Optimization-Based Routing and Scheduling of IED-Detection Assets in Contemporary Military Operations

by

Christopher E. Marks

B.S. Mechanical Engineering
United States Military Academy, 1998

SUBMITTED TO THE SLOAN SCHOOL OF MANAGEMENT IN PARTIAL FULFILLMENT OF THE REQUIREMENTS OF THE DEGREE OF

MASTER OF SCIENCE IN OPERATIONS RESEARCH

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 2009

Signature of Author: _____
Sloan School of Management
Interdepartmental Program in Operations Research
14 May 2009

Approved by: _____
LTC Darryl K. Ahner
Assistant Professor, United States Military Academy
Technical Advisor

Approved by: _____
Stephan E. Kolitz
The Charles Stark Draper Laboratory, Inc.
Technical Supervisor

Certified by: _____
Cynthia Barnhart
Professor, Civil and Environmental Engineering
Co-Director, Operations Research Center
Thesis Advisor

Certified by: _____
Dimitris Bertsimas
Boeing Professor of Operations Research
Co-Director, Operations Research Center

[This Page Intentionally Left Blank]

Optimization-Based Routing and Scheduling of IED-Detection Assets in Contemporary Military Operations

by

Christopher E. Marks

Submitted to the Sloan School of Management on 14 May 2009, in Partial Fulfillment of the Requirements for the Degree of Master of Science in Operations Research

ABSTRACT

Improvised Explosive Devices, or IEDs, have become a familiar and lethal part of contemporary military operations in Iraq and Afghanistan, producing more casualties than any other weapons system. One reason for their success is their practicality in an environment characterized by imbalances in the capabilities of opposing forces. The military forces conducting stability operations in Iraq and Afghanistan rely on the existing road networks to support logistical and operational movements. Insurgents with limited firepower and maneuver capabilities can place a bomb on the side of a road and detonate it anonymously to cause catastrophic effects on a passing convoy.

Route clearance teams were developed to combat the emerging threat of IEDs. Capable of detecting IEDs with minimal risk to troops, route clearance teams move along the road network in search of these destructive devices. This thesis explores a mathematical approach to planning and scheduling route clearance missions. To achieve this objective, we first develop a probability-based model of IED activities on a road network used by occupation forces. We then use approximate dynamic programming methods to generate potential route clearance missions that are effective at reducing the risk of IED attacks. Once the paths are generated, they are inputted into a mixed integer program that finds the most risk-reducing combination of missions that can feasibly be executed, given constraints on the availability of route clearance teams. A route clearance schedule and its associated risk-reduction metrics result.

We conduct several experiments on the methods developed to test its validity and applicability. Our first experiment examines the effects of mission timing on IED risk reduction, and shows the difficulty in relating this timing to our knowledge of IED risk in the road network. The second experiment demonstrates the trade-offs associated with assigning different sectors of the road network to different route clearance teams versus assigning all teams to the entire network. Our last experiment confirms the value of having convoy and patrol schedules available when conducting route clearance planning. We conclude that the planning method developed, integrated with a graphical control interface, would provide a useful decision support tool for military planners scheduling route clearance operations.

Thesis Advisor:	Professor Cynthia Barnhart Professor of Civil and Environmental Engineering Co-Director, Operations Research Center
Technical Supervisor:	Dr. Stephan E. Kolitz Member of the Technical Staff The Charles Stark Draper Laboratory, Inc.
Technical Advisor:	LTC Darryl K. Ahner Assistant Professor, Department of Mathematics United States Military Academy

[This Page Intentionally Left Blank]

Acknowledgments

I would like to express my appreciation to a few people who have mentored and supported me over the past two years. Their involvement helped shape this research and contributed to my professional growth as a student and as a career Army officer.

First, I thank those at the Draper Laboratory who were involved in this research. Dr. Stephan Kolitz simultaneously provided first-class supervision and abundant freedom to explore different approaches to the route clearance planning problem. Janet Lepanto and Milt Adams also supported this effort and provided sound guidance, which I greatly appreciate.

I also wish to express my appreciation to Lieutenant Colonel Darryl Ahner, who provided perceptive insight into the research problem and solution techniques from both military and technical standpoints. His input helped me to recognize aspects of the route clearance planning problem that I might have otherwise overlooked.

I owe much gratitude to Cynthia Barnhart at the Massachusetts Institute of Technology, who provided outstanding academic mentorship and research advice over the past two years. Her ideas were instrumental in the development of the methods of this thesis. I especially appreciate the time she dedicated to reading, listening to, and discussing my ideas, carefully considering them, and offering valuable feedback.

Finally, I thank my wife, Carol, and my daughter, Sarah, for the love and support they have given me as I have carried out this research. In addition to all of the efforts involved in providing a loving and healthy environment at home, Carol has also devoted her time in editing the first drafts of this thesis. I cannot thank her enough for her unwavering support.

This thesis was prepared at the Charles Stark Draper laboratory, Inc., under internal sponsored project funding and under contract W9124R-07-C-0027. Publication of this thesis does not constitute approval by Draper or the sponsoring agency of the findings herein. It is published for the exchange and stimulation of ideas.

As an active duty Army officer, I affirm that the views, analyses, and conclusions expressed in this document are mine and do not reflect the official policy or position of the United States Army, the Department of Defense, or the United States Government.

_____

Christopher E. Marks
Major, U.S. Army

[This Page Intentionally Left Blank]

Table of Contents

[This Page Intentionally Left Blank]

List of Figures

List of Tables

[This Page Intentionally Left Blank]

1 Introduction

In this thesis, we develop a technical approach for scheduling Route Clearance Teams (RCTs) to reduce effects of Improvised Explosive Device (IED) attacks during military operations. We also propose a method for integrating this technical approach into military planning and targeting processes. This chapter describes the research motivation, gives the problem statement, and provides an overview of the technical approach and experimentation methods employed.

1.1 Research Motivation

Improvised Explosive Devices have caused 40% of the casualties sustained by the United States in Iraq from April 2003 to April 2009 [icas09]. In Afghanistan, the number of IED attacks increased by 45% from 2007 to 2008, attaining their highest level since the war began in late 2001 [Br109]. These devices have been referred to as the "weapon[s] of choice for adaptive and resilient networks of insurgents and terrorists," which "threaten the safety of our service members deployed abroad as well as the long-term strategic interests of the United States and our allies" [JIE06]. As a result of the escalating use of IEDs in Iraq and Afghanistan, the United States Department of Defense established the Joint IED Defeat Organization (JIEDDO) in 2006 to lead efforts in overcoming the strategic effects of IEDs [JIE06].

IEDs in Iraq are often concealed along roads in order to attack passing military convoys. In Afghanistan, they are usually buried along routes used by the military for troop movements [Br209]. To combat these "roadside bombs", the JIEDDO began outfitting deploying units with equipment capable of detecting IEDs while keeping troops protected from their explosive effects. This equipment was organized into route clearance teams and manned by combat engineers and explosive ordnance disposal (EOD) personnel [JIE07].

Given this new capability to "clear" routes, military units were faced with the challenge of figuring out how best to employ it. Military operations in Iraq and Afghanistan involve small units operating over large regions. Some units are faced with urban areas, such as Baghdad, with thousands of miles of roadways. Other units operate in rural areas with only a few roads stretching for miles through unpopulated terrain. Cultural, economic, and social conditions as well as insurgent activities vary from region to region. These factors make each unit's IED

situation unique, and have led to different philosophies on the employment of RCTs to defeat their strategic and operational effects.

Unit staff personnel carry out military operational planning, including route clearance planning. These personnel are often faced with limited resources, competing priorities and not enough time. We focus especially on the route clearance planners, who have to make decisions on where and when to employ RCTs. These decisions are usually based on available information including what is known about recent IED attacks, recently conducted operations, planned operations, enemy intelligence, and the economic, political, and cultural atmosphere in the area of military operations.

IEDs are strategically effective because they are cheap, simple weapons that defeat technologically advanced ground combat systems and kill heavily armored troops. With IED attacks steadily increasing in Afghanistan and continuing to inflict casualties in Iraq, the United States and its allies will continue to confront IEDs in the foreseeable future. The methods proposed in this thesis facilitate automated route clearance plan generation with the goals of minimizing strategic effects of IEDs and increasing the planning capabilities of the staff.

1.2 Problem Statement

This thesis considers the *route clearance planning* problem, i.e., the problem of making planning and scheduling decisions for route clearance teams. We focus especially on planning conducted at the brigade and battalion level, and envision route clearance planners on these unit staffs as potential "users" of the methods we develop. Our goal is to produce a method that creates effective route clearance plans that minimize the strategic effects of IEDs. We consider "strategically effective" IEDs to be those that produce casualties or destroy major combat systems, although we leave the exact details of this definition to the user.

This problem requires us to develop an understanding of IED employment in the contemporary operational environment and route clearance operations and their effects. IED activities are not known in advance, so RCTs must be scheduled based on some measure of IED risk. The effectiveness of RCTs is another unknown. A particularly interesting consideration in this problem is the *enduring* effect of route clearance, i.e., how long a route can be considered "cleared" after an RCT clears it. How route clearance operations support and integrate with

other concurrent operations, especially those also meant to defeat IEDs, should be considered as well.

Ultimately, we seek an automated method that can be employed by the user as a decision support tool for scheduling and planning route clearance missions to defeat IEDs. This method should take advantage of computer technology to deal with the complexity of the problem, but should also be able to provide solutions quickly enough to give military planners more time to carry out other planning tasks.

1.3 Technical Approach

We employ operations research methods to find solutions to this problem. Operations research is "the discipline of applying advanced analytical methods to help make better decisions" [Bar06]. The specific methods we use are Markov processes, approximate dynamic programming, and mixed integer programming to model and solve this problem. These methods are useful because our problem is to support optimized decision making under stochastic, complex, and large scale conditions.

Markov processes prove useful in modeling IED events as well as other operational events that are not known in advance as stochastic Poisson arrival processes. They are especially useful in this application because of their decreasing dependence on initial conditions, which we use to represent the diminishing effects of a route clearance mission.

Dynamic programming deals with problems in which decisions can be made in stages, and the objective is to minimize some cumulative cost [Ber05]. Dynamic programming develops recursions that solve problems that evolve in time according to some underlying discrete-time dynamic system, such as a Markov process. Approximate dynamic programming uses the same techniques, but includes some simplifications to decrease the size or complexity of a problem in order to make it more tractable.

Mixed integer programming involves the minimization (or maximization) of some linear objective function subject to linear constraints, with integrality constraints on some, but not all, decision variables [BT097]. Many effective techniques have been developed to solve mixed integer problems. Mixed integer programming is useful in making cost-minimizing decisions for complicated problems with a large number of available choices.

1.4 Experimentation

To avoid any potential security concerns, we carry out our experimentation on notional data created to reasonably approximate the operational environment. We set up and conduct three experiments to evaluate the applicability and performance of our technical approach. The first experiment considers methods of finding the best times to execute route clearance operations. The second experiment examines the effects of dividing an area of operations into smaller regions to assign to different route clearance teams. The final experiment investigates the value of information about future troop and logistical movements in route clearance planning.

1.5 Thesis Organization

Chapter 2 provides an operational description of the IED problem, including explanations of IEDs and methods to defeat IEDs. The chapter also includes descriptions of the United States Army's organization and planning processes.

Chapter 3 takes a closer look at the Army's planning and targeting processes as they pertain to counter-IED operations. This chapter includes a functional decomposition of the route clearance planning problem.

Chapter 4 explains our modeling approach and the development of the route clearance planning algorithm. All methods employed are presented in detail in this chapter.

Chapter 5 provides the design, results, analysis, and conclusions of our three experiments, using the algorithm and methods developed in Chapter 4 on notional data sets.

Chapter 6 includes ideas for further research into the route clearance planning problem and suggests interface requirements for implementation of the route clearance planning algorithm as a decision support tool.

Chapter 7 provides a summary of our model and experimentation and concludes the thesis.

2 IEDs in the Contemporary Operating Environment (COE)

The goal of this chapter is to provide an overview of IED warfare and efforts to defeat IEDs in the contemporary operating environment.

2.1 Characteristics of the Contemporary Operating Environment

This section highlights some of the important aspects of the present-day military operational landscape.

2.1.1 Levels of War

United States (US) doctrine recognizes three levels of war: *strategic*, *operational*, and *tactical*. The strategic level of war describes the employment of armed forces to achieve national or multinational objectives. These objectives are the *strategic* objectives. The operational level encompasses campaigns and/or major operations (i.e. series of tactical actions) conducted in order to achieve strategic objectives. Examples of *operational* objectives include capturing major cities, taking control of critical resources, or rendering an opposing military incapable. The tactical level of war refers to the employment of units in combat. The object of these actions, e.g. to defeat an enemy force or capture an important terrain feature, are *tactical* objectives. These levels of war do not have clearly defined boundaries, especially in the COE, but they assist in clarifying the relationships between the combat actions of individual soldiers or small units (*tactical* actions) and the goals of the organization or nation (*strategic* objectives) [FM101]. We will make use of this terminology in our examination of the COE. Figure 2-1, which comes from Army Field Manual 3-0, depicts the relationships among the three levels of war [FM101].

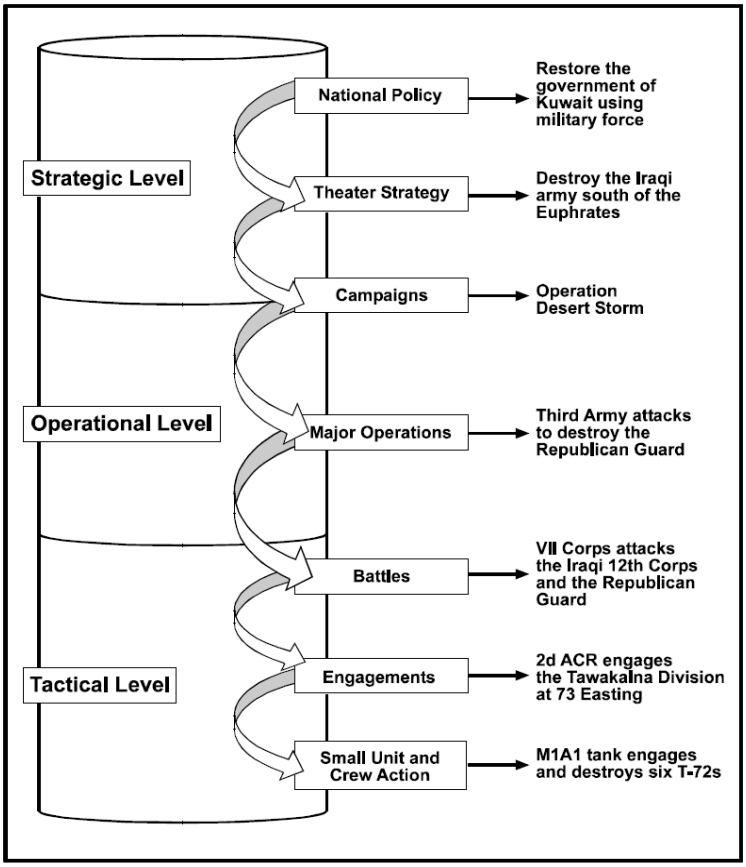

Figure 2-1: The Levels of War

2.1.2 Asymmetry

The COE is characterized by *asymmetric* warfare, or conflicts in which there is an imbalance in the ideology, technology, culture, economy, and/or military resources of the opposing forces [CAL02]. Under such conditions, each side attempts to leverage its strengths against its opponent's weaknesses. Conventional, force-on-force battle is replaced by a more protracted, unconventional struggle. Most current and potential US adversaries fall far behind the United States in their maneuver and firepower capabilities. These adversaries avoid (and will continue to avoid) confronting the US military in direct battle, instead relying on adaptive means to counter US material superiority without directly opposing it, while seeking to exploit perceived weaknesses [CAL02].

In asymmetric conflicts, the boundaries between the strategic, operational, and tactical levels of war are less clearly defined. It is not uncommon for a small-scale action involving only a few

troops to have far-reaching strategic implications. It is possible for a tactical success to achieve an operational or strategic setback (e.g. a successful raid might generate more local resistance, leading to less stability and security). Similarly, unsuccessful tactical actions can translate to strategic progress. Success in an asymmetric conflict thus requires careful nesting of tactical, operational, and strategic actions and objectives. By "nesting," we mean that tactical actions must be planned and executed in support of operational objectives, and likewise operations must be conducted so as to uphold, rather than conflict with, strategic aims. While nesting may seem intuitive, the complexity of asymmetric warfare creates conditions that can easily lead to conflicting tactical, operational, and strategic objectives.

2.1.3 Insurgent Threat

Contemporary United States adversaries employ asymmetric methods to offset US strengths. They make use of complex urban terrain and force dispersal to deny the United States the ability to mass its superior firepower. They seek to prolong the conflict, inflict unacceptable casualties, and erode public support, locally as well as globally. They remain highly adaptive in order to overcome developments in US and multinational capabilities [FM101].

In the ongoing conflicts in Iraq and Afghanistan, the adversarial forces have evolved into insurgencies. An *insurgency* is an "organized, protracted politico-military struggle designed to weaken the control and legitimacy of an established government, occupying power, or other political authority while increasing insurgent control [FM106]," and has been the common evolution of many asymmetric conflicts throughout history. To oppose an insurgency is to conduct *counterinsurgency* operations, defined as "military, paramilitary, political, economic, psychological, and civic actions taken…to defeat insurgency [JP101]."

2.2 Improvised Explosive Devices (IEDs)

Bombs often serve as ideal tactical weapons for insurgent forces. Employing bombs enables insurgents to anonymously create instability and lack of security, undermining the legitimacy of governing powers. They also enable insurgents to inflict casualties and cause damage without engaging powerful forces in direct battle and at a low cost. Recent conflicts in Northern Ireland and Israel provide good examples of the employment of bombs by insurgents.

In Iraq and Afghanistan, insurgent forces opposing the strategic goals of the United States have made frequent use of roadside bombs. These bombs, often referred to as "Improvised Explosive Devices," or IEDs, have become the most effective weapon employed against US and multinational forces. While they are somewhat less significant on the tactical level, they have proved to be very effective strategically, producing more US casualties than any other insurgent weapon system [Wes08] and damaging the legitimacy of the United States, multi-national forces, and the new Iraqi Government both locally and globally.

In addition to roadside bombs, the term "IED" has been applied to other types of insurgent bombings, such as car bombs or suicide vests. For the purpose of this thesis, we apply the term "IED" to refer *only* to roadside bombs targeting moving forces or civilians.

2.2.1 IED Tactics

Insurgents select places and times for IED attacks that will produce the best effects. Roadways, intersections, and other places that are frequented by US soldiers provide ample opportunities. The tactical goals of the insurgents are typically to produce casualties, harass and disrupt operations, and/or destroy equipment (all in support of their strategic aims). IEDs are often tailored to produce a very specific effect (e.g. penetrating the armor on a specific US vehicle), leading to many different types of IEDs. IEDs can be classified by their intended purpose, determined from their composition, and by their method of employment.

2.2.2 IED Composition

IEDs are comprised of an initiating system and a payload. The initiating system is designed to trigger the device when it receives a specific input. The payload (e.g., the explosives, shrapnel) is the part of the device that acts to produce the intended effects. Below is a summary of several of the more common types of initiating systems and payloads.

2.2.2.1 IED Initiating systems

Initiating systems are selected based on how and where an IED is going to be employed, its intended effects, and resources available. They are usually electrical and require a power supply (e.g., batteries). Following are a few examples of some common methods of IED initiation.

- *Remote-detonated (hard-wired).* An IED that is hardwired to a remote firing point can be detonated on command. This method of initiation is reliable but can be easy to detect, giving away the IED location and/or the insurgent's firing position. Because wire must be routed and serves as a physical connection between the insurgent and the IED, it can be difficult and risky to employ in crowded areas.
- *Remote-detonated (radio).* IEDs can be detonated remotely using virtually any kind of radio transmitter and receiver. These give the insurgent the advantage of being able to detonate an IED on command remotely without having to lay out any suspicious wire. However, electro-magnetic signals are subject to interference and jamming, and are therefore less reliable than a hardwired initiation system.
- *Victim-operated*. IEDs can be set up with a sensor that is remotely armed and detonates the device when it detects the target. These can be crude pressure sensors or passive infrared sensors. Often, more complicated IEDs such as armor-defeating IEDs will have this type of initiating system.

2.2.2.2 IED Payloads

The explosives and effects-producing components of IED can range from simple, crude designs to sophisticated ones. Some categories include:

- *Blast-Shrapnel effects*. These are the simplest IEDs that consist of an explosive charge and, often, some type of shrapnel. A paint can filled with home-made explosives, nuts and bolts achieves this effect. Military ordnance (e.g. artillery or mortar rounds, or mines) also fit into this category. These types of IEDs are generally effective against dismounted personnel and unarmored vehicles, but not as effective against armored targets.
- *Armor-defeating (directional) IEDs*. Some IEDs are designed to defeat armored vehicles. They are designed to somehow focus an explosion's energy to penetrate or destroy armored vehicles. A large amount of explosives buried under a road can produce this effect (because the road tamps the explosion, forcing it directly up into an unsuspecting target). Other, more technical methods require fewer explosives but more resources and precision to build. Shape charges are used to funnel explosive energy into a high pressure, high temperature gaseous dart in order to penetrate armor. Explosively Formed Penetrators (EFPs) use a similar concept to produce a high-kinetic energy metallic slug to punch through armor.
- *Vehicle-Borne IEDs*. Vehicle-Borne IEDs (VBIEDs) are vehicles laden with explosives. Provided there is a driver, this enables an IED to become mobile. In contrast to other types of IEDs, VBIEDs can be employed against stationary targets.
- *Decoy IED.* A suspicious device designed to appear as an IED, but not intended to operate as one is a decoy IED. Decoy IEDs do not produce casualties by themselves, but can be

employed in conjunction with other IEDs or direct fire, or merely as harassment. They also cause local instability by creating fear, stopping traffic, and intimidating local authorities or groups. The value of these IEDs demonstrate that tactical success (i.e., having the intended effect upon the intended target) is not always necessary to produce desired strategic effects (contributing to local instability, undermining US authority).

- *Other*. There are many other types of IEDs that have been employed or will be developed and employed by insurgents. As in the above cases, they are designed specifically with the purpose to produce certain tactical and/or strategic effects. The composition of an IED reflects its intended purpose.

2.2.3 Methods of Employing IEDs

In order to more fully understand the nature of IED warfare, one must consider *how* an IED is employed in addition to its composition and intended effects. Device positioning relative to its intended target, camouflage, incorporation of aiming/targeting devices (e.g. stakes or markers), and integration and positioning of any observers, triggermen, direct fire systems, or photographers are examples of some of the considerations that influence their tactical, operational, and strategic effects. They can also offer insight in how to defeat an IED or mitigate its effects. Figure 2-2 shows two pictures of IEDs [Glo09].

Figure 2-2: Examples of IEDs

2.3 Terrain

Terrain is the final feature of the military Contemporary Operating Environment we will consider. While ridgelines and hilltops often dominated the conventional battles of the past, civilian populations and infrastructure have become contemporary key terrain features. Insurgencies rely on some measure of support from local populations, while governing powers ultimately derive their legitimacy from the governed. The differences between urban areas and rural areas provide unique advantages and disadvantages to each opposing force.

For the purpose of IED warfare, the road network is the critical aspect of terrain. Government or occupying forces must rely on roads to conduct the regular ground movements of troops, equipment, and supplies necessary to conduct large-scale operations. In urban areas or on mountainous terrain, vehicular movement is often restricted to roadways, severely limiting ability to maneuver and providing ideal locations for attacks and ambushes. Civilians (and

insurgents) must also rely on the same road network in meeting their employment, religious, or general livelihood requirements. Roadway width, visibility, debris/litter, surface construction, proximity to buildings, traffic conditions, and intersection types all significantly influence insurgency and counterinsurgency operations, especially IED operations.

2.4 The United States Army

The remainder of this thesis examines IED warfare from the perspective of the United States Army. In order to understand and better analyze this perspective, we first review the structure of the Army and some of its operational processes.

