelphia, PA: Society for Industrial and Applied Mathematics.

Trimble, S. (2006). “U.S. Eyes Hyperspectral Technology for UAVs.” Jane’s Defense Weekly, 6 September, p. 31.

Walkenbach, J. (2004). “Excel 2003 Power Programming with VBA.” Wiley Publishing Inc. Indianapolis, IN.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
   Ft. Belvoir, Virginia

2. Dudley Knox Library
   Naval Postgraduate School
   Monterey, California

3. Marine Corps Representative
   Naval Postgraduate School
   Monterey, California

4. Director, Training and Education
   MCCDC, Code C46
   Quantico, Virginia

5. Director, Marine Corps Research Center
   MCCDC, Code C40RC
   Quantico, Virginia

6. Marine Corps Tactical Systems Support Activity (Attn: Operations Officer)
   Camp Pendleton, California

7. Director, Operations Analysis Division
   MCCDC, Code C45
   Quantico, Virginia

8. Professor Johannes O. Royset
   Naval Postgraduate School
   Monterey, California

9. Professor Robert F. Dell
   Naval Postgraduate School
   Monterey, California

10. COL Alejandro Hernandez, Associate Dean GSOIS
    Naval Postgraduate School
    Monterey, California

11. Professor Gordon H. Bradley
    Naval Postgraduate School
    Monterey, California

12. Professor Moshe Kress
Naval Postgraduate School
Monterey, California

13. Mr. Jack Keane
Johns Hopkins University Applied Physics Lab
Global Engagement Department
Laurel, Maryland

14. Dr. David Netzer, Emeritus Professor
Naval Postgraduate School
Monterey, California

15. Dr. Nancy Spruill
Director, Acquisition, Resources and Analysis
Office of the Under Secretary of Defense (Acquisition, Technology and Logistics)
Pentagon, Washington, D.C.

16. Mr. Ken Amster
Naval Air Systems Command, Maritime Patrol and Reconnaissance Aircraft
Patuxent River, Maryland

17. Mr. Steve Kirin
Joint IED Defeat Organization
Arlington, Virginia

18. Mr. Robert Portigue
Joint IED Defeat Organization
Arlington, Virginia

19. Dr. Thomas Allen
Institute for Defense Analysis
Arlington, Virginia