2.4.1 Tactical Organization

At the tactical level, the US Army is organized into deployable corps. A *corps* generally consists of two or more combined arms (i.e., combat) divisions and additional support, reconnaissance, engineer, and intelligence elements. There is no standard corps organizational structure; corps are tailored to meet the mission requirements placed on them. Each *division* is similarly organized into two or more brigade combat teams (BCTs) and comparable, though less robust, support elements. BCTs are organized into battalions; battalions are organized into companies, with fewer support elements in lower echelons. Figure 2-3 [FM196] shows a typical corps organization and demonstrates the range of assets available in a corps other than combat arms divisions. Figure 2-4 [FM396] depicts the organization of a typical combat division.

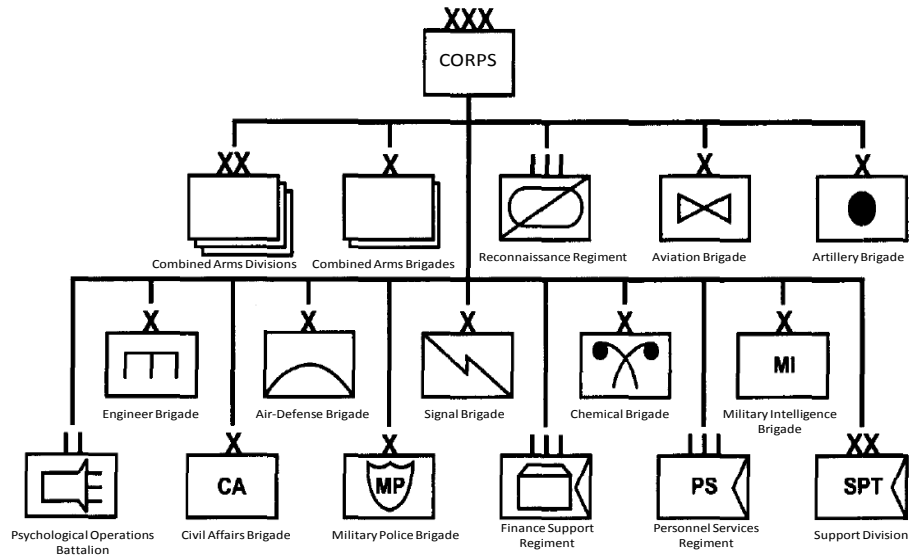

Figure 2-3: Typical Corps Organization

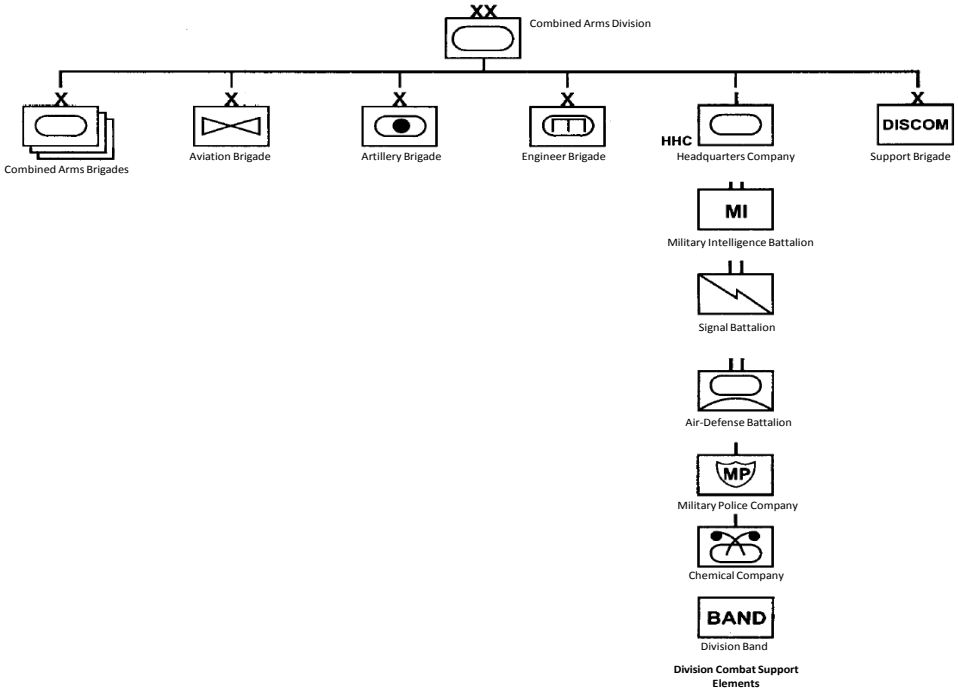

Figure 2-4: Typical Combined Arms Division Organization

2.4.2 Military Decision Making Process

In addition to having more assets to manage, higher echelons also have more robust staffs to conduct operational planning, command and control activities. Figure 2-5 shows a typical corps staff organization [FM196]. Boxes within this figure represent staff sections of varying sizes (e.g. G2: Intelligence, G3: Operations, CH: Chaplain, etc.). Each of these elements performs a specific function that contributes to the overall management of the unit. The staff is responsible for actions such as maintaining situational awareness, managing information, developing and disseminating orders, controlling and assessing operations, and coordinating with higher, lower, and adjacent units [FM206], as well as handling routine administrative affairs.

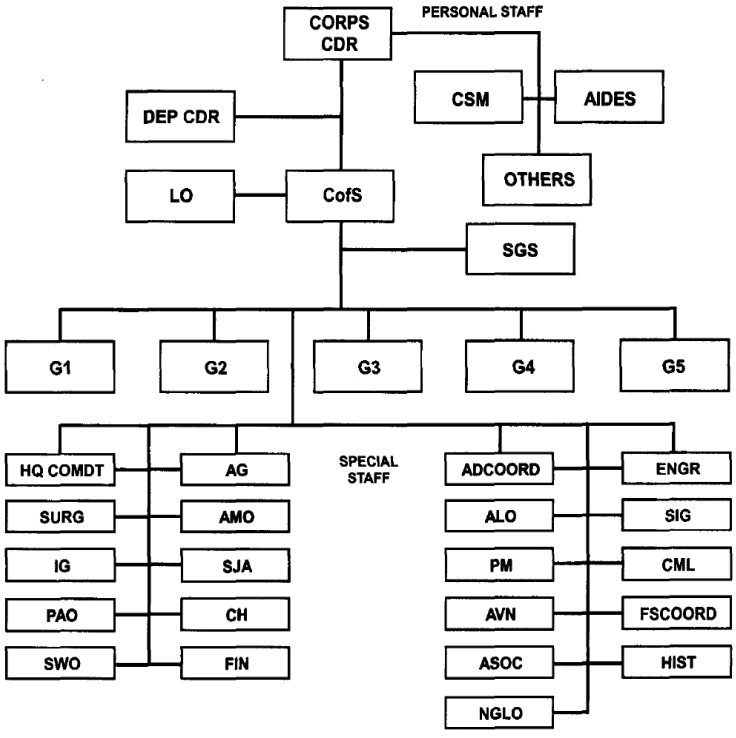

Figure 2-5: Typical Corps Staff Organization

In order to plan and carry out operations efficiently, unit staffs execute the Military Decision Making Process (MDMP). Figure 2-6 [FM105] illustrates the steps of the MDMP and shows the unit staff's and commander's involvement. The MDMP begins when the staff receives a mission

from its higher echelon headquarters (step 1). The staff then executes "mission analysis" (step 2). In conducting this important step, the staff works together to review all of the assets and capabilities available within the unit (e.g. all elements within a corps/division/BCT and their capabilities), assess all information known about the enemy assets, capabilities, and likely courses of action, and thoroughly examine the military aspects of the terrain over which the mission and supporting actions will take place. The intelligence officer, with assistance from other staff members, combines the analysis of the enemy, terrain and weather in his Intelligence Preparation of the Battlefield, or IPB (a sub-process that continues throughout the MDMP). The staff also creates a list of all tasks specified or implied in the mission order it has received, and determines which tasks must be completed in order to accomplish the given mission. The processes and products of the mission analysis provide information crucial to the rest of the decision process. Following mission analysis, the staff moves on to course of action (COA) development (step 3). It is during this step that staff members propose ways to allocate the resources available to accomplish each of the tasks gleaned from the mission order during the mission analysis. After carefully analyzing the costs and benefits of each COA (step 4) and comparing them against one another (step 5), the staff presents its results to the commander who approves a specific COA (step 6) or provides more direction to the staff. Once a COA has been approved, the staff produces an operations order (step 7) that states the scope of the unit's mission and provides specific tasks to each subordinate element. Subordinate units use this operations order in conducting their own MDMPs.

In addition to the operations order, the MDMP calls for the issuing of warning orders (WARNOs) after steps 1, 2, and 6. These WARNOs contain information relevant to the upcoming operation, including meaningful developments and decisions that have been made during the MDMP. WARNOs enable subordinate elements to conduct preliminary preparations and planning ahead of receiving the operations order.

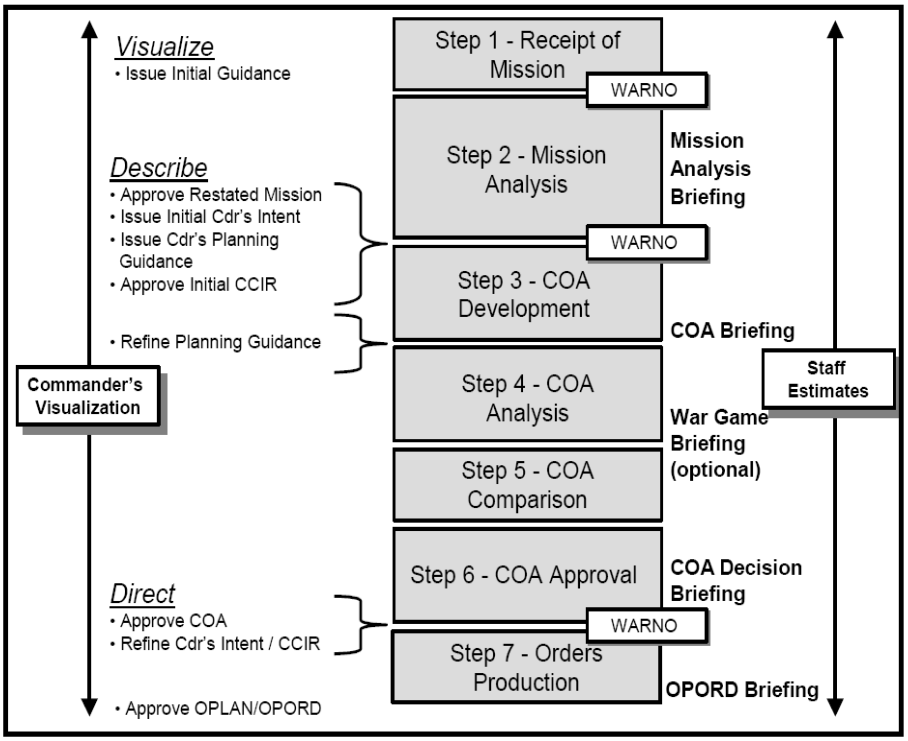

Figure 2-6: The Military Decision Making Process

2.4.3 Targeting

Targeting is the process by which military units identify and engage enemy or neutral personnel or equipment in order to achieve some benefit toward accomplishing the unit's mission. It is a "complex and multidisciplined effort that requires coordinated interaction among many groups," including fire support, intelligence, operations, and planning personnel [FM296]. The targeting process typically begins within the MDMP and continues throughout the duration of an operation. Targeting consists of four major activities [FM106]:

- *Decide*. The commander and the staff, especially the intelligence personnel, continuously analyze the unit mission and the current situation to determine what targets exist, how they should be engaged, and what priorities exist in the allocation of time and resources to a particular target. The IPB produced during the mission analysis and continuously updated is

a very useful input in this activity. This activity is performed using the tools and steps of the MDMP.

- *Detect*. Within the context of the targeting process, *detection* is the allocation and employment of reconnaissance and surveillance assets to determine or verify the target's validity, importance, best means of engagement, and effects of engagement, as well as any changes to previous information [FM104]. This activity also requires further analysis of available information and intelligence.
- *Deliver*. This action refers to the tactical planning and execution of missions against targets. It involves using tactical assets to engage targets in time and space.
- *Assess*. Assessing the effects of each operation provides important feedback to the other three targeting activities. New information or changes in conditions resulting from targeting operations often cause re-prioritization of targets as well as reconsideration of engagement techniques and effectiveness. These operational assessments provide important input into future targeting and often motivate future operations.

Figure 2-7 [FM296] depicts the Army Targeting methodology as well as how it integrates into the steps of the MDMP.

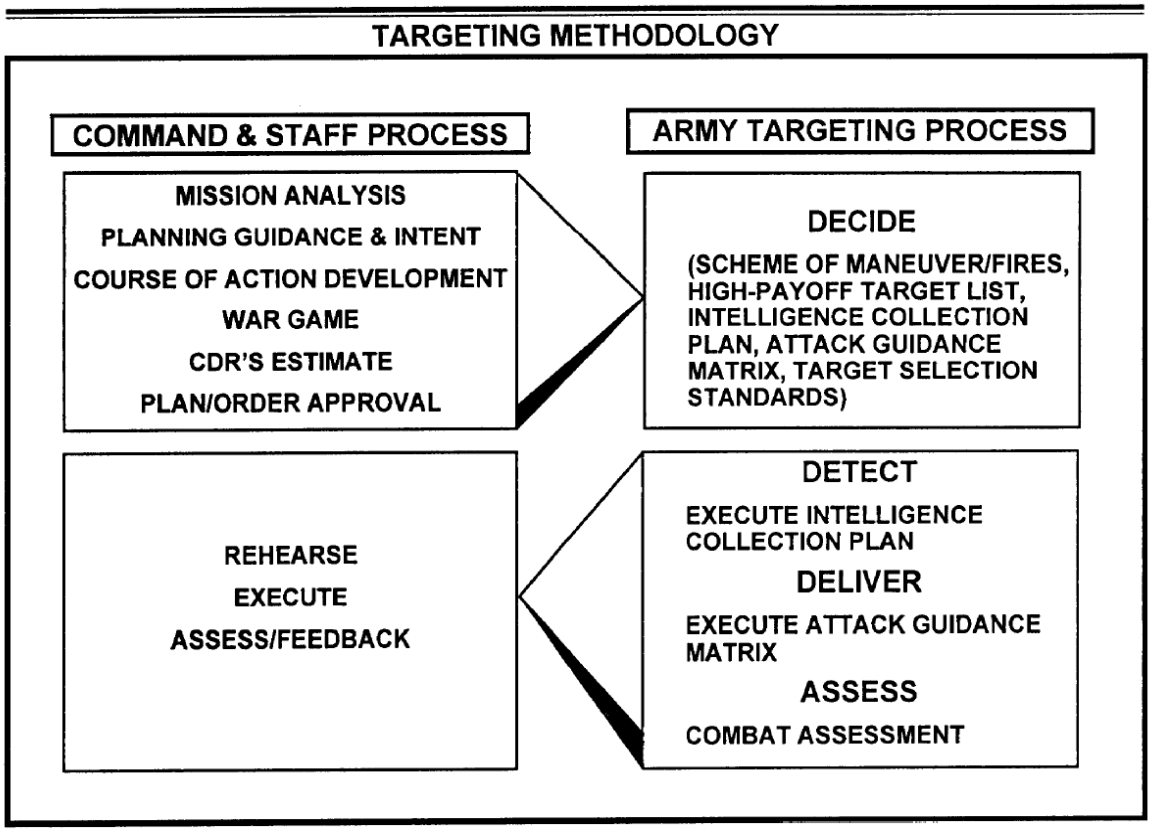

Figure 2-7: Targeting Methodology

2.4.4 Stability Operations

In contemporary (asymmetric) conflicts it is not uncommon for units to transition from "high-intensity" combat to conducting *stability operations*, consisting of actions that promote and protect US interests by influencing the threat, political, and information dimensions of the operational environment [FM101]. During stability operations, a unit may typically execute one mission for an extended time period. For example, a BCT operating in Iraq could spend its entire tour working to provide stability and security in its assigned area of operations (AO). In these types of operations, the MDMP and targeting process are applied continuously. A unit typically develops a *battle rhythm*, "a sequencing of standardized command and control activities within a headquarters and throughout the force to facilitate effective command and control [FM206]." This battle rhythm ensures the MDMP and targeting processes are synchronized and ongoing by providing a predictable schedule of update briefings, meetings, and working groups

to carry them out. Throughout the execution of its battle rhythm, a unit measures its progress, reviews guidance and orders from higher headquarters, analyzes new information, makes decisions about current or future operations, and issues orders to subordinate elements. Figure 2-8 [FM206] shows an example daily/weekly battle rhythm.

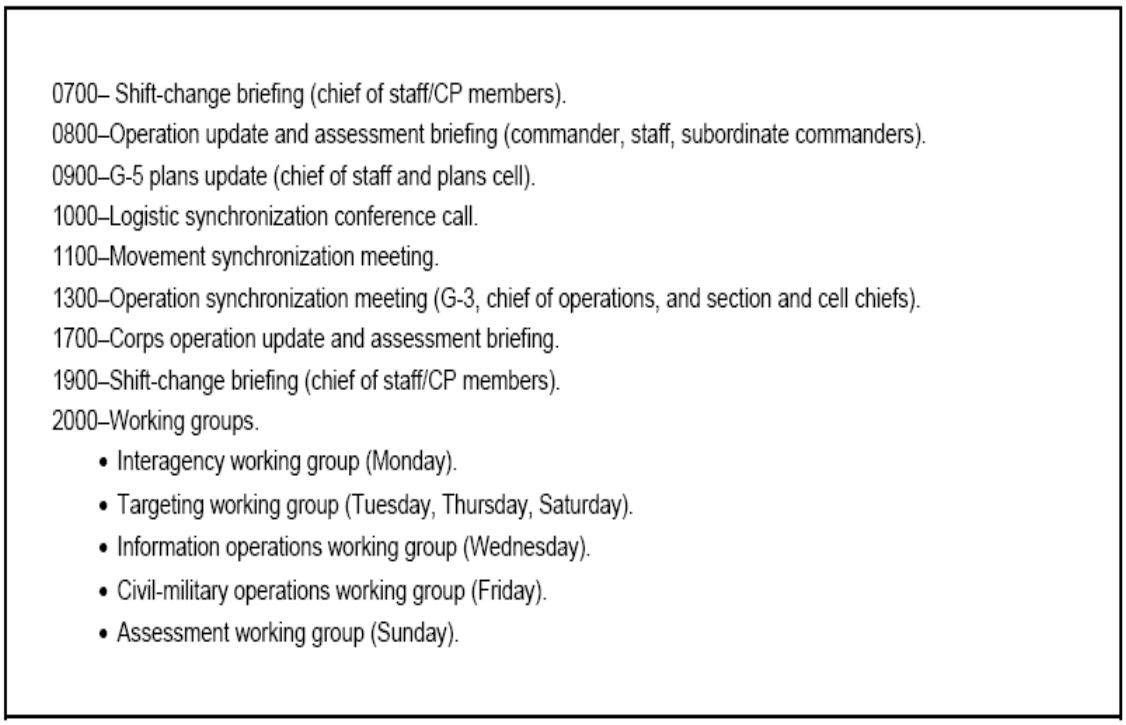

Figure 2-8: Sample Unit Battle Rhythm for Stability Operations

2.5 Counter-IED

Within the ongoing counterinsurgency campaigns in Iraq and Afghanistan, the United States and its allies conduct *counter-IED* operations to reduce or eliminate the strategic effects of insurgent IED warfare. The US Department of Defense has divided its counter-IED strategic approach into three simultaneous efforts, depicted in Figure 2-9 [Mac06]:

- *Train the Force*. This effort consists of comprehensive training prior to and during deployment on how individual soldiers and units can avoid IEDs and mitigate their effects.

- *Defeat the Device*. This effort includes actions that focus specifically on neutralizing the effects of Improvised explosive devices. Locating and neutralizing IEDs, increasing armor on troops and equipment, and routing patrols to avoid IED hazard areas are all ways to defeat the device.
- *Defeat (or Attack) the Network*. Attacking the network refers to operations conducted to prevent insurgents from emplacing IEDs, or to remove their capability. Such actions include raiding supply points or capturing insurgent IED emplacers, IED builders, financers, triggerman, and leadership. These operations are often not fully differentiable from other counterinsurgency actions, sharing the same objectives.

The remainder of this chapter focuses on tactical doctrine and methods employed by the US Army to defeat the device, concentrating on actions taken during or after IED emplacement in order to mitigate its effects.

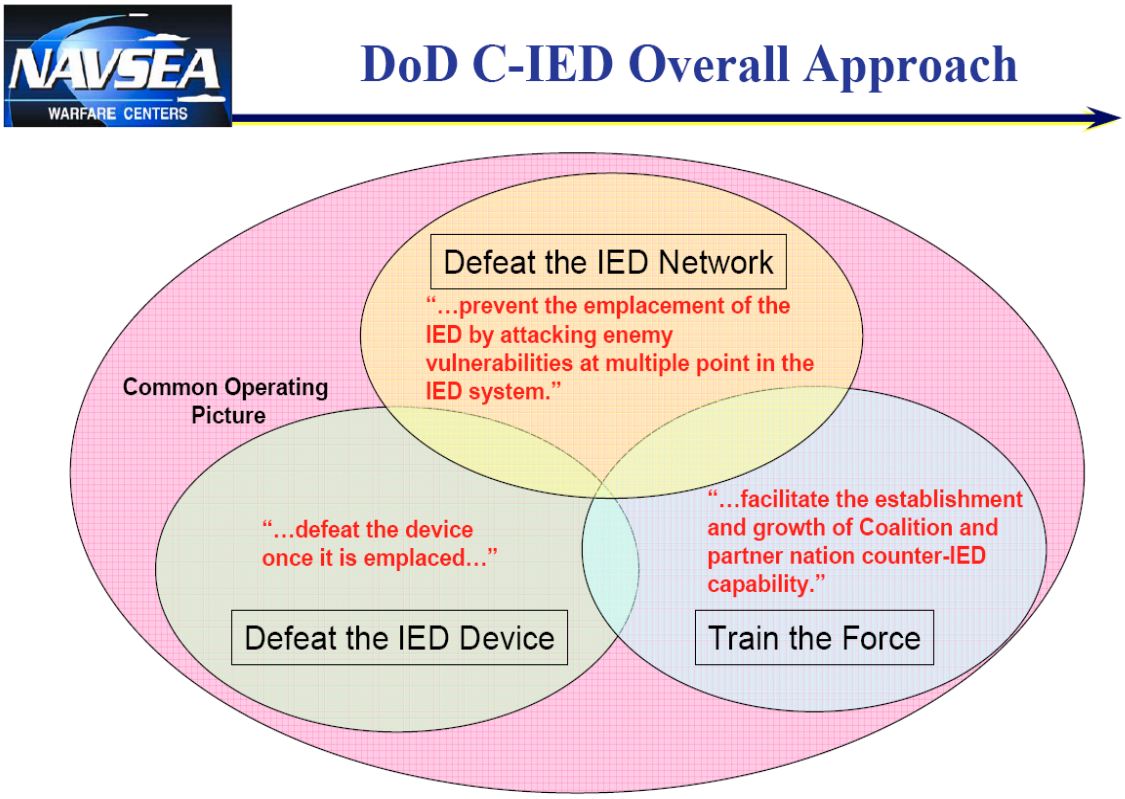

Figure 2-9: Counter-IED Strategy

2.5.1 Doctrine of Assured Mobility

Maintaining *mobility*, or the ability to continue movement, is generally considered a critical task in any ground operation. Traditionally "mobility" has referred to the capability to reduce field fortifications and move across obstacles, such as mine fields, barbed wire fences, walls, ditches, or rivers, with Army Engineers responsible for completing mobility-related tasks. Recently the US Army developed the doctrine of *assured mobility*, which applies to actions that enable forces to deploy, move, and maneuver freely and without disruption or delay to accomplish their missions [FM104]. Because IEDs constitute one of the largest perceived threats to mobility within the COE, the doctrine of assured mobility provides tactical basis for conducting counter-IED operations. The fundamentals of assured mobility are [FM104]:

- *Predict* possible mobility inhibitors.
- *Detect* any indicators of impediments to mobility.
- *Prevent* potential impediments to mobility through early action or intervention.
- *Avoid* detected or suspected impediments to mobility.
- *Neutralize*, breach, or reduce impediments to mobility that cannot be avoided.
- *Protect* against enemy counter-mobility effects.

We examine below how each of these fundamentals is applied in IED warfare to "defeat the device."

2.5.1.1 Predict

In order to defeat an IED (during or after emplacement) one must first have some knowledge that the threat of IEDs exists. Generally, we know that the threat of IEDs exists within the COE, specifically in Iraq and Afghanistan. However, the US does not have the assets to examine or watch over every section of roadway in either of these theaters. Thus unit staffs come up with methods to determine the most probable locations and times of enemy IED emplacement. They also determine locations and times for which IED emplacement would be most hazardous to their mission and objectives, and assess the probabilities of these "most dangerous" IEDs. Military Intelligence officers within the unit staff are responsible for determining these "most likely" and "most dangerous" enemy courses of action and briefing them to the commander and the planning staff. One tool used to predict attack likelihood is "pattern analysis" [Mag05], an examination of

records of past attacks for any predictable recurrences. Accurate IED prediction sets the conditions for successful IED prevention, avoidance, detection, and neutralization operations [Mit04].

2.5.1.2 Detect

Specialized route clearance teams (RCTs) are the primary IED detection assets. Route clearance teams consist of specialized equipment and personnel designed to detect IEDs while minimizing the risk to personnel and critical equipment. This equipment can include heavily armored mine detection equipment such as the Buffalo or Meerkat, armored personnel carriers or observation vehicles like the RG-31, robotic equipment, optical devices (such as those on the gun sights of combat vehicles), and more [Mit04]. These patrols move along designated routes searching for IEDs. When an IED is found, the area is secured and an Explosive Ordnance Disposal (EOD) team diffuses the device. In most cases, IED explosions do not have catastrophic effects on these patrols. These patrols are slow-moving and require significant resources. Because of equipment and human constraints, they are effective for a limited amount of time before the equipment must be re-fueled and maintained and the personnel must be rested. They are only successful when they find IEDs within the short time window between emplacement and execution. Route clearance is the primary method used to defeat emplaced IEDs. Figure 2-10 [Buf09] shows the Buffalo, one of the primary route clearance vehicles.

Figure 2-10: Buffalo Mine-Clearance Vehicle

Maneuver patrols or logistic convoys can also detect IEDs, but assume a higher risk and are not optimally equipped to perform this task exclusively. Unmanned aerial platforms and other surveillance assets could also be useful in detecting indicators of IEDs (such as an emplacement team or disturbed terrain), but probably would not have the fidelity to confirm IED presence in most cases.

2.5.1.3 Prevent

There are many ways to prevent successful IED execution during or after emplacement. Here we present three methods [Mag05]:

- *Frequent patrolling* deters IED emplacement by denying insurgents time to prepare the site and emplace the device. Patrolling requires significant assets in vehicles, weapon systems, and personnel. This method is more effective when the patrols are more frequent, but it also exposes patrols to the IED risk they are trying to mitigate. If not completely effective as a deterrent, frequent patrolling can have the effect of forcing insurgents to find hasty, discrete methods to emplace IEDs.

- *Observation posts* are overt military presences along routes, preventing IED emplacement along the observed portion of the route. Observation post effectiveness is limited to the portion of the route observed, and they generally require a significant amount of assets to operate, secure, and sustain.
- Placing *snipers* in positions where they can covertly watch over important or highly volatile segments of road can help prevent IED emplacement. The sniper uses direct fire to destroy any observed IED emplacer. The IED prevention is limited to the stretch of road that the sniper can observe. Sniper positions are difficult to occupy, conceal, and secure for long periods of time, especially in urban areas, and a sniper at long range might have trouble discerning IED activities from other, non-hostile activities. Finally, a sniper or sniper team can only remain for one engagement; after firing, the position is compromised and the sniper(s) must exfiltrate.

2.5.1.4 Avoid

Predicted or detected IED locations can be avoided by re-routing military traffic. Commanders and operations officers make decisions on which routes to use and which routes to make off-limits within their units. Known IED locations are often secured in order to prevent any traffic from traveling within its effective radius.

Many units also take steps to decrease the number of military movements through potentially hostile terrain. Meetings are held via communications networks, and some personnel and supplies can move via aircraft.

2.5.1.5 Neutralize

Once an IED has been detected, either by a route clearance patrol or by other means, Explosive Ordnance Disposal (EOD) specialists defuse it. Other methods to neutralize IEDs, such as with demolitions or direct fire, are possible but less preferred because they often involve more risk.

2.5.1.6 Protect

There are many ways to protect soldiers and equipment from the effects of IEDs. We briefly discuss three specific methods: armor, electronic jamming, and tactical movement techniques. Protecting vehicles and troops with additional armor decreases the effectiveness of IEDs, but too much armor can slow or inhibit movement and increase required maintenance and rest time. Electronic jamming equipment can disrupt radio signals used to initiate some IEDs (recall the

different types of initiating systems) but also increases vehicle weight, consumes limited space, and can interfere with tactical communications equipment [Put08]. Finally, units have developed a variety of movement techniques and tactical procedures to protect troops and equipment from the effects of IEDs. These tactics often become less effective over time and must be continuously updated.

2.6 Difficulties in Defeating the Device

Operations to defeat the device, including route clearance, are typically planned and synchronized by military staffs and units using the military decision making process and the targeting process in accordance with their established battle rhythms. For example, there might be a weekly meeting in which personnel from the staff (e.g. intelligence, operations, logistic, and engineer planners) meet, consider available information on likely future IED attacks, planned operations, and priorities, and determine where and when to allocate their route clearance teams in order to prevent IEDs from achieving tactical, operational, or strategic effects [Mit04]. Below we briefly mention some of the difficulties that arise in allocating assets to defeat the device.

- *Coordination among methods and echelons*. For example, route clearance is usually performed by engineers, while observation posts (or other counter-IED operations) are conducted by infantry, armor, or military police troops. The result is that assured mobility efforts might be planned and executed by different echelons, or by different staff sections within the same echelon. Without coordination these efforts can overlap resulting in a waste of resources, frustration, and potentially dangerous situations during mission execution.
- *Prioritization*. Route clearance assets are limited and cannot continuously clear the many miles of roadway in the COE. These assets must be configured into patrols and assigned to routes during times that provide the highest payoff. Determining which routes and times provide the highest payoff is difficult and must be estimated based on known information about insurgent activity, ongoing US and allied operations, and the terrain.
- *Quantity of Data*. In developed theaters of operations, analysis of stored information, such as reports and significant events, is a complicated task. Different kinds of reports are likely to be stored in different databases or unique locations (e.g., a supply report, a report on an attack, and an intelligence report are not likely to be stored together). There might also be other data available. If military vehicles are equipped with GPS tracking systems (as many US military vehicles are), statistical data can be available on operational movement patterns. Considering the excessive amount of data that can be gathered, maintained and stored using

computer technology, and determining what data is important can be an overwhelming task for a small military staff with limited planning time.

- *Measuring effectiveness*. Planners have limited ability to assess the effectiveness of route clearance or other assured mobility efforts, and to predict effectiveness of future operations. In an operation such as route clearance, there is no good way of knowing whether a patrol failed to detect an IED. If an IED incident occurs following a route clearance mission, it could be that the route clearance patrol failed to detect the device, but it could also be that the device was emplaced immediately after the patrol cleared the area. Even in the latter case, one could question the effectiveness of the patrol, and even the very notion of what a desirable "effect" is. Normally some level of effectiveness must be assumed, based in part on available information.
- *Change*. The final difficulty we mention is the dynamic nature of the battlefield. New data is always arriving, operations are continuously planned, commander's priorities are constantly being updated. In some cases, the time it would take to complete a detailed analysis of available information would render the analysis obsolete by the time it was completed. Planners must be able to produce flexible plans and be prepared for a wide range of situations that might arise.

3 Route Clearance Targeting

The goal of this chapter is to analyze the functions involved in the planning and execution of route clearance missions to detect IEDs. The targeting process will provide the framework for this analysis, and the military decision making process (MDMP) and fundamentals of assured mobility will help to identify some of the specific functions that occur within each of the four major targeting activities.

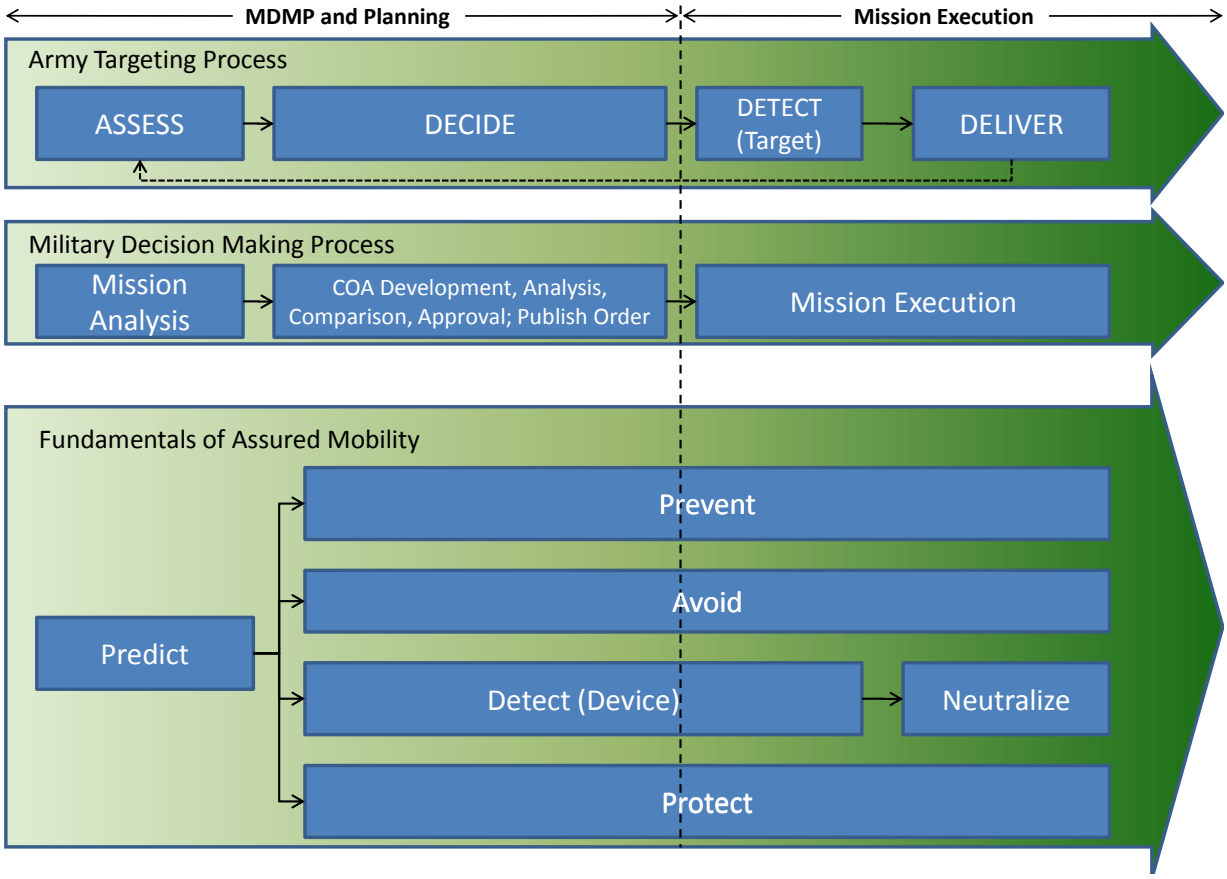

Figure 3-1: Integration of The Targeting Process, MDMP, and Fundamentals of Assured Mobility to Defeat the Device

3.1 Defeating the Device: A Functional Overview

Figure 3-1 depicts the synchronization of the targeting process, the MDMP, and the fundamentals of assured mobility in the context of defeating the device. One shortcoming of this figure is that it depicts clearly defined boundaries between the steps in the MDMP, the targeting activities, and the fundamentals of assured mobility, whereas in reality there might be significant

overlap. For example, the *protect, detect, and neutralize* fundamentals of assured mobility all have considerable overlap, that is not shown (detection assets are normally protected with heavy armor and can have the capability to neutralize). Another example is the *mission analysis* step of the MDMP, which does not necessarily end with COA development but continues at some level throughout the planning and execution of an operation. The figure nonetheless provides insight into the integration of the targeting process and MDMP in operations to defeat the device. It shows how efforts to detect IEDs (e.g., Route Clearance) are planned and executed in parallel with other efforts to defeat the device using the steps of the MDMP and the targeting activities. We see that the *predict* fundamental of assured mobility is part of the MDMP's *mission analysis*, which (typically) falls under the *assess* targeting activity. The remainder of the fundamentals of assured mobility are incorporated into the courses of action generated by the MDMP (*decide* targeting activity) and carried out in execution of the approved mission plan (*detect* and *deliver* targeting activities). Because of the associations depicted, our analysis of the *detect* fundamental of assured mobility will require some understanding of the functions involved in the *predict* and *neutralize* fundamentals. This figure provides a frame of reference for the functional analysis in this chapter.

3.2 Functional Decomposition of Route Clearance Targeting Activities

We now analyze the functions that comprise route clearance operations in the context of the targeting process, using the simple "input-output" functional model. Figure 3-2 depicts the Army targeting process for "defeat the device" operations, with an emphasis on route clearance. The process is motivated externally by a mission to conduct military operations in an environment in which enemy forces employ IEDs.

One characteristic of route clearance targeting is the importance of the *detect* targeting activity. In most targeting operations, such as when individuals or military equipment serve as "targets", the process reaches its culmination in *delivery*, or target engagement. When the targets are IEDs, *detection* becomes the critical task because, unlike other military targets, once an IED has been found it is usually comparatively easy to render ineffective through avoidance or remote neutralization.

The targeting process is cyclic. We begin the functional breakdown that follows with the *decide* activity, which inputs information from the current mission analysis, including targeting assessments from ongoing or recent operations. Figure 3-1 depicts this relationship, with the *assess* activity appearing to the left of *decide* and aligned with the mission analysis step in the MDMP. Figure 3-2 better depicts the cyclic nature of the targeting process. While we mention operational assessments as inputs into the *decide* targeting activity, we discuss the *assess* targeting activity in more detail in paragraph 3.2.3.

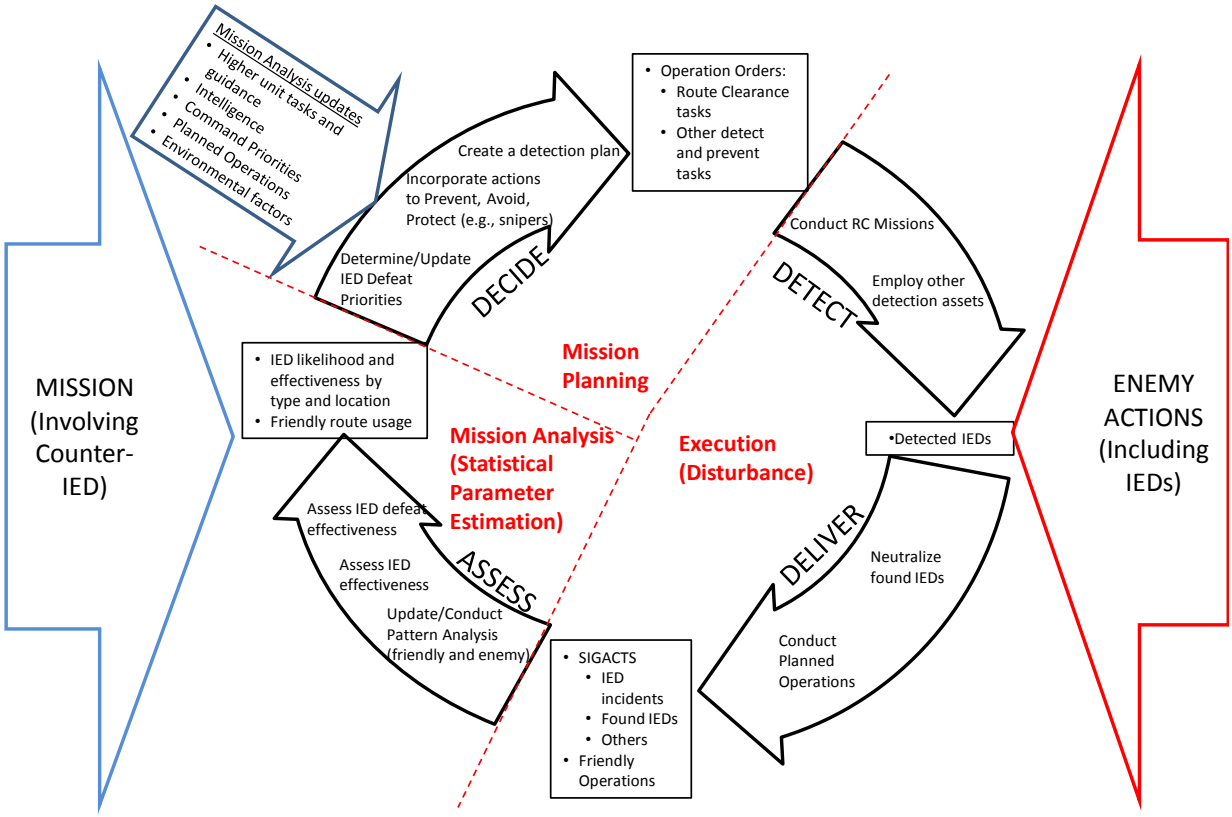

Figure 3-2: The Route Clearance Targeting Process

3.2.1 Decide

With respect to route clearance operations, the *decide* targeting activity carries out the last five steps of the MDMP: COA development, COA analysis, COA comparison, COA approval, and orders production. The updated mission analysis, including analysis of the effects of recently conducted or ongoing operations (from the *assess* targeting activity) and analysis of new

information (such as new requirements from a higher headquarters, changes in priorities, intelligence, planned operations, or environmental developments), serves as the primary input in the *decide* targeting activity. The output of this targeting activity is a route clearance plan, normally in the form of a published schedule assigning RCTs to clear specific routes at certain times. We now consider each of the five processes within the *decide* targeting activity individually, focusing on their applications in route clearance operations.

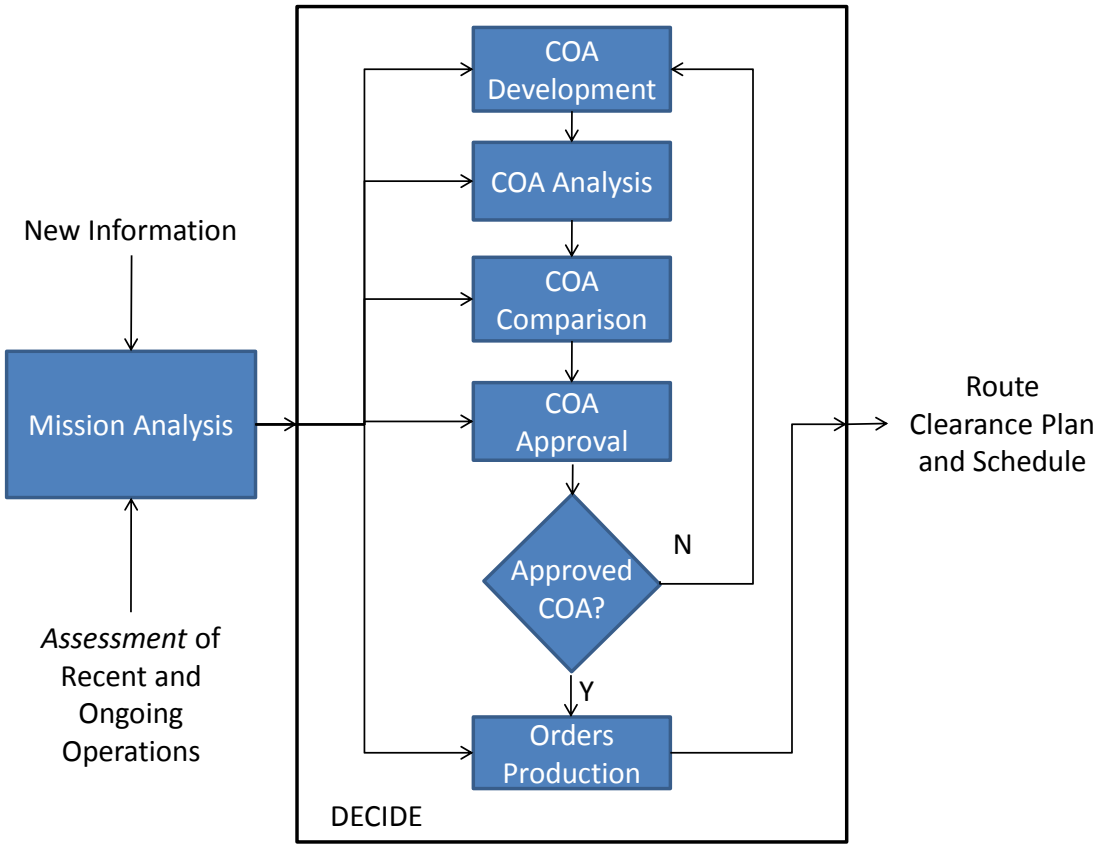

Figure 3-3: Inputs, Outputs, and Processes of the Decide Targeting Activity

3.2.1.1 Course of Action (COA) Development

The inputs to COA development are the outputs from the mission analysis step of the MDMP: most likely and most dangerous enemy courses of action, friendly assets and capabilities, terrain analysis, and an initial mission statement and commander's intent. The current mission analysis is a fusion of new information (e.g., intelligence or operation orders) and the outputs from the *assess* targeting activity. It provides the COA development function with:

- *Most likely and most dangerous enemy courses of action*. For route clearance operations, these courses of action consist of information on potential IED emplacements in the area of military operations. This information ideally would include some measures of the likelihood and associated danger of specific IED attacks at locations and times within area of interest. "Most dangerous" and "most likely" IED attacks would be easy to glean from such data. Information about enemy capabilities and constraints, such as supplies available, is also included.
- *Friendly Assets available*. This input includes the route clearance teams available and the composition and capabilities of each. Some of these capabilities are IED detection effectiveness (perhaps for different types of IEDs or roads), clearance speeds, and mission availability.
- *Terrain analysis*. Analysis of the roadway conducted during the mission analysis provides the size and shape of the road network and certain aspects of each of the routes. The size and shape of the network is described by the routes and their orientations, intersections, and lengths. Important aspects of a route might include road width, surface type, shoulder construction, bridge or overpass presence, and other features that could have an impact on military operations.
- *Initial mission statement and commander's intent.* Any command guidance that pertains to defeating IEDs is an input to COA development. This guidance could include priorities of effort, critical friendly assets to protect, or an overall route clearance objective, such as to minimize IED casualties or to keep specific routes cleared.

The process of COA development involves allocating available assets to carry out assigned tasks in a synchronized way so as to accomplish the unit mission and the commander's intent. Knowing the capabilities and availability of each RCT and the likelihood and risk associated with enemy IEDs, detection tasks are identified and allocated to RCTs in accordance with the guidance and intent of the commander. A course of action is a feasible allocation of tasks to all RCTs that meets the commander's intent and supports the overall unit mission. The number of COAs generated can vary based on external considerations such as available time, planning guidance, and the nature of the mission.

The outputs of this function are the generated COAs.

3.2.1.2 COA Analysis

The COA analysis function receives as its input all of the inputs and outputs from COA development, along with the current mission analysis. COA analysis uses the information from the mission analysis to evaluate each generated course of action.

This function's process involves individually "wargaming" [FM105] each COA. Each RCT is presumed to conduct its assigned tasks according to the COA being analyzed. These actions are considered along with the IED emplacers' most likely and most dangerous courses of action. Based on the capabilities and effectiveness of the RCTs and known information about IED likelihoods and risks, outcomes are predicted for each COA. Metrics are created to measure how well an outcome accomplishes the mission and commander's intent, preserves resources, and/or minimizes risk. The predicted outcome for each COA is evaluated according to these metrics.

The outputs of COA analysis are the evaluations for each COA.

3.2.1.3 COA Comparison

Like COA development and COA analysis, the COA comparison function inputs the most updated products of the mission analysis. It also inputs the courses of action and their respective evaluations from the COA analysis function.

COA comparison examines the evaluations of each COA and compares them against each other. Evaluation metrics can be weighted to emphasize aspects that are more closely tied to the mission or commander's intent. A decision matrix or similar tool can be used for quantitative comparisons.

This function outputs relative advantages and disadvantages for each COA, along with a recommended (or best) COA that is justified by the evaluation metrics.

3.2.1.4 COA Approval

The inputs to this function are the mission analysis products, the recommended COA, as well as all the COAs with their associated metrics, advantages, and disadvantages.

The COA approval process is not easy to characterize. At the sole discretion of the unit commander, the recommended COA can be approved, another (non-recommended) COA can be

approved, or additional guidance can be issued as a new input and the process returned to COA development, COA analysis, or COA comparison.

The output of this function is either a COA decision (with or without additional guidance) or additional guidance and direction to return to previously completed functions.

3.2.1.5 Orders Production

The orders production function inputs the approved COA, including its evaluation metrics and associated commander's guidance, and the updated information from the mission analysis.

This function organizes the available information from the mission analysis, the tasks and synchronization information from the approved COA, the commander's intent, the unit mission, and the operational support and communications requirements into a standard format for communication. Information about IED likelihood and risk will be contained in this format, along with directives to each RCT to clear specific routes at specific times, which we refer to as "route clearance missions."

The output of orders production is the operations order. It is normally written but in rare cases may be issued verbally. For operations that include route clearance, this order includes directives that schedule RCTs to clear specific routes at specific times.

3.2.2 Detect and Deliver

The *detect* and *deliver* targeting activities comprise a single "mission execution" function. Within this function IED incidents (detections and explosions) occur, confirming or disproving the predictions made during mission analysis. Note that the detect activity in the targeting process can integrate many different types of assets, e.g., unmanned aircraft, scouts, or informants. In operations intended to defeat the device, RCTs are the primary detection assets, so we limit our discussion of detection to route clearance only (i.e., the *detect* fundamental of assured mobility). Similarly, we consider IED neutralization as the *deliver* activity in the route clearance targeting process.

The mission execution function involves three inputs: (1) the route clearance input, which includes the route clearance plans from the *decide* targeting activity, (2) the enemy input,

including the set of unknown insurgent IED emplacements, and (3) the operational input, consisting of all convoys and movements on the road network.

Within this function the planned route clearance missions are carried out; routes are cleared by assigned RCTs at designated times. Other operational convoys and missions are also executed. Emplaced IEDs that are detected prior to employment are neutralized. IEDs that are not detected by an RCT are "employed" against convoys or other military movements when they arrive at the emplacement location. The success of these IED attacks varies considerably. Some are very successful, producing casualties and having strategic effects. Others fail to detonate or are located and neutralized prior to detonation (by forces other than RCTs), resulting in little or no success. If no potential IED target arrives at an emplaced IED, the IED eventually expires.

The mission execution function has four outputs: the set of RCT-detected IEDs along with their locations and compositions, the set of employed IEDs (including those found by forces other than RCTs) with reports on their effects and known or probable compositions, the number and rate of convoys and military movements using each route, and updates to terrain aspects for each route used for military movements. Expired IEDs are not reported or outputted by this function.

3.2.3 Assess

The *assess* targeting activity takes the information coming from the *decide*, *detect*, and *deliver* targeting activities and carries out the *predict* fundamental of assured mobility using the processes of mission analysis. Specifically, the functional inputs are: the reports of all IED activities (found and detonated), the amounts of operational (military) traffic along routes, feedback on the important terrain aspects of the routes of the network, and the current mission analysis. The *assess* function employs statistical methods, such as pattern analysis, to make determinations about future IED attack distributions and their associated risks along the routes of the road network. The *assess* targeting activity outputs the updated enemy most likely and most dangerous courses of action, consisting of IED likelihoods and risks, and amendments to the terrain analysis. These outputs are updates to the mission analysis.

3.3 The Targeting Process in Different Echelons

Following from figure 2-8, figure 3-4 depicts the relationship between the Army command and staff processes (i.e., mission planning and execution) and the Army targeting process as it occurs simultaneously at different echelons. These processes are continuous, with multiple activities occurring simultaneously throughout operations, but they typically begin in an echelon (e.g., a division) with the reception of an operations order from its higher headquarters. "Operations orders," mentioned previously, are official communications of intelligence, information, missions, objectives, purposes, planned operations, specified tasks, coordinating instructions, instructions for logistical support, and/or administrative assignments from an echelon's command to its subordinate elements. These orders are usually contained in official documents, but can also be issued verbally. Once a unit receives an operations order, the unit's staff performs the steps of the MDMP, analyzing the order, performing IPB, etc. Simultaneously, the *decide* targeting activity is carried out. The decisions made as a result of the MDMP and *decide* targeting activity are issued to the division's subordinate elements (i.e., brigade combat teams, special battalions and companies) in an operations order.

This operations order accomplishes two things of importance to this discussion of the targeting cycle. First, it directs specific subordinate elements to perform tactical tasks in order to accomplish the *detect* and *deliver* activities within the order-issuing unit's targeting process (these are depicted by the arrows pointing down). Directed subordinates carry out their assigned tasks and report these activities and their results back to the division headquarters, which then *assesses* these outcomes and their effects, updates the mission analysis, and continues the targeting process and MDMP with this new information as previously discussed. Second, it provides the critical input into each subordinate element's mission analysis and targeting process, depicted by arrows that cross echelon boundaries from left to right.

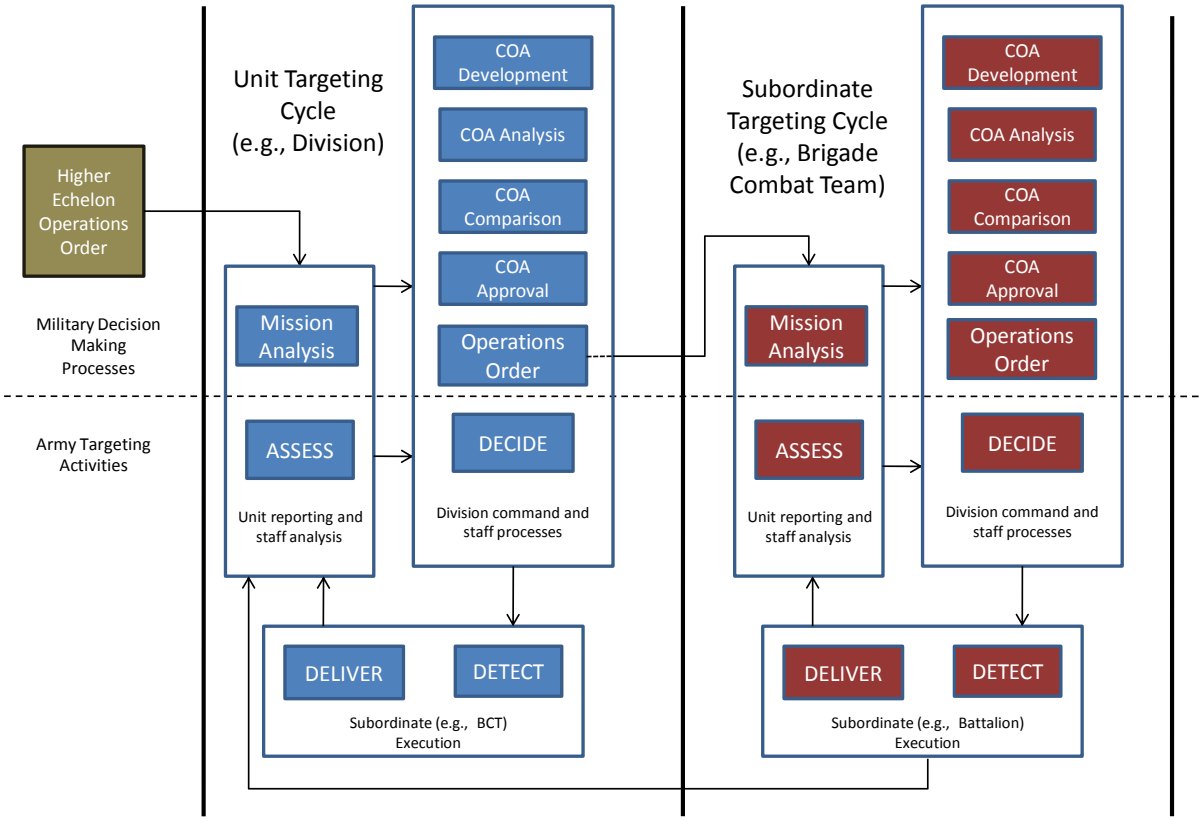

Figure 3-4: Multi-Echelon Targeting Process Interaction

As an example, suppose a brigade combat team is conducting support and stability operations and receives an order from its division headquarters to capture a division target that has been detected at a specific location. The brigade gives this assignment to one of its battalions, which further plans and executes this mission (using the targeting process and MDMP). These brigade and battalion actions comprise the *deliver* activity within the division targeting cycle. However, suppose also that the division operations order contains additional information that assists the brigade in locating another target that is not a division target. The brigade's staff analyzes this new information, and updates its target list and priorities, and assigns another subordinate battalion to conduct surveillance on this new target. These actions constitute *decide* and *detect* activities in the brigade targeting cycle. The important point to note is that subordinate units, in addition to carrying out their own targeting processes, are often tasked to execute the *detect* and *deliver* activities in their higher echelons' targeting cycles. As we begin the mathematical

modeling of RCT targeting in the next chapter, we assume we are working within the planning processes of a single echelon.

[This Page Intentionally Left Blank]

4 Solving the Problem

This chapter explains how we can model IED and counter-IED activities, and how we can apply approximate dynamic programming and mathematical optimization techniques to our model to assist in carrying out the functions of the *decide* targeting activity.

4.1 The Model and Inputs

In order to model the complex interactions that make up IED warfare mathematically, we create a model that captures key characteristics in spite of its simplifications and assumptions. The model we present separates the operational environment into three components: the insurgent (or IED) forces, the terrain, and the counter-insurgent (or counter-IED) forces. We model two types of interaction among these components: IED activities, which involve insurgents interacting with the terrain, and counter-IED activities, which consist of counter-insurgent forces' interaction with the terrain. Because our focus is to defeat the device, we do not model direct interaction between opposing forces, although we account for the effects of such activities on IED and counter-IED activities. The purpose of this section is to describe the data model we use to mathematically represent these components and their interactions, discuss the information (or inputs) required to construct these parameters, and validate the implicit assumptions in the data model parameters. This section ends with a summary of all input parameters.

4.1.1 The Insurgent Component

The insurgent forces emplace and employ IEDs. We allow insurgents in this model to make use of different types of IEDs and IED attacks. Each "IED type" in the model has an associated effectiveness parameter, which is the probability it produces catastrophic effects conditioned on its interaction with a military convoy. The definition of "catastrophic effects" is left up to the user, and can include human casualties or disabled combat vehicles. "Interaction" with a military convoy means that a convoy (not a route clearance team) and the emplaced IED are present on the same road segment at the same time, resulting in IED detonation, malfunction, or discovery and disposal by the passing convoy. We assume conditional IED effectiveness probabilities can be determined statistically from reports on known convoy-IED interactions, which are well documented.

The number of IED types considered and their defining characteristics are left open to the requirements of the user. Modeled IED types can include incidents involving more than one explosive device. As an example, a common insurgent IED tactic might be to employ two IEDs of a certain composition in a single, coordinated attack. In this case we would define an IED-type "*k*" in our model that corresponds to this particular coordinated double-IED attack. An occurrence of an attack of this kind would be considered single IED incident of type *k* in the model.

4.1.2 The Terrain Component

We consider two components of terrain in which events occur: location in the road network and time. Our model of IED warfare evolves in time over road segments that comprise a road network.

4.1.2.1 The Road Network

To represent location, we model the road network as a graph comprised of a set of nodes (*N*) and a set of directed arcs (*A*). Each arc joins two nodes in the graph, representing a specific road segment and direction of travel. For convention, we assign indices *i* and *j* to nodes in the graph and *a* to the arcs, so that arc $a = (i, j) \in A$ indicates the existence of an arc (*a*) going from node *i* to node *j*, where *i* and *j* are each members of set *N*. By using directed arcs we limit each arc to modeling one-way traffic, requiring a road with two-way traffic to be modeled as a pair of separate, *anti-parallel* arcs, e.g. (*i*,*j*) and (*j*,*i*). Figure 4-1 is a small example of a directed graph.

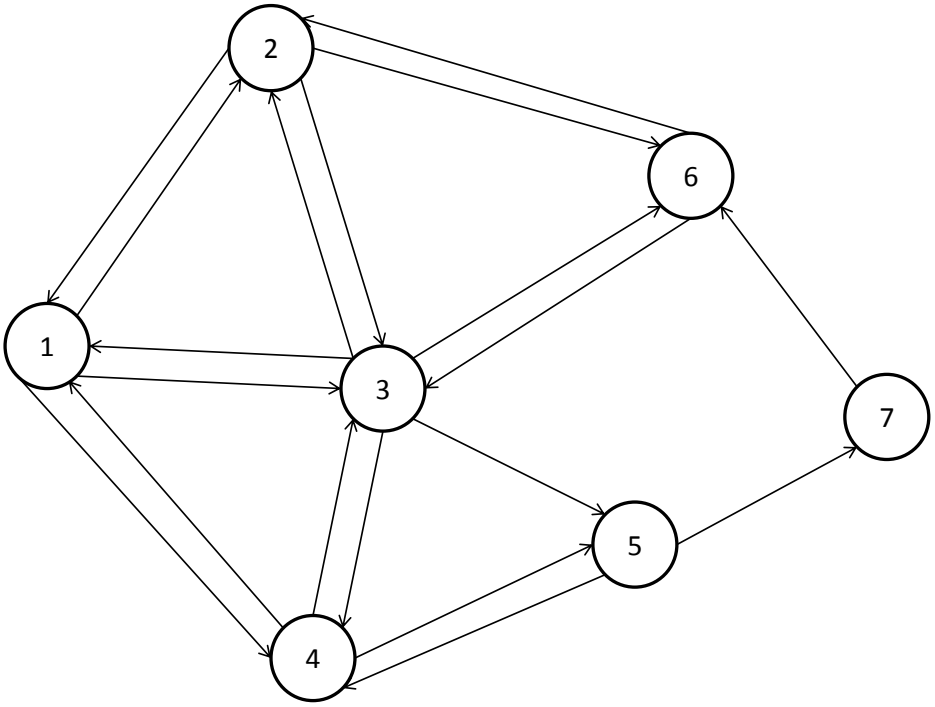

Figure 4-1: A Directed Graph

We use the graph model to characterize the physical shape of the road network. The set of arcs (*A*) provides us with a collection of road segments that comprise the area of interest. We account for the varying physical aspects of terrain by assigning parameters to each arc that describe the associated road segment. The two parameters we use to describe each arc are the modeled road segment length, which is available from map and satellite data, and roadway classification, a categorization that accounts for road features such as width, surface type, construction, state of repair, "clear areas", traffic conditions, etc. The inclusion of this general "classification" variable allows users to sort roads into as many categories as desired, based on any features determined to have an impact on IED and counter-IED activities.

While it is possible to model these terrain aspects as time-dependent parameters, we assume these aspects to be static in our current formulation. This "time-invariant terrain" assumption is valid for features such as those mentioned above, which are not likely to change or, if they did change (e.g., when a road is re-surfaced), the affected arcs could be re-classified manually. However, by making this assumption we do not allow for the time-dependent effects of weather

and light. To include for these transient effects, one could create more arc classifications and allow this parameter to take different values at different times for the same arc.

4.1.2.2 Time

We model time as a set of discrete time increments, or stages, that span our planning horizon, including enough additional time beyond the planning horizon to allow for completion of any route clearance missions that might be in progress at the end of the planning horizon. Discrete time increments are assumed to be reasonably short, so that the likelihood of multiple events (such as IED attacks) at a single location during a single stage is negligible. We do not assign any parameters to model unique characteristics associated with each stage, as we did with the arcs representing different locations.

4.1.3 The Counter-Insurgent Component

4.1.3.1 Route Clearance Teams

We introduce RCTs in the model as the primary IED-detection assets. Each available RCT is assigned a unique index integer *h*, to which its properties are assigned. As with the arcs in the terrain model, RCTs are classified into groups with similar characteristics, such as capabilities and composition. The number of different types of RCTs available and user discretion dictate the number of different classifications within the model.

We also consider constraints on availability of each RCT. This information comprises several items. The first set consists of minimum and maximum planned mission times for each RCT. This information comes from unit operating procedures and evaluation of the capabilities of the RCTs. For example, a leader might decide that route clearance missions are planned to last five hours, assessing that beyond that time interval the effectiveness of the team diminishes. The staff would then plan missions that they estimate will take between 4.5 and 5.5 hours to complete in order to comply with this guidance. These numbers constitute minimum and maximum planned mission times.

In addition to limits on the duration of planned missions, one must also consider minimum idle time, or *dwell*, between route clearance missions and the maximum number of missions a RCT can conduct in a day. Logistical and personnel constraints, for example, might dictate that a

RCT is capable of running no more than two missions each day (each planned within specified duration limits), with at least 90 minutes of dwell time between missions for refit, resupply, and planning updates. These are realistic constraints that must be considered in a model.

Lastly, we must know when each RCT is available for use. If there are no external constraints on when an RCT can conduct missions within the planning horizon, then there would be no need for this additional information. Often, however, such constraints do exist and arise from operational or logistical considerations, such as command guidance restricting operations to daylight hours only (an operational constraint), or scheduled maintenance activities (a logistical constraint).

4.1.3.2 Other units

We also consider units and patrols other than RCTs in the model, but do not assign any characteristics or parameters to them. We assume that these units carry out military operations and sustain the military force logistically, but we do not distinguish them based on type, size, or composition.

4.1.4 Insurgent—Terrain Interaction

We assume that insurgent forces employing IEDs do so in statistically observable patterns or in response to environmental variables (such as ongoing military operations, civil operations, cultural or religious observances, social demographics, etc.), giving us the capability to predict the likelihood of IED activities. In other words, we have the ability to determine with some level of accuracy the rate at which insurgents place each type of IED along each road segment for each stage in the model. This assumption is not unrealistic. Insurgents operate under a set of constraints that limit the range of courses of action available to them. It is also common for insurgents (or any military force) to repeat tactics that have proven successful. Thus, patterns of IED employment are likely to emerge as a function of time, place, and current events.

Estimating the probable rates of emplacement from empirical data is beyond the scope of this thesis. In the current insurgent/counter-insurgent conflicts in Iraq and Afghanistan, data on known IED incidents, political events, military operations, and convoy movements are collected and saved (and typically classified at a level that does not allow open access). Ongoing

statistical analysis of these large and ever-increasing data collections can be used to determine IED emplacement rates.

We model IED-emplacements for each type of IED on each arc as a Poisson process, using the statistically estimated emplacement rates as distribution parameters. From the resulting distributions, we can estimate the probabilities of insurgents physically emplacing of each type of IED on every arc in the graph for each stage in the model.

We use these IED emplacement probabilities, along with counter-insurgent—terrain parameters, to model how the probability that an IED of each type is in place on each arc evolves over time. Because the probability of an IED existing on an arc during a stage depends on whether an IED existed on the arc during the previous stage, our model also requires initial conditions that provide the probability of each type of IED existing on each arc at the beginning of the planning horizon. These probabilities can be obtained from previous planning iterations if available, can be estimated, or can be chosen within reasonable limits, assuming that the dynamic nature of the problem will quickly marginalize the effects of the initial conditions.

4.1.5 Counter-Insurgent—Terrain Interaction

4.1.5.1 Route Clearance Missions

Route Clearance Teams move along roadways, detecting and neutralizing IEDs. In order to model this function, we assign speed and clearance effectiveness parameters to each RCT. RCT clearance speed is a function of the team's configuration and the road type, and can be determined from analysis of the routing and duration of past missions, or can be estimated by individuals familiar with RCT capabilities. Knowing the RCT's clearance speed for each road type enables us to compute the amount of time it takes the RCT to clear each arc graph representing the road network. RCT effectiveness, or probability that an RCT finds an emplaced IED, depends on the IED type, the road type, and the RCT configuration. RCT effectiveness would typically be estimated and analyzed during the mission analysis and COA development stops of the MDMP. They would be difficult to determine from empirical data because of the imprecision involved in assessing whether an IED was missed by an RCT on a mission, and because of the infeasibility of conducting reliable controlled tests. We assume in our model that RCTs are successful (i.e., better than 70% detection rates) at finding the IEDs they are

configured to detect. Without an assumption of at least moderate success, it would not make sense to plan and conduct route clearance missions.

The final piece of information required to model a RCT's interaction with terrain is its home location, or base. This is the place from which it begins and ends its missions, which we represent as a node in the road network graph. We assume that RCTs begin and end missions at the same location. It is unlikely for a route clearance mission to end at a location different from its starting point because it would entail the inefficiency of having to provide RCT equipment and personnel support facilities at both locations.

4.1.5.2 Other Military Operational Movements

Aside from the parameters pertaining directly to Route Clearance Teams, we assume to know something about the other military movements' (e.g., supply convoys and combat patrols) densities through the road network. This data can be gathered and compiled statistically, much as the insurgents' use of IEDs, from archived data on historical convoys, patrols, and other military movements. An arguably good source of data pertaining to ongoing operations in Iraq and Afghanistan is the archived Blue Force Tracker (BFT) database. This database consists of BFT-equipped military vehicle locations recorded at regular intervals using satellite communication. This information is not perfect; not all vehicles are equipped with BFT, and not all recorded locations are accurate. Nonetheless, the database provides reasonable estimates of friendly (i.e. counter-insurgency forces') traffic patterns along routes.

From this data, we can approximate future rates of use for each road segment for different periods throughout the day. Like IED emplacements, we assume that friendly use of a road segment is approximated by a time-dependent Poisson process, for which the use rates determined from available data supply the distribution parameters.

We make an assumption that a non-route clearance military movement always interacts with an existing IED on an arc it traverses, resulting in the IED's detonation or discovery. This assumption neglects the chance that a convoy passes an IED without noticing it, while the IED remains in place to detonate on a future convoy. We justify this assumption by examining the risk an insurgent assumes by leaving an IED emplaced longer than necessary to achieve effects. Regardless of the amount of traffic on a route, increasing the amount of time an IED is in place

and idle increases the likelihood that it will be discovered, or that it will malfunction. It is possible certain IEDs might be designed to attack specific types of convoys or movements, but even in these cases we can expect the IEDs to be emplaced so as to minimize the idle time before the targeted convoy arrives and the IED is employed. It is possible to imagine instances in which some possible reward exists by delaying an IED initiation, or some circumstance temporary inhibits an IED's initiation, but we disregard these scenarios for now as exceptional cases. If we wanted to remove this assumption, we could introduce probabilities for IED interaction given a convoy arrival, and even define different convoy configurations with different probabilities of triggering, finding, or bypassing each type of IED. This change might make the model more realistic, but determining the probability of a convoy passing and IED without knowing it would be difficult because of a lack of available information on the frequency of such occurrences. Alternatively, we could consider these probabilities of interaction as implicit in the arc use probabilities.

4.1.5.3 Summary of Input Parameters

Table 4-1 names, defines, and summarizes each input parameter used in the model as discussed throughout this section.

Input Parameter	Description
Insurgent Modeling Parameters	
K	The number of different IED types employed. By convention, $k \in \{1..K\}$.
P_eff_k	The probability that an IED of type k is effective, given it is "triggered."
Terrain Modeling Parameters	
N	The set of nodes in the graph representing the physical road network. By convention, a node $i \in N$.
A	The set of arcs in the graph representing the physical road network. By convention, an arc $a \in A$.
D	The number of days in the planning horizon under consideration. By convention, $d \in \{1..D\}$.
δ	The number of minutes in a discrete time step, or stage.
T	The number of stages in the model, i.e. $T = \frac{D \times 24 \times 60}{\delta} + \max_h (T\_max_h)$. By convention, $t \in \{0..T\}$.
$Road\_type_a$	The road type classification of the road segment represented by arc a.
$Length_a$	The length of the road segment represented by arc a.
Counter-Insurgent Modeling Parameters	
$\bar{h}$	The number of route clearance teams (RCTs). By convention, $h \in \{1..\bar{h}\}$.
RCT_type_h	The configuration type of RCT h.
T_min_h	The minimum mission duration for planning purposes, in stages, for RCT h.
T_max_h	The maximum mission duration for planning purposes, in stages, for RCT h.
$dwell_h$	The minimum idle time between missions for RCT h in stages.
Max_day_h	The maximum number of missions RCT h can conduct in one day.
$Avail\_start\_0_{h,d}$	The beginning of RCT h available mission start time on day d, such that $Avail\_start\_0_{h,d} \in \{(d-1) \times \frac{24 \times 60}{\delta}, .., d \times \frac{24 \times 60}{\delta}\}$.
$Avail\_start\_1_{h,d}$	The end of RCT h available mission start time on day d, such that $Avail\_start\_1_{h,d} > Avail\_start\_0_{h,d}$, and $Avail\_start\_1_{h,d} \in \{(d-1) \times \frac{24 \times 60}{\delta}, .., d \times \frac{24 \times 60}{\delta}\}$.
Insurgent—Terrain Interaction Modeling Parameters	
$P\_0_{a,k}$	The initial condition probability that a type-k IED is in place on arc a at the beginning of the planning horizon (stage 0).
$\lambda_{a,t,k}$	The rate at which insurgents physically emplace IEDs of type k on arc a during stage t. These rates can be determined using statistical analysis and/or qualitative inputs.
Counter-Insurgent—Terrain Interaction Modeling Parameters	
$base_h$	The operating base for RCT h.
$P\_detect_{q,k,r}$	The probability of a RCT configuration type q detecting and IED type k on road type r.
$V_{q,r}$	The speed at which a RCT configuration type q clears a road of type r.
$\mu_{a,t}$	The rate at which counter-insurgency forces (or targeted forces) traversing the road segment modeled by arc a at stage t.

Table 4-1: Algorithm Inputs

4.2 The Route Clearance Planning Algorithm

4.2.1 Overview of Algorithm

To determine a feasible, coordinated route clearance plan that seeks to minimize the risk associated with effective IED attacks, we run our input parameters through a series of computation and optimization functions. Each of these functions is described in detail in the following paragraphs. The functions are depicted in Figure 4-2, which illustrates the iterative and sequential process in which they are carried out.

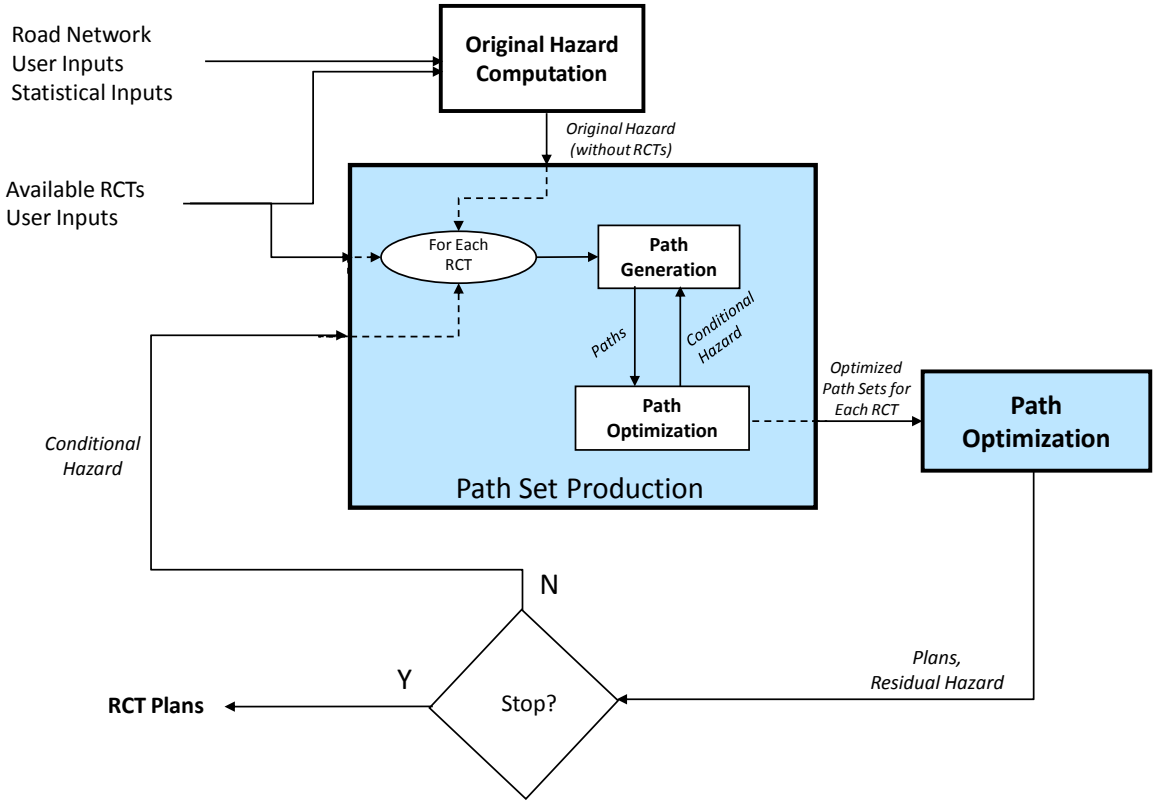

Figure 4-2: Route Clearance Planning Algorithm Overview

4.2.2 The Objective Function

Our objective in this algorithm is to minimize the probability of an effective IED attack. We refer to an IED attack as "effective" if it produces catastrophic effects, e.g., causes human

casualties or destroys combat systems. We assume in the formation of our objective that IED events of different types, at different times, or at different locations are independent. Let $\theta_{a,t,k}$ be the probability of at least one effective type-*k* IED attack occurring on arc *a*, at stage *t*. Using independence among arcs, times, and IED types, we can arrive at an expression for the probability of at least one effective IED attack occurring over the terrain in the model:

$$1 - \prod_{a \in A} \prod_{t \in \{1..t\}} \prod_{k \in \{1..K\}} (1 - \theta_{a,t,k}) \qquad \text{(OBJ1)}$$

Our objective is to minimize this expression, or equivalently to perform the maximization:

$$maximize \ \prod_{a \in A} \prod_{t \in \{1..t\}} \prod_{k \in \{1..K\}} (1 - \theta_{a,t,k}). \qquad \text{(OBJ2)}$$

Because the logarithm function is monotonically increasing, applying the logarithm transforms the expression without changing the solution:

$$maximize \ \log(\prod_{a \in A} \prod_{t \in \{1..t\}} \prod_{k \in \{1..K\}} (1 - \theta_{a,t,k}))$$
$$= maximize \ \sum_{a \in A} \sum_{t \in \{1..t\}} \sum_{k \in \{1..K\}} \log(1 - \theta_{a,t,k}) \qquad \text{(OBJ3)}$$

Finally, we transform this expression again to a minimization problem:

$$minimize \ \sum_{a \in A} \sum_{t \in \{1..t\}} \sum_{k \in \{1..K\}} -\log(1 - \theta_{a,t,k}) \qquad \text{(OBJ4)}$$

All of the objective functions above are equivalent in solution, even though they return different optimal objective values. The final objective function (OBJ4) has several advantages. By taking the logarithm, we make our problem additive instead of multiplicative. Making use of this property, we can define a function to quantify IED exposure risk associated with subsets of IED types, arcs, or stages in our model. Specifically, given a set $\tilde{S} = \{\tilde{A}, \tilde{T}, \tilde{K}\}$, for which $\tilde{A} \subset A$, $\tilde{T} \subset \{1..T\}$, and $\tilde{K} \subset \{1..K\}$, we define the *Hazard* function:

$$\Xi(\tilde{S}) = \sum_{a \in \tilde{A}} \sum_{t \in \tilde{T}} \sum_{k \in \{\tilde{K}\}} -log(1 - \theta_{a,t,k}).$$

This function has the following desirable properties:

- If $Pr\{\text{At least 1 effective IED attack in } \tilde{S}\} = 0$, $\Xi(\tilde{S}) = 0$
- For $Pr\{\text{At least 1 effective IED attack in } \tilde{S}\} = 1$, $\Xi(\tilde{S}) = \infty$.
- $\Xi(\tilde{S})$ monotonically increases with $Pr\{\text{At least 1 effective IED attack in } \tilde{S}\}$.
- For two disjoint subsets $\tilde{S}_1$ and $\tilde{S}_2$, $\Xi(\tilde{S}_1 \cup \tilde{S}_2) = \Xi(\tilde{S}_1) + \Xi(\tilde{S}_1)$.

These properties make the Hazard function useful for computation of total risk, given risks associated with all the arc-times in a network, and for risk-based comparisons of differing courses of actions.

Using the Hazard function, we can rewrite our objective:

$$\begin{aligned} &\text{minimize } \Xi(A, \{1..T\}, \{1..K\}) \\ &= \text{minimize } \sum_{a \in A} \sum_{t \in \{1..t\}} \sum_{k \in \{1..K\}} \eta_{a,t,k} \qquad \text{(OBJ5)} \\ &\text{where } \eta_{a,t,k} := \Xi(a,t,k) = -\log(1 - \theta_{a,t,k}) \end{aligned}$$

We refer to $\eta_{a,t,k}$ as the *hazard*, or IED exposure risk, of IED type *k* on arc *a* at stage *t*. These hazard values serve as the basis for comparison throughout the functions and optimization methods in this model.

4.2.3 Original Hazard Computation

The Original Hazard Computation function computes hazard values, as defined above, for all IED types, on all arcs, at all stages in the model, *assuming that no route clearance missions are conducted*. The function inputs the sets of nodes (*N*), arcs (*A*), and stages ($\{0..T\}$) that model the terrain, the IED emplacement rates ($\lambda_{a,t,k}$), IED effectiveness probabilities (P_eff_k), initial IED conditions on the arcs ($P\_0_{a,k}$), and the counter-insurgent force arc use rates ($\mu_{a,t}$). This function models IED emplacement and friendly arc usage as independent Poisson processes. It employs a two-state Markov chain to model the evolution probability that an IED of type *k* ($k \in K$) is in place on a road segment *a* ($a \in A$), conditioned on the absence of route clearance missions, for all arcs and IED types in the model. Figure 4-3 illustrates this two-state Markov process with transition probabilities. The first state (on the left in Figure 4-3) denotes the absence of an IED of type *k* on arc *a*, while the second state indicates the presence of an IED of type *k* on arc *a*. In this model, one of four events occurs at each stage. Two of these events cause state transitions:

- Given arc *a* does not have an IED of type *k* in place at stage *t*, insurgent forces emplace an IED of type *k* on arc *a* at stage *t*. We approximate the occurrences of this event as a Poisson process with parameter $\lambda_{a,t,k}$. The probability of this event occurring is

  $$P\_em_{a,t,k} := 1 - e^{-\lambda_{a,t,k}\delta}.$$

  This event causes a transition to the "Type *k* IED in place" state.

- Given a type *k* IED is present on arc *a* at stage *t*, a military convoy (other than a RCT) traverses the arc and interacts with the IED. We approximate the occurrences of this event as a Poisson process with parameter $\mu_{a,t}$. The probability of this event occurring is

  $$P\_use_{a,t} := 1 - e^{-\mu_{a,t}\delta}.$$

  This event causes a transition to the "No type *k* IED in place" state.

The remaining two events are complements to the events described above, and result in the Markov chain remaining in the same state:

- Given arc *a* does not have an IED of type *k* in place at stage *t*, insurgent forces *do not* emplace an IED of type *k* on arc *a* at stage *t* with probability

  $$1 - P\_em_{a,t,k} = e^{-\lambda_{a,t,k}\delta}.$$

  This event causes the system to remain in the "No type *k* IED in place" state.

- Given a type *k* IED is present on arc *a* at stage *t*, a military convoy (other than a RCT) *does not* traverse the arc and interact with the IED with probability

  $$1 - P\_use_{a,t} = e^{-\mu_{a,t}\delta}.$$

  This event causes the system to remain in the "Type *k* IED in place" state.

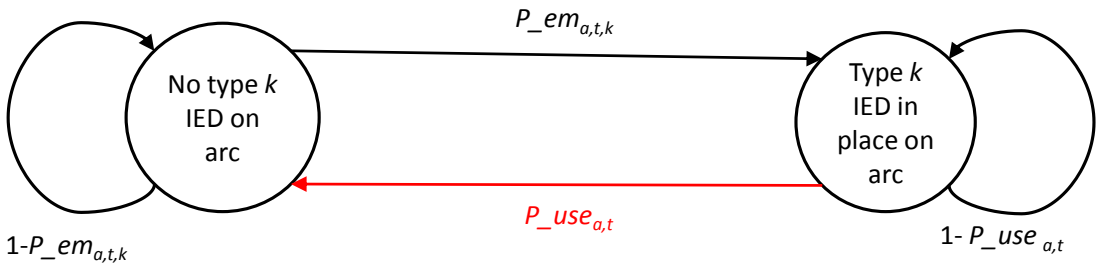

Figure 4-3: Two-state Markov Model for Arc a, IED type k, Stage t

Each of these events is modeled as occurring instantaneously in one discrete time increment, or stage. This is an approximation, as convoy traversals and IED attacks can sometimes occur over a period of time that spans several stages. Also, the discrete time increments are assumed to be short enough to allow us to neglect the probability of multiple events occurring within the same stage.

In addition, using this two-state Markov chain does not allow for representation of more than one IED of a single type simultaneously being in place on the same arc. Adding states to the Markov chain to allow for multiple IEDs is possible, but would make hazard computations more difficult without guaranteeing a more accurate description of IED activity. Building a graph using short arcs to model areas with high IED attack probabilities strengthens the validity of this approximation by marginalizing the probability of multiple IEDs simultaneously existing on the same arc.

A final assumption implicit in this model is that IED attacks are accurately characterized by the Markov property: future state transitions (IED emplacements or uses) are conditionally independent of past state information, given knowledge of the present state of an arc. This assumption also comes from modeling IED emplacements and military convoy arc usage as Poisson processes. We make this assumption because it seems reasonable to assume that a past IED incident or convoy occurring on an arc would not significantly affect the distribution of future IEDs or convoys.

The limitations and approximations imposed on our model by our use of the two-state Markov process to determine evolution of IED attack probabilities do not undermine its use in mission planning. Because at the time of planning we assume no knowledge about future IED attacks beyond their probabilities, increasing the complexity of our model by adding more states, transitions, or probabilities would result in increasing our computational effort with no guarantee of producing a better representation of the dynamics of IED emplacements and interactions.

Using the initial conditions for each arc and the defined Markov process, we can find a recursion to compute the probabilities of each type of IED present on each arc for all stages, conditioned on no route clearance. We define this quantity:

$$P_{a,t,k} := \text{Prob}\{\text{IED type } k \text{ in place on arc } a \text{ at stage } t \text{ given no route clearance}\}.$$

Using the emplacement and arc use probabilities, along with the arc initial conditions, we can find $P_{a,1,k}$:

$$P_{a,1,k} = P\_em_{a,0,k} \times \left(1 - P\_0_{a,k}\right) + \left(1 - P\_use_{a,0}\right) \times P\_0_{a,k}.$$

Similarly for subsequent stages we have the recursion:

$$P_{a,(t+1),k} = P\_em_{a,t,k} \times \left(1 - P_{a,t,k}\right) + \left(1 - P\_use_{a,t}\right) \times P_{a,t,k}.$$

In each of these calculations we add the probabilities of the two disjoint events that comprise the probability that there is a type *k* IED on arc *a* at stage *t*.

Assuming friendly arc-usage at a given stage is independent of IED existence on the arc, we can compute the probability of an effective type *k* IED attack occurring on each arc at each stage, given no route clearance:

$\Theta_{a,t,k}$:= Prob{Effective IED type *k* attack on arc *a* at stage *t* given no route clearance}

$$= P_{a,t,k} \times P\_use_{a,t} \times P\_eff_k$$

The original hazard values, $H_{a,t,k}$ for all arcs, stages, and IED types, can now be computed. These values comprise the output of the Original Hazard Computation function, and are used as a baseline for comparison throughout the algorithm:

$$H_{a,t,k} = -\ln\left(1 - \Theta_{a,t,k}\right).$$

By substituting, we can create a recursion for original hazard based solely on the inputs (including $P\_em_{a,t,k}$ and $P\_use_{a,t+1}$, which are determined directly from the emplacement and usage rates):

$$H_{a,t+1,k} =$$

$$-\ln\left(1 - \left[P\_em_{a,t,k} \times \left(1 - \frac{1-e^{-H_{a,t,k}}}{P\_use_{a,t} \times P\_eff_k}\right) + \left(1 - P\_use_{a,t}\right) \times \frac{1-e^{-H_{a,t,k}}}{P\_use_{a,t} \times P\_eff_k}\right] \times P\_use_{a,t+1} \times P\_eff_k\right).$$

4.2.4 Path Set Production

The Path Set Production sub-process in the algorithm can be decomposed into two components: a Path Generation function, which creates a mission plan for a given RCT beginning at a given mission start time, and a set of controls that execute the Path Generation function for different inputs, evaluates the outputs, and produces the path set for follow-on optimization in the master algorithm.

4.2.4.1 Path Generation

This function inputs a RCT *h*, along with its characteristics (composition type, detection probabilities, clearance speeds, base node, availability, and mission duration constraints), the original hazard values $H_{a,t,k}$, conditional hazard values $\overline{H}_{a,t,k}$, and a stage t_0 in the model that provides the mission start time. The *conditional* hazard values are hazard values, computed for all arcs, stages, and IED types, conditioned on some set of route clearance activities. For the first iteration, this set of assumed route clearance activities is empty, and the conditional hazard values are the same as the original hazard values. The Path Generation function outputs a path $\{base_h, i_1, i_2, .. i_{l-1}, base_h\}$ for the RCT to follow beginning at stage t_0 that seeks to minimize hazard on the graph, based on the inputs. We refer to this path as a *mission* path. The function also outputs two sets of reduced hazard values: (1) the *reduced original* hazard values $\xi^*_{a,t,k}$, which are hazard values for all arcs, stages, and IED-types that would result if the generated mission were the only route clearance mission carried out during the planning horizon, and (2) the *reduced conditional* hazard values $\bar{\xi}^*_{a,t,k}$, which are the hazard values that would result if the generated mission were carried out in addition to the route clearance activities generated in previous iterations, which are implicit in the conditional hazard values.

The Path Generation problem is to find a path through the graph that the input RCT *h* can feasibly clear, given its location, capabilities, mission constraints, and start time, that results in the most conditional hazard reduction. We assume that the RCT departs its base node at the stage given as the mission start time and continuously traverses arcs until it completes the mission, returning to its base node between the minimum and maximum mission time constraints T_min_h and T_max_h. We do not allow the RCT to sit idle at any node other than its base

location. The time it takes a RCT to clear an arc is determined from its capabilities and the arc's characteristics, and rounded to an integer number of stages.

We develop a method that uses an approximate dynamic programming heuristic to create a mission route, or path, for a given RCT departing its base at a given stage. In order to give some background on the development of the heuristic, and to illustrate the difficulty of solving this problem exactly, we first model the Path Generation problem as a shortest path problem using dynamic programming methods. We also briefly consider how we could model the same problem as a mixed integer program. Finally, we describe two heuristics that find approximate solutions in relatively short time intervals, and briefly discuss the merits of each.

4.2.4.1.1 The Effect of Route Clearance on IED Hazard

Before we begin searching for a good route clearance mission path, we need to define mathematically how to compute the reduction of original hazard and conditional hazard that results from route clearance.

4.2.4.1.1.1 Reduced Original Hazard

To compute all hazard reductions, we apply the two-state Markov model used in the Original Hazard Computation with different transition probabilities, as shown in Figure 4-4. Given the decision to clear an arc with a RCT, the transition probabilities shown in Figure 4-4 are applied immediately *after* the stage the RCT would complete clearing the arc (recall that events are modeled as occurring instantaneously in a single stage). As long as the RCT's detection probabilities are high enough, this action significantly reduces the probability of an IED being on the arc, which in turn reduces the resulting hazard.

For example, suppose a RCT is assigned to clear arc *a*, beginning on stage $t-2$. Based on the arc's characteristics and the RCT's capabilities, it is determined that the arc will take two stages to clear. In this case, we apply the Markov state transition probabilities from Figure 4-4 immediately after stage $t-1$ (after the stage $t-1$ emplacement and use probabilities have already been applied). The original probability of an IED type *k* existing on the arc at stage t ($P_{a,t,k}$) is reduced by the probability of detection, thereby reducing the hazard for this stage:

$$\rho_{a,t,k} = \left(1 - P\_detect_{q,k,r}\right) \times P_{a,t,k} = \left(1 - P\_detect_{q,k,r}\right) \times \frac{1-e^{-H_{a,t,k}}}{P\_use_{a,t} \times P\_eff_k},$$

$$\xi_{a,t,k} = -\ln\left(1 - \rho_{a,t,k} \times P\_use_{a,t} \times P\_eff_k\right)$$
$$= -\ln\left(1 - \left(1 - P\_detect_{q,k,r}\right) \times \left(1 - e^{-H_{a,t,k}}\right)\right).$$

In these expressions, q is the RCT configuration type and r is the classification of arc a. For subsequent stages, assuming no additional clearances of arc a, the probabilities from Figure 4-3 are applied to this reduced probability, presumably causing it to climb, along with the associated hazard, back toward some steady state levels for the given emplacement and use probabilities:

$$\rho_{a,(t+s),k} = P\_em_{a,t+s-1,k} \times \left(1 - \rho_{a,t+s-1,k}\right) + \left(1 - P\_use_{a,t+s-1}\right) \times \rho_{a,t+s-1,k},$$
$$\xi_{a,(t+s),k} = -\ln\left(1 - \rho_{a,(t+s),k} \times P\_use_{a,t+s} \times P\_eff_k\right), \forall s \in \{1 \ldots T - t\}.$$

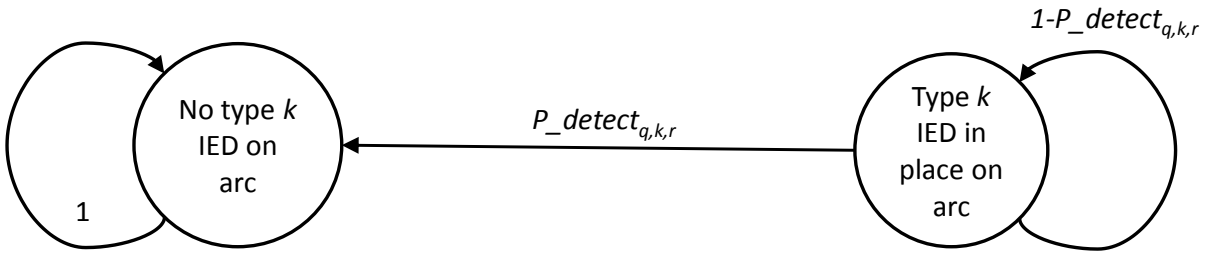

Figure 4-4: State Transition Probabilities for an Arc When it is Cleared

In cases in which an arc is cleared multiple times within the planning horizon, we always compute the immediate reduction in IED probability (and IED hazard) from the original IED probabilities $P_{a,t,k}$ and hazard values $H_{a,t,k}$, without accounting for reductions of previous

clearances. In other words, the calculations in the above example remain correct regardless of whether the RCT cleared the same arc prior to stage *t*. The reason we use this method is to avoid the assumption that successive arc clearances are independent and therefore multiplicative in IED probability reduction. It is more realistic to assume that if a RCT fails to detect an existing IED, then the same RCT is unlikely to find the IED on a subsequent pass because it is probably concealed well. Computing all hazard reductions from the original hazard values is also convenient because it provides a common basis of comparison for route clearance mission effectiveness. It also places a limit on the capability of route clearance to reduce IED risk on each arc. In the Path Generation heuristic, using this method of hazard reduction to limit RCT effectiveness also serves to deter a RCT from clearing the same arc multiple times in succession.

If an arc a' has not been cleared during any stage in $\{1..t\}$, then the reduced original hazard values for arc *a* at stage *t* are the same as the original hazard values, i.e.:

$$\xi_{a',t,k} = H_{a',t,k}.$$

4.2.4.1.1.2 Reduced Conditional Hazard

Reduced conditional hazard is the IED hazard remaining on the graph assuming all missions implicit in the conditional hazard values are carried out in addition to the mission path being generated. We define reduced conditional hazard:

$$\bar{\xi}_{a,t,k} = \min\left[\xi_{a,t,k}, \bar{H}_{a,t,k}\right], \forall a \in A, t \in \{1 \ldots T\}, k \in \{1 \ldots K\}.$$

This definition limits the hazard reduction on arc *a*, stage *t*, for IED type *k* to the best of all the clearances that have occurred on the arc. It ensures that no hazard reduction is achieved when a RCT with low detection probabilities clears an arc immediately behind a RCT with high detection probabilities.

Now that the reduced original hazard and reduced conditional hazard values have been defined, we can consider approaches to solving the Path Generation Problem, which is to find a path that minimizes reduced conditional hazard.

4.2.4.1.2 The Shortest Path Approach to Path Generation

A *shortest path problem* entails finding the minimum cost path from a given start node to a given destination node on a graph (defined by sets of nodes and arcs) in which each arc has some

associated cost. Shortest path problems are deterministic, finite state problems that can be modeled and solved using dynamic programming and related methods, such as Dijkstra's label-setting algorithm [Ber05]. We now use dynamic programming modeling to define our Path Generation problem and translate it to a shortest path problem in order to examine exact solution techniques.

4.2.4.1.2.1 Review of Deterministic Dynamic Programming and Notation

A deterministic dynamic programming model consists of a state space, *S*, a cost function, *g*, a decision space, *U*, and a state transition function, *f*. The Path Generation problem is one of finite horizon because of the time limits on mission duration. The general form of the deterministic dynamic program is:

States	$x_{\tilde{t}} \in S^{\tilde{t}}$,
Decisions	$u_{\tilde{t}} \in U(x_{\tilde{t}})$
Costs (per stage)	$g_{\tilde{t}}(x_{\tilde{t}}, u_{\tilde{t}})$, for decision $u_{\tilde{t}}$ from state $x_{\tilde{t}}$
State transitions	$x_{\tilde{t}+1} = f(x_{\tilde{t}}, u_{\tilde{t}})$
Stages	$\tilde{t} \in \{0 \dots \tau\}$
Terminal cost:	$J_\tau(x_\tau) = g_\tau(x_\tau)$.
Optimal cost to go	$J^*_{\tilde{t}}(x_{\tilde{t}}) = \min_{u_{\tilde{t}} \in U(x_{\tilde{t}})}\left[g_t(x_{\tilde{t}}, u_{\tilde{t}}) + J^*_{\tilde{t}+1}\left(f(x_{\tilde{t}}, u_{\tilde{t}})\right)\right]$

Once these parameters have been defined, the optimal cost to go function is solved iteratively from stage τ backward to stage 0 for each state in each stage's state space, $S^{\tilde{t}}$. The dynamic program returns values for the optimal cost-to-go function $J_0(x_0)$ for all states in S^0, as well as an optimal decision policy, $\pi^* = \{\varphi^*_0(x_0), \dots, \varphi^*_\tau(x_\tau)\}$ for all states in the state space corresponding to each stage, where $\varphi^*_{\tilde{t}}(x_{\tilde{t}})$ is the decision $u_{\tilde{t}}$ that minimizes cost to go from state $x_{\tilde{t}}$, i.e.:

$$\varphi^*_{\tilde{t}}(x_{\tilde{t}}) := \text{argmin}_{u_{\tilde{t}} \in U(x_{\tilde{t}})}\left[g_{\tilde{t}}(x_{\tilde{t}}, u_{\tilde{t}}) + J_{\tilde{t}+1}\left(f(x_{\tilde{t}}, u_{\tilde{t}})\right)\right].$$

4.2.4.1.2.2 Problem Horizon

The horizon in our Path Generation dynamic program is the set of stages $\tilde{t} \in \{0..\tau\}$ that comprise the time period over which the proposed route clearance mission takes place. These stages correspond to a subset of the discrete time increments in the planning horizon of the larger route clearance optimization model. We can write an expression that relates a stage in the Path

Generation dynamic program to a stage in the larger route clearance planning model:

$$\tilde{t} = t - t_0,$$

where $\tilde{t}$ is the stage in Path Generation, t is the stage in the larger model, and t_0 is the mission start time input to the Path Generation function. The dynamic program stage $\tilde{t}$ takes values from 0, corresponding to the mission start time, to T_max_h, the input RCT's maximum mission time. Therefore, for each iteration of the Path Generation function, the horizon is defined by the set $\{0..\tau\}$, in which $\tau = T\_max_h$.

4.2.4.1.2.3 State Space

In defining the state space, we must consider how state transitions and transition costs model the problem. The set of possible RCT locations at each stage, i.e. the nodes and arcs it can reach, seems to be a good selection because the transitions between states are easy to define. However, this state space does not provide enough information to calculate the cost-per-stage function, g_t, which is related to hazard reduction on the arcs. Specifically, at each state we need to know both the RCT location and something about the IED state probabilities (from our two-state Markov model) for each arc we have a choice of clearing. Without these probabilities, we cannot compute the reduced conditional hazard values that would result from clearing an arc or compare those values to the conditional hazard values associated with not clearing the arc.

To understand why this is the case, consider the three-node graph in Figure 4-5, and assume a RCT is positioned at the center node planning for a mission beginning at time zero and consisting of exactly four stages. Also, assume for simplicity that the road segments represented by the arcs in Figure 4-5 are the same length and road type, take one stage to clear, and have equal original and reduced hazard values which are constant for each stage. To further simplify things, let the RCT be 100% effective, so that if an IED is in place on a road it will find it. Finally, we'll assume once an arc is cleared by the RCT, reduced hazard (original and conditional) on that arc will reset to zero for the remainder of the mission.

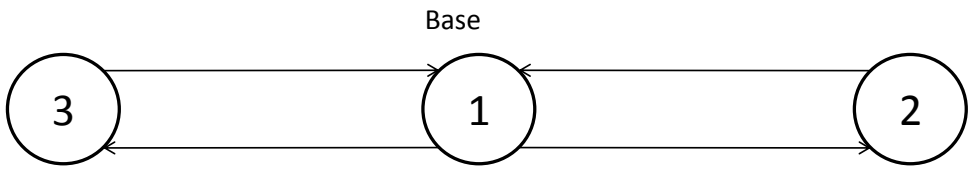

Figure 4-5: Three-Node, Four-Arc Road Network

Now consider the center node (1) in the second stage, the state we would define as $x_2 = 1$ (which is also the only state in S^2). From this state we can either clear the arc leading to node 2, or the arc leading to node 3; these two choices form our set of decisions. The problem is to define $g_2(x_2, u_2)$, the cost (or benefit) of each of these decisions from this position. Knowing the reduced hazard on each arc at the second stage would answer this problem, but we cannot compute these values without knowing which path the RCT took to arrive at its current state. A method of overcoming this problem is through *state augmentation* [Ber05], in which all information needed in the computation of costs and transitions is included in defining each state. For the Path Generation problem, it is enough to include in the definition of each state information giving the RCT's location and a vector of reduced hazard values for all arcs and IED types at the current stage. We now define a (augmented) state in this dynamic program as:

$\boldsymbol{x}_{\tilde{t}} := (i, \boldsymbol{\xi}_{\tilde{t}})$, for states corresponding to the RCT located at a node, or

$\boldsymbol{x}_{\tilde{t}} := (a, \tilde{t}_c, \boldsymbol{\xi}_{\tilde{t}})$, for states corresponding to the RCT in the process of clearing an arc.

In these definitions, i is a node in the graph, $a = (i, j)$ is an arc in the graph, $\tilde{t}_c$ is the number of stages spent clearing arc a in the current pass, and $\boldsymbol{\xi}_{\tilde{t}}$ is a vector of reduced original hazard values for each arc in the graph for each IED-type at the current stage. The reduced original hazard values are determined by applying the methods given in paragraph 4.2.4.1.1 for all arc clearances that preceded the RCT's arrival at state $\boldsymbol{x}_{\tilde{t}}$. The problem with this state space definition is that it allows the state space to increase exponentially with the number of stages. Essentially, we now have a state for each unique, feasible path that exists for each stage in the dynamic program, because each unique path will result in a unique reduced original hazard vector $\boldsymbol{\xi}_{\tilde{t}}$ resulting from clearing a different sequence of arcs.

4.2.4.1.2.4 Decision Space

The decision space in this formulation is simply the choice of which arcs can be cleared from any given state. For simplicity, we define the decision $u_{\tilde{t}}$ at each stage as the node we choose to clear toward. For any state $\boldsymbol{x}_{\tilde{t}} = (i, \boldsymbol{\xi}_{\tilde{t}}) \in S^{\tilde{t}}$,

$$U(x_{\tilde{t}}) := \left\{ j | (i,j) \in A, j \in N^{\tilde{t}+\tilde{c}_{a,h}} \right\}, \text{ for } \tilde{t} \in \{0..T\_min_h - 1\},$$
$$U(x_{\tilde{t}}) := i \cup \left\{ j | (i,j) \in A, j \in N^{\tilde{t}+\tilde{c}_{a,h}} \right\}, \text{for } \tilde{t} \in \{T\_min_h..T\_max_h\}, i = base_h,$$
$$U(x_{\tilde{t}}) := \left\{ j | (i,j) \in A, j \in N^{\tilde{t}+\tilde{c}_{a,h}} \right\}, \text{for } \tilde{t} \in \{T\_min_h..T\_max_h\}, i \neq base_h.$$

We define $N^{\tilde{t}}$ as the set of nodes that make up the $\tilde{t}$th stage of all feasible mission paths (with $N^{\tilde{t}} = \emptyset \; \forall \; \tilde{t} \notin \{0..\tau\}$), and $\tilde{c}_{a,h}$ as the number of stages it takes RCT h to clear arc $a = (i, j)$. Essentially, the set of decisions are the set of nodes that are adjacent to i, from which there exists a clearance path back to the RCT's base within the remaining mission time. The second expression above allows for the RCT to remain at its base if the planned mission path meets the minimum mission time requirement. Finally, for any state $\boldsymbol{x}_{\tilde{t}} := (a, \tilde{t}_c, \boldsymbol{\xi}_{\tilde{t}})$,

$$U(x_{\tilde{t}}) := \{ j | (i,j) = a \}.$$

If the RCT is clearing an arc, it we only allow for it to continue clearing until it reaches the next node.

4.2.4.1.2.5 Costs-per-stage

We define cost-per-stage as

$$g_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}}) := \sum_{a\in A}\sum_{k\in K}\bar{\xi}_{a,\tilde{t}+1,k}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})}.$$

This next stage reduced conditional hazard values, $\bar{\boldsymbol{\xi}}_{\tilde{t}}^{\,u_{\tilde{t}}}$, are computed using the current state information $\boldsymbol{x}_{\tilde{t}}$ and decision $u_{\tilde{t}}$.

$$\bar{\xi}_{a,\tilde{t}+1,k}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})} = \min\left[\xi_{a,\tilde{t}+1,k}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})}, \bar{H}_{a,t+1,k}\right], \forall a \in A, \tilde{t} \in \{1\ldots\tau-1\}, k \in \{1\ldots K\}.$$

In this expression, the conditional hazard values $\bar{H}_{a,t+1,k}$ are inputs in the Path Generation function. The reduced original hazard values are computed as described in paragraph 4.2.4.1.1. If the RCT completes clearing arc a in stage $\tilde{t}$, the reduced original hazard for arc a at stage $\tilde{t}+1$ is:

$$\xi_{a,\tilde{t}+1,k}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})} = -\ln\left(1-\left[\left(1-P\_detect_{q,k,r}\right)\times\left(1-e^{-H_{a,t+1,k}}\right)\right]\right), \forall k \in \{1\ldots K\}.$$

Otherwise, the reduced original hazard at stage $\tilde{t}+1$ is:

$$\xi_{a,\tilde{t}+1,k}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})} =$$
$$-\ln\left(1-\left[P\_em_{a,t,k}\times(1-\rho_{a,t,k})+\left(1-P\_use_{a,t}\right)\times\rho_{a,t,k}\right]\times P\_use_{a,t+1}\times P\_eff_k\right),$$
for $\rho_{a,t,k} = \frac{1-e^{-\xi_{a,\tilde{t},k}}}{P\_use_{a,t}\times P\_eff_k}$, $\forall k \in \{1\ldots K\}$.

The relation $\tilde{t} = t - t_0$ holds in all of the above definitions.

4.2.4.1.2.6 State Transitions

We model this problem as a deterministic dynamic program, i.e., the state transitions are only a function of current state and current decision, with no stochastic disturbance:

$$\boldsymbol{x}_{\tilde{t}+1} = f(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}}) =$$
$$\begin{cases}\left(a, 1, \boldsymbol{\xi}_{\tilde{t}+1}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})}\right), \text{for states } \boldsymbol{x}_{\tilde{t}} = (i, \boldsymbol{\xi}_{\tilde{t}}), \tilde{c}_{a,h} > 1 \\ \left(a, \widetilde{\tilde{t}_c}+1, \boldsymbol{\xi}_{\tilde{t}+1}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})}\right), \text{for states } \boldsymbol{x}_{\tilde{t}} = (a, \tilde{t}_c, \boldsymbol{\xi}_{\tilde{t}}), \tilde{c}_{a,h} > \tilde{t}_c + 1 \\ \left(u_{\tilde{t}}, \boldsymbol{\xi}_{\tilde{t}+1}^{(\boldsymbol{x}_{\tilde{t}},u_{\tilde{t}})}\right), \text{for states } \boldsymbol{x}_{\tilde{t}} = (a, \tilde{t}_c, \boldsymbol{\xi}_{\tilde{t}}), \tilde{c}_{a,h} = \tilde{t}_c + 1, \text{or } \boldsymbol{x}_{\tilde{t}} = (i, \boldsymbol{\xi}_{\tilde{t}}), \tilde{c}_{a,h} = 1\end{cases}$$
in which $a = (i, u_{\tilde{t}})$.

Reduced original hazard values are computed using the the cost-per-stage calculations given in the above paragraph.

4.2.4.1.2.7 Cost-to-go

We can now define the optimal cost-to-go recursion as

$$J_{\tilde{t}}^{*}(\boldsymbol{x}_{\tilde{t}}) = \min_{u_{\tilde{t}} \in U(x_{\tilde{t}})} \left[g_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}}) + J_{\tilde{t}+1}^{*}\left(f(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}})\right) \right], \text{for } \tilde{t} \in \{0, .. \tau - 1\},$$

$$g_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}) = J_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}) = 0 \; \forall \boldsymbol{x}_{\tilde{t}} \in S^{\tilde{t}}, \text{for } \tilde{t} = \tau$$

In the second expression above, we set the terminal cost to zero. This approximation does not account for or compare any reductions in hazard that endure beyond the final stage of the mission. However, are cases in which the reduced hazard values for stages following the completion of a mission could be more costly on some arcs than on others. For example, if a large convoy were scheduled on some arcs following the route clearance mission, with minimal movements on other arcs, any hazard remaining on the arcs with scheduled movements is likely to increase significantly after the mission due to projected increases in use probabilities. By discounting these enduring effects of route clearance, we assume that the emplacement and use probabilities are such that route clearance effectiveness, measured as the difference between conditional hazard values and reduced conditional hazard values, quickly decays over time.

4.2.4.1.2.8 Solution Procedure and Shortest Path Model

We start at the dynamic program at stage τ and compute the values for the cost-to-go function for all states in S^{τ} (in this case it's 0 for all $\boldsymbol{x}_{\tau}$). Cost-to-go is then recursively computed for each state in each stage from $\tau - 1$ to 0. Note that

$$J_0^{*}(\boldsymbol{x}_0) = \min_{\pi} \sum_{\tilde{t} \in \{0..\tau-1\}} g_{\tilde{t}}\left(\boldsymbol{x}_{\tilde{t}}, \varphi_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}})\right) = \min_{\pi} \sum_{\tilde{t} \in \{0..\tau-1\}} \sum_{a \in A} \sum_{k \in K} \bar{\xi}_{a,\tilde{t}+1,k}^{\left(\boldsymbol{x}_{\tilde{t}}, \varphi_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}})\right)}.$$

In this expression, π is a set of decisions $\{\varphi_0(\boldsymbol{x}_0), \varphi_1(\boldsymbol{x}_1), .. \varphi_{\tau-1}(\boldsymbol{x}_{\tau-1})\}$ corresponding to each possible state that ultimately determine the output mission path. Note that this expression is very similar to OBJ5 in Paragraph 4.2.2, summed over the subset of stages in the planning horizon that correspond to the duration of the route clearance mission being planned.

As mentioned before, this dynamic program results in a state space that becomes very large as the number of stages in the dynamic program grows. Also, because the reduced original hazard vector for each state is a function of all previous decisions, solving for this vector for each possible state in stage S^{τ} (i.e., each feasible path) requires reduced hazard computation for all

states in stages $\{0 \ldots \tau - 1\}$. These calculations are necessary to carry out the first cost-to-go recursions, because the recursions require reduced hazard vectors for each state to be defined. Using the dynamic programming algorithm to solve for cost-to-go recursively, beginning from the final stage and progressing backwards, is not an ideal solution procedure when the cost-per-stage function depends on decisions from previous stages.

To make this dynamic program easier to understand and solve, we take advantage of its special structure and model it as a shortest path problem. First, we find that we can reverse the direction of the cost-to-go recursion, solving from stage 0 to state $\tau - 1$. This change removes the minimization from the cost-to-go function because for all stages $\{0, \ldots, \tau - 1\}$, for each state $\boldsymbol{x}_{\tilde{t}+1}$ in $S^{\tilde{t}+1}$, there is only one previous state and decision pair $(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}})$ for which $f(\boldsymbol{x}_{\tilde{t}}, j) = \boldsymbol{x}_{\tilde{t}+1}$. More generally, the states and transitions associated with this dynamic program can be modeled as a *tree*, or a graph with no cycles, indicating that there is only one possible path from the origin $\boldsymbol{x}_0$ to each state $\boldsymbol{x}_{\tilde{t}}$. The second modification we make is to introduce an artificial termination state Z with a cost-free transition from each state in S^{τ}. Figure 4-6 depicts this modified dynamic program for a road network consisting of three nodes and six arcs, shown on the left. For simplicity, we assume that clearance times on each arc are unitary. The state space and transitions for the corresponding Path Generation dynamic program are drawn to the right of the road network depiction, with the path corresponding to each state annotated as superscripts. This figure demonstrates the exponential growth of state space in each stage and shows the structure of the dynamic program, without the artificial termination state, is represented as a tree.

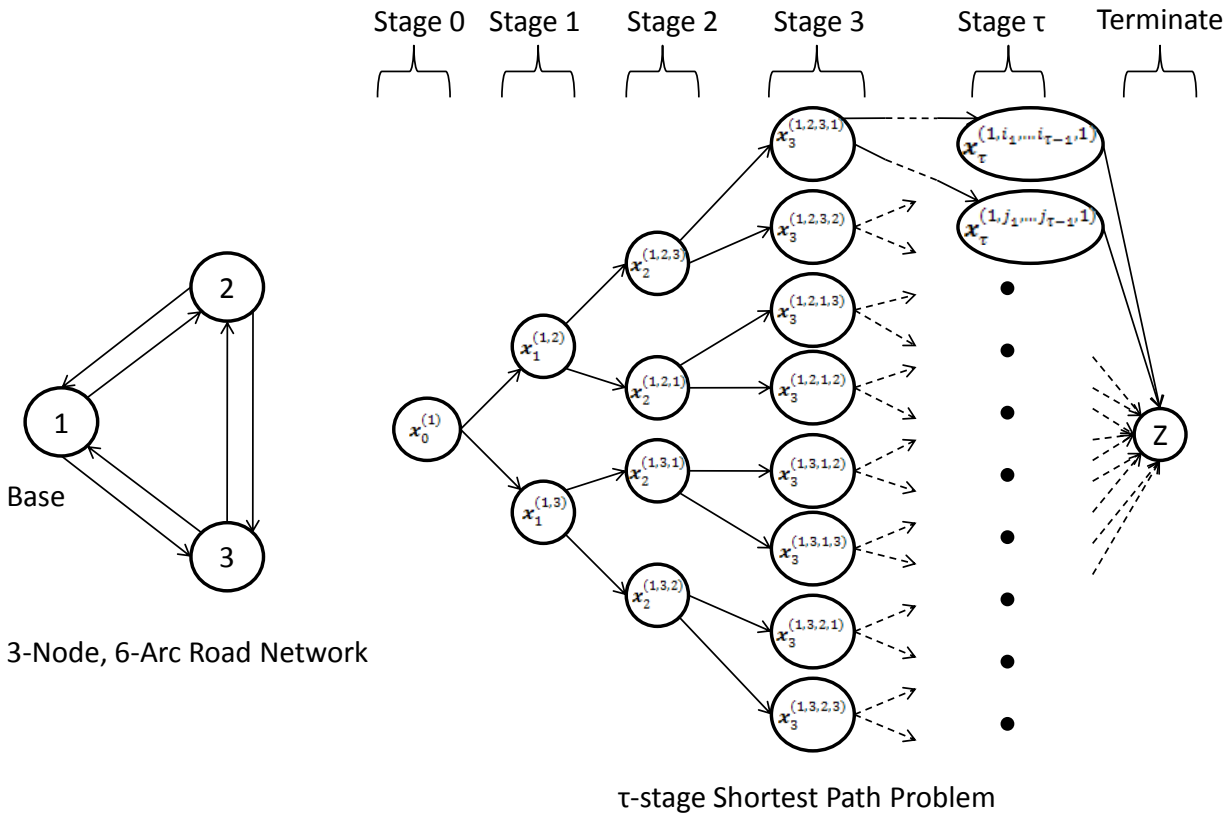

Figure 4-6: Path Generation as a Shortest Path Problem

After making these two modifications, a forward "cost incurred" recursion can be written:

$$g_0(\boldsymbol{x}_0) = \tilde{J}_0(\boldsymbol{x}_0) = 0, \forall \boldsymbol{x}_0 \in S^{\tilde{t}} = \{base_h\},$$
$$\tilde{J}^*_{\tilde{t}+1}(\boldsymbol{x}_{\tilde{t}+1}) = g_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}, u_{\tilde{t}}) + \tilde{J}^*_{\tilde{t}}(\boldsymbol{x}_{\tilde{t}}), \text{ for } \tilde{t} \in \{1, .. \tau - 1\}, u_{\tilde{t}} = \{j \mid f(\boldsymbol{x}_{\tilde{t}}, j) = \boldsymbol{x}_{\tilde{t}+1}\}.$$
$$\tilde{J}^*_z(z) = \min_{x_\tau \in S^\tau} \tilde{J}^*_\tau(\boldsymbol{x}_\tau).$$

This recursion can be thought of as a shortest path label-setting algorithm that proceeds from stage 0 to the artificial termination state, computing cost incurred for each state along the way. The optimal cost incurred values serve as labels for the states in the graph, and upon reaching the termination state point back to the optimal solution path.

Rewriting the problem as a shortest path problem allows us to contract the state space by considering only the states corresponding to node locations. States corresponding to arc clearance activities between nodes, for which only one control exists, need not be modeled in the

shortest path formulation as long as their associated costs are still accumulated in transitions from one consolidated state to another. Because these costs must still be computed and accumulated, this contraction does not save much in terms of computation effort. However, allowing transitions that span several stages to model arc clearance over a period of time can be more intuitive than defining a state for each stage spent clearing an arc.

While solving this problem using a label-setting or label-correcting algorithm gives more intuition than the reverse dynamic program, it does not significantly reduce the amount of computation required to solve it. Applying a good depth-first search heuristic can provide a "cut-off" total cost value, but cost (i.e., reduced conditional hazard) would still have to be calculated for each feasible path until a stage was reached for which the accumulated cost exceeds the cut-off value. Using the small network modeled in Figure 4-5 as an example and setting $\tau = 50$, we can easily see that the number of states in $S^{\tilde{t}}$ is $2^{\tilde{t}}$, for $\tilde{t} \in \{0, .. \tau - 1\}$. Because in a system with relatively constant IED emplacement and arc usage rates it is unlikely that many feasible mission paths accumulated a total cost (i.e., reduced conditional hazard) over 25 stages that exceeds the optimal path's total cost (over all 50 stages), the number of label setting computations for the 25^{th} stage is on the order of 2^{25}. We conclude that this method of solving the Path Generation problem is intractable for missions over more than about 25 stages. See appendix C for a brief analysis and comparison of the performance of this exact dynamic program and several heuristics for Path Generation over a simple road network and a small number of stages.

4.2.4.1.3 Mixed Integer Programming

Another approach to solving the Path Generation problem optimally is to use mixed integer programming. Mixed integer programming optimizes an objective function subject to constraints, including integrality constraints on some variables. In this problem, the objective is to minimize the reduced hazard summed over all IED types on all arcs at all stages in the problem. We do not consider separate conditional and original hazard values in this formulation. As we did in the dynamic programming formulation, we seek the minimum reduced hazard only over the subset of stages that coincides with the particular mission, assuming that a route clearance mission's effects on hazard reduction beyond the mission's duration are reasonably approximated by the reductions during the mission, and that any lingering hazard reductions

decay quickly following mission completion. We make use of the same notation for stages in the Path Generation problem, i.e., $\tilde{t} = t - t_0$. Finally, we apply the same assumptions about hazard reduction due to route clearance (e.g., route clearance on an arc occurs instantaneously after final stage of clearance on the arc).

In mixed integer programming, non-integer constraints must be linear functions of the variables; common solvers such as ©ILOG CPLEX do not solve nonlinear problems with integer constraints. We propose the following mixed-integer formulation for the Path Generation problem:

mimimize $\sum_{a\in A}\sum_{\tilde{t}=0}^{\tau}\sum_{k\in\{1..K\}}\bar{\xi}_{a,\tilde{t},k}$

subject to

$\sum_{\alpha\in O(n)} x_{\alpha,\tilde{t}+\tilde{c}_{\alpha,h}} - \sum_{\beta\in I(n)} x_{\beta,\tilde{t}} = 0, \forall\, \tilde{t} \in \{1..\tau-1\}, \forall\, n \in N\backslash base_h$ (M1a)

$\sum_{\alpha\in O(base_h)} x_{\alpha,\tilde{t}+\tilde{c}_{\alpha,h}} - \sum_{\beta\in I(base_h)} x_{\beta,\tilde{t}} = 0, \forall\, \tilde{t} \in \{1..T\_min_h - 1\}$ (M1b)

$\sum_{\zeta\in\{T\_min_h..\tau\}}\left(\sum_{\alpha\in O(base_h)} x_{\alpha,\zeta+\tilde{c}_{\alpha,h}} - \sum_{\beta\in I(base_h)} x_{\beta,\zeta}\right) = 1$ (M2a)

$\sum_{a\in O(base_h)} x_{a,\tilde{c}_{a,h}} = 1$ (M2b)

$\sum_{a|a\in A\backslash base_h} x_{a,\tilde{c}_{a,h}} = 0$ (M3a)

$\sum_{a|a\in A\backslash base_h} x_{a,\tau} = 0$ (M3b)

$\bar{\xi}_{a,\tilde{t},k} \geq f_{\bar{t}}(\vec{x}_{a,\tilde{t}}), \forall a \in A,\ \forall\tilde{t} \in \{0..\tau\},\ \forall\bar{t} \in \{0..\tilde{t}\}$ (M4)

$x_{a,\tilde{t}} \in \{0,1\}\ \forall a \in A,\ \forall\tilde{t} \in \{0..\tau\}$ (M5)

We define the following notation:

$O(n) :=$ The set of outgoing arcs from node *n*.

$I(n) :=$ The set of incoming arcs to node *n*.

$\bar{\xi}_{a,\tilde{t},k} :=$ The reduced hazard variable for arc *a*, stage $\tilde{t}$, for IED type *k*.

$x_{a,\tilde{t}} :=$ The arc clearance decision variable for arc *a*, stage $\tilde{t}$. $x_{a,\tilde{t}} = 1$ if a clearance is completed on arc *a* at the end of stage *t-1*.

$f_{\bar{t}} :=$ A linear function establishing a lower bound on an arc's reduced hazard value given a route clearance solution vector and search parameter $\bar{t} \in \{0..\tilde{t}\}$.

$\vec{x}_{a,\tilde{t}} :=$ A feasible route clearance solution vector, $\{x_{a,0}x_{a,1}\ldots x_{a,\tau}\}$.

$\tilde{c}_{a,h} :=$ The number of stages required for RCT *h* to clear arc *a*, based on the RCT's capabilities.

The objective function reflects our goal to minimize the reduced conditional hazard over the subset of stages in the planning horizon that pertain to the specific route clearance mission. The first three constraint sets (M1a, M1b, and M1c), (M2a and M2b), and (M3) contain the *network flow* constraints, which, along with the integrality constraints (5), force the $\boldsymbol{x}$ variables to take values that represent a feasible route clearance mission path. Constraint set (M1) indicates that the route clearance flow into a node must equal the route clearance flow out of the node for all nodes and all stages, except the base node for the initial stage and for stages following the minimum mission time. Constraint set (M2) applies network flow constraints to the base node that cause it to release the RCT at the mission start time and to re-absorb it once the minimum mission time has elapsed. Constraints (M3) limit RCT movement to arcs departing the base node for the initial stage, and to arcs arriving at the base node for the final stage. These three constraint groups apply the same mission constraints modeled in the dynamic programming formulation.

Constraint set (4) relates the reduced conditional hazard variables to route clearance variables. While there are several ways to define the functions $f_{\bar{t}}$, the important aspect is that each stage $\tilde{t}$ in the problem requires $\tilde{t}$ additional linear inequality constraints for each arc. If we let m equal the number of arcs in *A*, the number of constraints in constraint set (4) is $m \times \frac{T\_max_h \times (T\_max_h + 1)}{2}$.

The mixed-integer program presented has $m \times T\_max_h$ free variables, on the order of $m \times T\_max_h$ integer variables (although this number might be significantly less than the number of free variables), and $m \times \frac{T\_max_h \times (T\_max_h + 1)}{2} + n \times T\_max_h$ linear constraints, where n denotes the number of nodes in *N*. Given a graph containing 120 arcs and 50 nodes and a maximum mission time of 60 stages, we have on the order of 7200 integer variables, 7200 free variables, and 219600 linear constraints in the mixed-integer program. The reason these numbers are not exact is because some variables might be removed from the problem if they represent arc clearances that are not feasible given the mission constraints. They do, however, give an accurate order of magnitude as long as the RCT can feasibly reach each arc in the road network model. While problems of this size can be solvable, solution software packages use combinatorial techniques, such as branch and cut, which rely on methods similar to the label correcting algorithm described in section 4.2.4.1.2.8. Again, as the number of stages under consideration increases, computation complexity quickly increases and the problem becomes

intractable. Because our route clearance planning algorithm requires many iterations of Path Generation, we seek a much more efficient method of finding solutions to this problem. We now describe two heuristics that were developed using approximate dynamic programming techniques.

4.2.4.1.4 Primary Path Generation Heuristic

We propose a heuristic that uses a label-correcting algorithm to solve an approximation of the shortest path problem previously developed. The primary difference between the heuristic and the exact problems is that in the heuristic, we consolidate the state space to make the problem tractable. We associate a vector of reduced original hazard values with each state, and define a labeling function on this vector. The labeling function enables us to compare a set of paths to each state and select the "best" path for future reduced hazard calculations.

4.2.4.1.4.1 Consolidated State Space

We now re-define the state space as the subset of nodes the RCT can reach during each stage $\tilde{t}$, which we denoted earlier as $N^{\tilde{t}}$. Node i is a member of $N^{\tilde{t}}$ if a $\tilde{t}$-stage feasible clearance path from the input RCT's base to node *i* exists, and a feasible clearance path exists from node *i* back to the base in no more than $\tau - \tilde{t}$ stages (in which $\tau = T\_max_h$, the RCT's maximum mission time). In other words, the RCT can clear a path to the location associated with the node from its base in $\tilde{t}$ stages, and clear a return path back to the base before the remaining mission time has expired.

This definition of state space prevents the number of states in each stage from exceeding the number of nodes in the graph. The initial stage set of states N^0 contains only one state ($base_h$), similar to state space of the exact formulation. However, the state space of the final stage, N^{τ}, also contains only one state (again the node $base_h$), instead of a state representing every feasible τ-stage mission path as we had in the exact problem.

The controls that determine the transitions in this formulation are defined by the set of outgoing arcs from the node that defines the current state. These transitions lead to other nodes in the state space corresponding to future stages. Given a state $\tilde{x}_{\tilde{t}} = i$, we can define the set of controls as

$$\widetilde{U}(\tilde{x}_{\tilde{t}}) = \{j | (\tilde{x}_{\tilde{t}}, j) \in A, j \in N^{\tilde{t}+\tilde{c}_{a,h}}\}.$$

The arc clearance time $\tilde{c}_{a,h}$ in this expression is the number of stages required for RCT *h* to clear arc $a = (\tilde{x}_{\tilde{t}}, j)$, and is computed using the RCT clearance speeds and the arc length and classification. Our state transition function is:

$$f(\tilde{x}_{\tilde{t}}, \tilde{u}_{\tilde{t}}) = \tilde{x}_{\tilde{t}+\tilde{c}_{(\tilde{x}_{\tilde{t}},\tilde{u}_{\tilde{t}}),h}} = \tilde{u}_{\tilde{t}}.$$

We do not model arc clearance states for which there is only one control; arc clearances that require more than one stage result in transitions that span multiple stages. We can depict the state space and control sets we have defined as a graph in which states are represented by nodes and controls and their resulting transitions are represented by arcs. We refer to this graph, which is a *time-space* network, as a "mission network" because it contains all of the feasible paths a RCT can follow subject to its location, capabilities, and mission constraints. Unlike the graph used in the exact formulation, the mission network does not depict each path as a separate state. Figure 4-7 shows how a road network with three nodes and six arcs with unitary expected travel times is expanded into a five-stage mission network, beginning and ending at node 1.

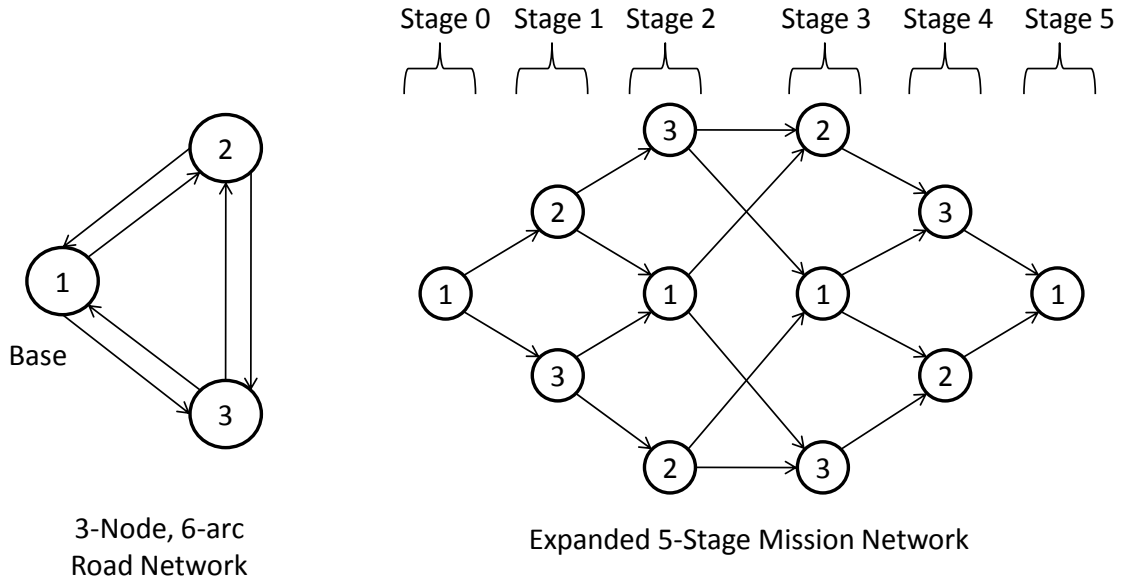

Figure 4-7: Expansion of a Road Network into a Mission Network

4.2.4.1.4.2 Label-Setting Function

From this state space we cannot define a cost function that relies solely on current state and current decision; we must also know something about reduced hazard values at each state. To overcome this problem, we store the vector of reduced original hazard values $\boldsymbol{\xi}_{\tilde{t}}^{\tilde{x}_{\tilde{t}}}, \forall a \in A, \forall k \in \{1..K\}$ for each state. Unlike in the exact problem formulation, this vector does not contribute to defining the state, but is instead a variable parameter assigned to the state. From this reduced original hazard vector we create a state labeling function $L(\tilde{x}_{\tilde{t}})$. Our heuristic method computes and updates these labels for each node in each stage, proceeding from stage 0 to stage τ. When multiple state transitions converge onto a single state $\tilde{x}_{\tilde{t}} = j$, we choose the transition $(\tilde{z}_{\tilde{\zeta}}, \tilde{x}_{\tilde{t}})$ that minimizes the label, and determine the state's reduced original hazard values from this transition's corresponding arc clearance. We set as our labeling function the total reduced conditional hazard at stage $\tilde{t}$:

$$L(\tilde{x}_{\tilde{t}}) = \min_{\{\tilde{z}_{\tilde{\zeta}} | \tilde{x}_{\tilde{t}} \in \tilde{U}(\tilde{y}_{\tilde{\zeta}})\}} [\sum_{a \in A} \min\left(\boldsymbol{\xi}_{a,\tilde{t},k}^{(\tilde{z}_{\tilde{\zeta}}, \tilde{x}_{\tilde{t}})}, H_{a,\tilde{t}+t_0,k}\right)].$$

$$\boldsymbol{\xi}_{\tilde{t}}^{\tilde{x}_{\tilde{t}}} = \boldsymbol{\xi}_{a,\tilde{t},k}^{(\tilde{z}_{\tilde{\zeta}}^*, \tilde{x}_{\tilde{t}})} \text{ for } \tilde{z}_{\tilde{\zeta}}^* = \operatorname{argmin}_{\{\tilde{z}_{\tilde{\zeta}} | \tilde{x}_{\tilde{t}} \in \tilde{U}(\tilde{y}_{\tilde{\zeta}})\}} [\sum_{a \in A} \min\left(\boldsymbol{\xi}_{a,\tilde{t},k}^{(\tilde{z}_{\tilde{\zeta}}, \tilde{x}_{\tilde{t}})}, H_{a,\tilde{t}+t_0,k}\right)]$$

In other words, we use the reduced original hazard values for each state $\tilde{z}_{\tilde{\zeta}}$ from which there exists a transition (i.e., arc clearance) into state $\tilde{x}_{\tilde{t}}$ to compute candidate reduced original hazard vectors $\boldsymbol{\xi}_{\tilde{t}}^{(\tilde{z}_{\tilde{\zeta}}, \tilde{x}_{\tilde{t}})}$ for state $\tilde{x}_{\tilde{t}}$. We select the reduced original hazard values that result in the minimal reduced conditional hazard summed over all arcs and all IED types for stage $\tilde{t}$. These reduced original hazard values correspond to the best path to state $\tilde{x}_{\tilde{t}}$. We also note the state $\tilde{z}_{\tilde{\zeta}}^*$ as being the best state from which to transition to $\tilde{x}_{\tilde{t}}$. In summary, when comparing two or more paths that converge on the same node at the same stage, we select the path that minimizes the reduced conditional hazard at that stage summed over all arcs and all IED types. We keep track of this "best" path and its associated reduced original hazard values as parameters belonging to state $\tilde{x}_{\tilde{t}}$, from which we can compute hazard reductions for future states. The other transitions considered and their corresponding paths are discarded before considering future decisions. This process of "pruning" paths at each stage might result in the rejection of one or more paths that

would yield a lower reduced conditional hazard in future stages than the selected "best" path, but is necessary if we are to set labels on the condensed state space of the mission network.

4.2.4.1.4.3 Heuristic Algorithm

This primary Path Generation Heuristic performs a "greedy" search for good paths by stage; it selects a good path (with minimal reduced conditional hazard) to each node in a current stage, then uses the resulting paths to search for good paths to each node in subsequent stages. The algorithm is greedy in the sense that it sets labels and prunes paths at each stage based on evaluation of immediate hazard reduction, without comparing any enduring effects of route clearance. Following is the algorithm:

0. Let $N^{\tilde{t}}$ be the set of nodes that are on the $\tilde{t}$th stage of all feasible mission paths,
   Set $\boldsymbol{\xi}_{\tilde{t}}^{\tilde{x}_{\tilde{t}}} = \boldsymbol{H}_{\tilde{t}+t_0}, \forall x_{\tilde{t}} \in N^{\tilde{t}}, \forall \tilde{t} \in \{0..\tau\}$
   Set $L(\tilde{x}_{\tilde{t}}) = \sum_{k\in K}\sum_{a\in A} \overline{H}_{a,\tilde{t}+t_0,k}, \forall x_{\tilde{t}} \in N^{\tilde{t}}, \forall \tilde{t} \in \{0..\tau\}$
   Initialize parameters $Path_{i,\tilde{t}}\ \forall i \in N, \tilde{t} \in \{1..\tau\}$.
1. For each stage $\tilde{t}$ in $\{0..\tau\}$,
   a. Set $t = \tilde{t} + t_0$
   b. For each state $\tilde{x}_{\tilde{t}}$ in $N^{\tilde{t}}$,
      i. For each state $\tilde{y}_{\tilde{\zeta}} \in \widetilde{U}(\tilde{x}_{\tilde{t}}), \tilde{\zeta} = \tilde{t} + \tilde{c}_{\left(\tilde{x}_{\tilde{t}},\tilde{y}_{\tilde{\zeta}}\right),h}$,
         (1) Compute $\boldsymbol{\xi}_{\tilde{\zeta}}^{\left(\tilde{x}_{\tilde{t}},\tilde{y}_{\tilde{\zeta}}\right)}$, the reduced original hazard on all arcs in the network for all IED types at stage $\tilde{\zeta}$, given the decision to clear arc $\left(\tilde{x}_{\tilde{t}}, \tilde{y}_{\tilde{\zeta}}\right)$, using the cost-per-stage recursions described in paragraph 4.2.4.1.2.5.
         (2) If $\sum_{k\in K}\sum_{a\in A} \min\left[\xi_{a,\tilde{\zeta},k}^{\left(\tilde{x}_{\tilde{t}},\tilde{y}_{\tilde{\zeta}}\right)}, \overline{H}_{a,t+\tilde{c}_{(i,j),h},k}\right] < L\left(\tilde{y}_{\tilde{\zeta}}\right)$,
            (a) Set $\boldsymbol{\xi}_{\tilde{\zeta}}^{\tilde{y}_{\tilde{\zeta}}} = \boldsymbol{\xi}_{\tilde{\zeta}}^{\left(\tilde{x}_{\tilde{t}},\tilde{y}_{\tilde{\zeta}}\right)}$.
            (b) Re-set the label
            $L\left(\tilde{y}_{\tilde{\zeta}}\right) = \sum_{k\in K}\sum_{a\in A} \min\left[\xi_{a,\tilde{\zeta},k}^{\tilde{y}_{\tilde{\zeta}}}, \overline{H}_{a,t+\tilde{c}_{(i,j),h},k}\right]$
         (3) Set $Path_{\tilde{y}_{\tilde{\zeta}},\tilde{\zeta}} = \tilde{x}_{\tilde{t}}$.
2. Set $\bar{\xi}_{a,t,k}^{*} = \overline{H}_{a,t,k}, \forall a \in A, t \in \{0,..,t_0\}, k \in \{1..K\}$.
   Set $\xi_{a,t,k}^{*} = H_{a,t,k}, \forall a \in A, t \in \{0,..,t_0\}, k \in \{1..K\}$.
3. Set $\tilde{t} = \tau$ and $\tilde{x}_{\tau}^{*} = base_h$. While $\tilde{t} > 0$,
   a. Set $\tilde{t}' = \tilde{t} - \tilde{c}_{\left(Path_{\tilde{x}_{\tilde{t}}^{*},\tilde{t}},\tilde{x}_{\tilde{t}}^{*}\right),h}$.

b. Set $\tilde{x}^*_{\tilde{t}'} = Path_{\tilde{x}^*_{\tilde{t}},\tilde{t}}$.

c. Reset $\tilde{t} = \tilde{t}'$.

4. Set $t = t_0 + 1$. While $t \leq T$:

a. Set $\tilde{t} = t - t_0$

b. If in the solution path $\{\tilde{x}^*_0, \tilde{x}^*_{\tilde{t}_1}, \tilde{x}^*_{\tilde{t}_2} \ldots, \tilde{x}^*_\tau\}$ an arc is cleared at time $\tilde{t} - 1$, i.e. if $\exists\ \tilde{x}^*_{\tilde{t}}$ in the solution path:

i. For $a = \left(Path_{\tilde{x}^*_{\tilde{t}'},\tilde{t}}, \tilde{x}^*_{\tilde{t}}\right)$, for all $k \in \{1..K\}$, set $\xi^*_{a,t,k} = -\ln\left(1 - \left[\left(1 - P\_detect_{q,k,r}\right) \times \left(1 - e^{-H_{a,t,k}}\right)\right]\right)$, where q is the RCT configuration type, and r is the classification of arc a.

ii. For all $a \in \mathrm{A} \backslash \left(Path_{\tilde{x}^*_{\tilde{t}'},\tilde{t}}, \tilde{x}^*_{\tilde{t}}\right)$, for all $k \in \{1..K\}$, for $t = \tilde{t} + t_0$:

(1) Set $\rho_{a,t-1,k} = \frac{1 - e^{-\xi^*_{a,t-1,k}}}{P\_use_{a,t-1} \times P\_eff_k}$.

(2) Set $\rho_{a,t,k} = P\_em_{a,t-1,k} \times \left(1 - \rho_{a,t-1,k}\right) + \left(1 - P\_use_{a,t-1}\right) \times \rho_{a,t-1,k}$.

(3) Set $\xi^*_{a,t,k} = -\ln\left(1 - \rho_{a,t,k} \times P\_use_{a,t} \times P\_eff_k\right)$.

c. If in the solution path no arc is cleared at time $\tilde{t} - 1$, i.e. if $\nexists\ \tilde{x}^*_{\tilde{t}}$:
For all $a \in$ A, for all $k \in \{1..K\}$, for $t = \tilde{t} + t_0$:

i. Set $\rho_{a,t-1,k} = \frac{1 - e^{-\xi^*_{a,t-1,k}}}{P\_use_{a,t-1} \times P\_eff_k}$.

ii. Set $\rho_{a,t,k} = P\_em_{a,t-1,k} \times \left(1 - \rho_{a,t-1,k}\right) + \left(1 - P\_use_{a,t-1}\right) \times \rho_{a,t-1,k}$.

iii. Set $\xi^*_{a,t,k} = -\ln\left(1 - \rho_{a,t,k} \times P\_use_{a,t} \times P\_eff_k\right)$.

d. For all $a \in A, k \in \{1..K\}$,
Set $\bar{\xi}^*_{a,t,k} = \min\left[\xi^*_{a,t,k}, \overline{H}_{a,t,k}\right]$.

e. Reset $t = t + 1$.

This heuristic prunes paths by seeking to minimize reduced conditional hazard on the graph at each stage. Step 0 initializes the parameters of the heuristic. The parameter $\boldsymbol{H}_{\tilde{t}}$ is the vector of original hazard values $H_{a,\tilde{t},k}$ for all arcs in the graph, for all IED types at stage *t*. Step 1 is the label setting heuristic, carried out in stages, in which the label for state $x_{\tilde{t}}$ is set to the minimum total reduced conditional stage $\tilde{t}$ hazard from the set of reduced hazards corresponding to arriving transitions from previous states. Reduced original hazard values corresponding to this "best" transition are also computed and stored for state $x_{\tilde{t}}$. Each transition that yields a minimal reduced conditional hazard for a state is stored in the ***Path*** matrix.

Steps 2, 3, and 4 produce the outputs of the heuristic. Step 2 sets the mission's reduced original hazard values and reduced conditional hazard values to the input original and conditional hazard values, respectively, for stages prior to the mission start time t_0. Step 3 recovers the solution mission path from the ***Path*** matrix. Step 4 uses the mission path and the recursions for determining hazard reductions to calculate the mission's reduced original hazard and reduced conditional hazard, from the mission start time to the end of the planning horizon. The outputs of this algorithm are the solution path, $\{\tilde{x}_0^*, \tilde{x}_{\tilde{t}_1}^*, \tilde{x}_{\tilde{t}_2}^* .. \tilde{x}_\tau^*\}$, where $\tilde{x}_0^* = \tilde{x}_\tau^* = base_h$ (the set RCT location at the mission start and end stages), and the original and conditional hazard reductions, $\xi_{a,t,k}^*$ and $\bar{\xi}_{a,t,k}^*$, for all arcs and IED-types in the model, for all stages in the planning horizon.

To demonstrate why the heuristic does not necessarily provide an optimal solution, consider the earlier example using the simple three-node, four-arc graph in Figure 4-5. Assume again a four-stage mission with each arc taking one stage to clear, the center node serving as the RCT base, and equal conditional and original hazard values. In this example, however, suppose hazard values increase by some fixed amount γ for all arcs, for all stages, with one exception: arc (2,1) has a hazard increase (*M*) after the third time-step that is much larger than γ. Again, we allow the RCT to be 100% effective. Finally, we set the initial hazard values at 0 for the arcs connecting nodes 1 and 3, and at some $\varepsilon > 0$ for the arcs connecting nodes 1 and 2.

Applying the heuristic algorithm, we find that the shortest path (i.e. that path that results in minimizing total reduced hazard per stage) from the origin to state $\tilde{x}_2 = 1$ clears the arcs (1,2) and (2,1), resulting in a total reduced hazard of 5γ at stage 2, which is only slightly lower than the alternative path that clears arcs (1,3) and (3,1), resulting in a reduced hazard of 5γ+2ε. From this state there is a choice to clear back to node 2 or go on to node 3 before returning to 1. Again, the algorithm determines it is best to clear arcs (1,2) and (2,1) in order to minimize the impact of the *M*-increase in hazard on arc (2,1). The heuristic returns the path {(1,2),(2,1),(1,2),(2,1)} with a stage-4 total reduced hazard of 9γ. Contrast this cost with the reduced hazard resulting from clearing arcs (1,3) and (3,1) in the first two steps, followed by arcs (1,2) and (2,1), which is 6γ. Reduced hazard results summed across all four stages also show that the alternate path {(1,3),(3,1),(1,2),(2,1)} is the better choice by an additional hazard reduction of about 4γ.

A small analysis of this heuristic and others against exact dynamic programming methods is in Appendix C.

4.2.4.1.5 Alternate Path Generation Heuristic

In order to out-perform the primary Path Generation Heuristic, we propose an alternative that uses the same methodology, but a different labeling function. Rather than minimizing the instantaneous reduced conditional hazard at each stage, we maximize the difference between the conditional hazard value inputs and the reduced conditional hazard values computed and summed over a user-defined number of stages beyond the current stage. This difference is the reward for clearing an arc and is to be maximized. Using this labeling function as opposed to instantaneous reduced hazard makes the heuristic less greedy by considering both the immediate and the enduring effects of clearing each arc. The distinction is evident in Figure 4-8, which shows the effect of route clearance on hazard on an arc. In this example, the original hazard values were also used as conditional hazard values.

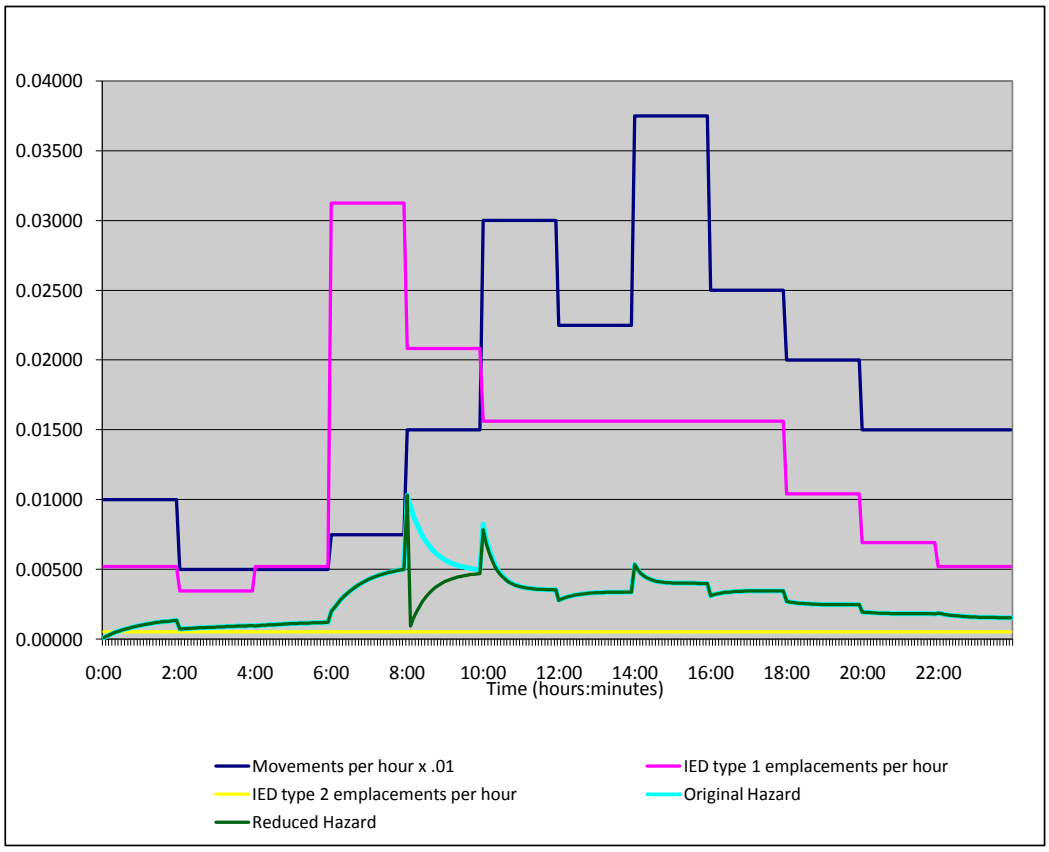

Figure 4-8: Effect of Route Clearance on Hazard on an Arc

The plot depicts a mission plan in which a route clearance team clears the arc at about 8:00, causing an immediate reduction in hazard. In the primary heuristic, the cost (or label) for this specific arc is the reduced hazard line, which we seed to minimize instantaneously at each stage. The alternate Path Generation Heuristic, however, considers the area between the original (or updated) hazard values resulting from a single arc clearance as the reward to be maximized. For example, for an arc clearance at 8:00, the alternate heuristic computes reduced conditional hazard for Δ (a user-defined parameter) stages after 8:00, subtracts them from the conditional hazard values for the same stage, and takes the sum of these reductions as the reward for clearing the arc at 8:00.

This method remains approximate because, like the primary heuristic, it still prunes paths without considering their effects on the costs associated with future decisions in order to keep the state space condensed. The alternate heuristic gives the same solution as the primary heuristic in

the three-node, four-arc example of the previous paragraph. By accounting for the enduring effects of arc clearance, however, the alternate heuristic makes a more complete estimate of the reward associated with clearing each arc when setting labels.

4.2.4.2 Controls for Path Set Production

Each Path Generation method provided above is deterministic; given the same set of inputs it will produce the same outputs. In order to generate a diverse set of mission paths to optimize over, we can run the Path Generation function many times with different inputs each time. In the algorithm we propose, depicted in Figure 4-2, we methodically vary the input RCT, mission start time, and conditional hazard values to obtain different results from Path Generation. Specifically, the controls are as follows:

0. For first iteration, set conditional hazard equal to the original hazard, i.e. $\overline{\boldsymbol{H}} = \boldsymbol{H}$.
   Initialize a user-defined start-time increment ψ, e.g. $\psi = 1$ hour.
   Set $\bar{\imath} = 1$.
1. For each RCT $h \in \{1..\bar{h}\}$:
   a. Randomize the first start time. Uniformly pick a start time X on $(0, \psi]$ and determine which stage *t* this start time falls in. Set $\bar{t}_0^1$ equal to this stage, i.e., $\bar{t}_0^1 = \lceil X/\delta \rceil$.
   b. Set $\bar{\jmath} = 1$.
   c. While $\bar{t}_0^{\bar{\jmath}} < T - \max[T\_max_h]$.
      i. Run the Path Generation function using the current RCT, $\bar{t}_0^{\bar{\jmath}}$, and conditional hazard values ($\overline{\boldsymbol{H}}$), along with the original hazard values ($\boldsymbol{H}$).
      ii. Set $\boldsymbol{\nu}^{\bar{\jmath}} = \boldsymbol{\xi}^*$, $\bar{\boldsymbol{\nu}}^{\bar{\jmath}} = \bar{\boldsymbol{\xi}}^*$, $\omega^{\bar{\jmath}} = h$, and $\varphi^{\bar{\jmath}} = \{\tilde{x}_0^*, \tilde{x}_{\tilde{t}_1}^*, \tilde{x}_{\tilde{t}_2}^* .. \tilde{x}_{\tau}^*\}$.
      iii. Re-set $\bar{\jmath} = \bar{\jmath} + 1$.
      iv. Set $\bar{t}^{\bar{\jmath}} = \bar{t}^{\bar{\jmath}-1} + \psi$.
   d. Set $\bar{J} = \bar{\jmath} - 1$.
   e. Run the Path Optimize function inputting $\overline{\boldsymbol{H}}$ as the existing hazard, $\bar{\boldsymbol{\nu}}^{\bar{\jmath}}$ $\forall \bar{\jmath} \in \{1..\bar{J}\}$ as the path reduced hazard vectors, $\bar{t}_0^{\bar{\jmath}}$ $\forall \bar{\jmath} \in \{1..\bar{J}\}$ as the as the path start times, and $\omega^{\bar{\jmath}}$ as the path RCTs.
   f. For all $\bar{\jmath}$ such that $x_{\bar{\jmath}} = 1$, i.e. for all paths included in the solution outputted by the Path Optimize function:
      i. Set $\boldsymbol{\eta}^{\bar{\imath}} = \boldsymbol{\nu}^{\bar{\jmath}}$.
      ii. Set $\Omega^{\bar{\imath}} = \omega^{\bar{\jmath}}$.
      iii. Set $t_0^{\bar{\imath}} = \bar{t}_0^{\bar{\jmath}}$.

iv. Set $\Phi^{\bar{\iota}} = \varphi^{\bar{J}}$

v. Re-set $\bar{\iota} = \bar{\iota} + 1$.

g. Set $\bar{\boldsymbol{H}} = \boldsymbol{\gamma}^{*}$, the optimal hazard output from the Path Optimization.

This control algorithm randomizes an initial start time, and then generates paths at regular intervals for a RCT over the entire planning horizon. Once these paths are generated, it uses the Path Optimization function, discussed in the next section, to select a subset of the generated paths to retain in the Path Set. The other generated paths are not kept. Finally, it uses the optimal hazard output from the Path Optimization function as the conditional hazard input to generate paths for the next RCT. The output of the Path Set Production process is a set of mission paths with the following information:

$\boldsymbol{\eta}^{\bar{\iota}}$	The reduced original hazard associated with mission path $\bar{\iota}$.
$\Omega^{\bar{\iota}}$	The RCT assigned to clear mission path $\bar{\iota}$.
$t_0^{\bar{\iota}}$	The start time for mission path $\bar{\iota}$.
$\Phi^{\bar{\iota}}$	The sequence of nodes that comprise mission path $\bar{\iota}$.

This information, along with previously generate Path Sets, is inputted into the Path Optimization function.

4.2.5 Path Optimization

The Path Optimization function takes as input existing hazard values ($H'_{a,t,k} \forall a \in A$, $t \in \{1..T\}, k \in \{1..K\}$) and a set of mission paths Y, $\bar{\iota} \in Y$, along with their associated reduced hazard values ($\eta'^{\bar{\iota}}_{a,t,k}$), start times ($t'^{\bar{\iota}}_0$), and assigned RCTs ($\Omega'^{\bar{\iota}}$). This function determines the best feasible combination of these paths, given the mission constraints on each RCT, to minimize hazard on the graph over the planning horizon. In order to compare different combinations of paths, we define how multiple route clearance missions affect the hazard on each arc. Let $Y_{feasible}$ denote a subset of mission paths that is possible for the RCTs to execute within their capabilities and constraints, and variable $\gamma_{a,t,k}$ be the *residual* hazard on arc *a*, at stage *t*, for IED type *k*, assuming all missions in $Y_{feasible}$ are carried out. We define the residual hazard (hazard remaining after multiple clearance missions are conducted), $\gamma_{a,t,k}$ as follows:

$$\gamma_{a,t,k} := \min_{\bar{\iota} \in Y_{feasible}} \left(\eta'^{\bar{\iota}}_{a,t,k} \right), \forall a \in A, t \in \{1..T\}, k \in \{1..K\}.$$

In defining each residual hazard value as the minimum of the path hazard values in the set of "active" missions, we do not allow for multiplicative effects of multiple clearances of the same

arc. If two RCTs were to clear the same arc in two adjacent stages, the residual hazard on the arc for subsequent stages for each type of IED would reflect the hazard reduction of only the most effective RCT. Using this rule for the effects of multiple clearances on an arc limits the hazard reduction of each arc clearance to that of the most effective RCT clearing the arc, and does not assume each clearance on an arc is an independent trial. We imply in this definition that, if an IED exists on an arc and is not found by a RCT, then another RCT with less or equal effectiveness clearing the arc at the same stage is also not going to find it. Alternatively, a more effective RCT clearing the arc will reduce the hazard values to the same level whether or not any less effective RCTs have cleared it. This method of evaluating hazard reduction as a result of multiple arc clearances is a generalization of the methods we used to compute hazard reductions in the Path Generation function for cases in which an arc was cleared more than once.

Before optimizing, the set of constraints that define the feasible combinations of paths must be determined. This is completed by searching all of the mission start times generated for each RCT and writing constraints to ensure proper mission spacing (subject to T_max_h and $dwell_h$), and density (from Max_day_h). For any two mission paths $\bar{\imath}$ and $\bar{\jmath}$ such that $\Omega'^{\bar{\imath}} = \Omega'^{\bar{\jmath}} = h$ and $\left|t'^{\bar{\imath}}_0 - t'^{\bar{\jmath}}_0\right| \leq T\_max_h + dwell_h$, we create the mission spacing constraint:

$$y_i + y_j \leq 1.$$

To ensure the number of missions per day does not exceed Max_day_h for any RCT, we let $Y_{h,d}$ be the subset of paths for which $\Omega'^{\bar{\imath}} = h$ and $t'^{\bar{\imath}}_0 \in \bar{D}^d$, where $\bar{D}^d$ is the set of stages that comprise day *d*, and we create the constraint:

$$\sum_{i \in Y_{h,d}} y_i \leq Max\_day_h, \forall d \in \{1..D\}, \forall h \in \{1..\bar{h}\}.$$

The variables $y_i, i \in Y$ in these constraints and in the mixed integer program are binary decision variables that indicate which mission paths are to be executed. A variable y_i takes value 0 if path *i* is not included in the feasible subset of paths to be executed, or value 1 if path *i* is to be executed.

We present two approaches to Path Optimization, each using mixed integer programming. The approach forms a large mixed integer program and applies Bender's Decomposition to dissect it into more manageable components to be solved individually. The second method requires a significant preprocessing effort but forms a smaller mixed integer program that solves faster.

4.2.5.1 First Mixed Integer Formulation and Bender's Decomposition

We form the following mixed integer program:

$$\text{minimize} \sum_{k\in\{1..K\}} \sum_{t\in\{1..T\}} \sum_{a\in A} \gamma_{a,t,k}$$

subject to

$$y_{\bar{\imath}} + y_{\bar{\jmath}} \leq 1, \ \forall \{(i,j) | \ \Omega'^{\bar{\imath}} = \Omega'^{\bar{\jmath}} = h, \left|t'^{\bar{\imath}}_0 - t'^{\bar{\jmath}}_0\right| \leq T\_max_h + dwell_h\} \quad (1)$$

$$\sum_{\bar{\imath}\in Y_{h,d}} y_{\bar{\imath}} \leq Max\_day_h, \forall\, h \in \left\{1 \ldots \bar{h}\right\}, \forall d \in \{1,..,D\} \quad (2)$$

$$\gamma_{a,t,k} = \sum_{\bar{\imath}\in Y} \eta'^{\bar{\imath}}{}_{a,t,k}\, x^{\bar{\imath}}_{a,t,k}, \forall a \in A, t \in \{1,\ldots,T\}, k \in \{1,\ldots K\} \quad (3)$$

$$x^{\bar{\imath}}_{a,t,k} \leq y_{\bar{\imath}}, \forall a \in A, t \in \{1,\ldots,T\}, k \in \{1,\ldots K\}, \bar{\imath} \in Y \quad (4)$$

$$\sum_{\bar{\imath}\in Y} x^{\bar{\imath}}_{a,t,k} \geq 1, \forall\, a \in A, t \in \{1,\ldots,T\}, k \in \{1,\ldots K\} \quad (5)$$

$$y_{\bar{\imath}} \in \{0,1\}, \ \forall\, \bar{\imath} \in Y \quad (6)$$

The objective function to be minimized is the sum of all residual hazard on the graph. Constraint sets (1) and (2) are the mission compatibility constraints determined as discussed above. Constraint set (3) forces the resulting residual hazard values ($\gamma_{a,t,k}$) to be equal to some linear combination of the reduced hazard values associated with each path, ($\eta'^{\bar{\imath}}{}_{a,t,k}, \forall \bar{\imath} \in Y$), which constraint sets (4) and (5) further define. Constraint set (4) ensures that a path $\bar{\imath}$ must be in the active set of feasible paths, i.e. $y_{\bar{\imath}} = 1$, in order for its reduced hazard values to have positive coefficients in setting the lower bound on the residual hazard variables. Constraint set (5) forces the sum of the coefficients of each linear combination in constraint set (3) to be at least 1. With an objective function seeking to minimize the residual hazard variables, constraint sets (3), (4), and (5) together ensure that these variables in the optimum solution will be defined as stated previously:

$$\gamma_{a,t,k} = \min_{\bar{\imath}\in Y_{feasible}} \left(\eta'^{\bar{\imath}}{}_{a,t,k}\right), \forall a \in A, t \in \{1..T\}, k \in \{1..K\}.$$

Notice that we can eliminate constraint set (3) and the γ variables by re-writing the objective function as:

$$\text{minimize} \sum_{k\in\{1..K\}} \sum_{t\in\{1..T\}} \sum_{a\in A} \sum_{\bar{\imath}\in Y} \eta'^{\bar{\imath}}{}_{a,t,k}\, x^{\bar{\imath}}_{a,t,k}$$

A problem arises again in the number of variables and constraints in the mixed integer program. If m is the number of arcs, T is the number of stages, K is the number of IED types, $|Y|$ is the number of paths, and D is the number of days in the planning horizon, we have $|Y|$ integer variables, $|Y|mTK$ continuous variables, and approximately $mTK+|Y|mTK+3|Y|D$ constraints (assuming about 2 conflicting mission constraints per day per RCT). For a 120-arc graph, optimizing over 10 mission paths of 60 stages each, two IED types, and a 1440-stage planning horizon, we have 10 integer variables, 144,000 continuous variables, and 158,550 constraints. This program is comparable in size to the mixed-integer program considered for the Path Generation function, which became too large to solve efficiently as the size of the network increased.

However, this mixed-integer program has a special structure that can be exploited for more efficient solving. First, it has a much smaller number of integer variables. Secondly, there is only one constraint set (4) linking the integer variables to the continuous variables; all other constraints deal strictly with either integer variables or with continuous variables. Furthermore, there are no constraints that couple different arcs, different stages, or different IED types. In other words, once the values for the integer variables have been determined, this problem becomes a sequence of mTK separate, simple minimizations of $|Y|$ reduced hazard values. This problem set-up is ideal for a constraint generating technique, such as Bender's Decomposition [BT097].

4.2.5.1.1 Bender's Decomposition

We now re-formulate the mixed-integer program defined above, referred to as the *master problem*, using Bender's Decomposition. The problem is also decomposed into series of *sub-problems*, which are used to generate constraints to be applied to the reduced master problem. Rather than completely decompose the problem into all mTK sub-problems, we decompose only by arc, giving us m sub-problems. We define the reduced master problem as an aggregated objective function (by arc) subject only to the constraints pertaining to the integer variables, and an additional constraint bounding the aggregated residual hazard values to be nonnegative:

$$\text{minimize} \sum_{a \in A} \gamma'_a$$

subject to

$$y_{\bar{\imath}} + y_{\bar{\jmath}} \leq 1, \ \forall \{(i,j) | \ \Omega'^{\bar{\imath}} = \Omega'^{\bar{\jmath}} = h, \left| t'^{\bar{\imath}}_0 - t'^{\bar{\jmath}}_0 \right| \leq T\_max_h + dwell_h\} \quad (1)$$

$$\sum_{\bar{\imath} \in Y_{h,d}} y_{\bar{\imath}} \leq Max\_day_h, \forall \ h \in \{1 \ldots \bar{h}\}, \forall d \in \{1, .., D\} \quad (2)$$

$$y_{\bar{\imath}} \in \{0,1\}, \ \forall \ \bar{\imath} \in Y \quad (6)$$

$$\gamma'_a \geq 0, \forall a \in A \quad (7)$$

The variables in the objective function γ'_a represent the total residual hazard on each arc *a*:

$$\gamma'_a = \sum_{t \in \{1..T\}} \sum_{k \in \{1..K\}} \gamma_{a,t,k}.$$

This reduced master problem has $|Y|$ integer variables, *m* continuous variables, and approximately $m+3|Y|D$ constraints. Solving this master problem in this form is very fast and results in an optimal objective value of 0 for any feasible combination of *y* variables.

We now define a sub-problem for each arc ($a \in A$) in the network. Each sub-problem receives the solution $\boldsymbol{y}^*$, $\boldsymbol{\gamma}'^*$ from the master problem, and solves:

$$\text{minimize} \sum_{k \in \{1..K\}} \sum_{t \in \{1..T\}} \sum_{\bar{\imath} \in Y} \eta'^{\bar{\imath}}_{a,t,k} \, x^{\bar{\imath}}_{a,t,k}$$

subject to

$$x^{\bar{\imath}}_{a,t,k} \leq y^*_{\bar{\imath}}, \forall \ t \in \{1, \ldots, T\}, k \in \{1, \ldots K\}, \bar{\imath} \in Y \quad (4^*)$$

$$\sum_{\bar{\imath} \in Y} x^{\bar{\imath}}_{a,t,k} \geq 1, \forall \ t \in \{1, \ldots, T\}, k \in \{1, \ldots K\} \quad (5)$$

This problem is a linear program with $|Y|TK$ variables and $TK+|Y|TK$ constraints (1,200 variables and 1,320 constraints, using the 160-arc, ten 60-stage paths, 2-IED type, 1,440-stage problem). Optimization software packages such as CPLEX can solve a problem of this size very rapidly. (Recall that $\gamma_{a,t,k} = \sum_{\bar{\imath} \in Y} \eta'^{\bar{\imath}}_{a,t,k} \, x^{\bar{\imath}}_{a,t,k}$).

The solution to a sub-problem is of interest to us if the optimal objective value is greater than its corresponding variable solution in the reduced master problem, i.e. if for the sub-problem for arc *a*,

$$\sum_{k \in \{1..K\}} \sum_{t \in \{1..T\}} \sum_{\bar{\imath} \in Y} \eta'^{\bar{\imath}}_{a,t,k} \, x^{\bar{\imath}}_{a,t,k} = \sum_{t \in \{1..T\}} \sum_{k \in \{1..K\}} \gamma_{a,t,k} > \gamma'^*_a.$$

If this inequality holds, we introduce a constraint on γ'_a into the reduced master problem, as the solution value (γ'^*_a) violates the condition $\gamma'_a = \sum_{t \in \{1..T\}} \sum_{k \in \{1..K\}} \gamma_{a,t,k}$, where

$$\gamma_{a,t,k} = \min_{\bar{\imath} \in Y_{feasible}} \left({\eta'}^{\bar{\imath}}{}_{a,t,k} \right), \forall a \in A, t \in \{1..T\}, k \in \{1..K\}.$$

We can determine the constraint to add from the sub-problem's dual solution, $\boldsymbol{\pi} = \left(\boldsymbol{\pi}^{4^*}, \boldsymbol{\pi}^5\right)$:

$$\gamma'_a \geq \mathbf{1}^{\mathrm{T}} \boldsymbol{\pi}^5 + \sum_{\bar{\imath} \in Y} \mathbf{1}^{\mathrm{T}} \boldsymbol{\pi}_{\bar{\imath}}^{4^*} \times y_{\bar{\imath}},$$

where $\boldsymbol{\pi}^5$ is the vector of dual variables for constraint set (5), which are nonnegative, $\boldsymbol{\pi}_{\bar{\imath}}^{4^*}$ is the vector of dual variables for constraints in set (4*) on path $\bar{\imath}$, which are non-positive for all $\bar{\imath}$, and $\mathbf{1}$ is a vector of ones.

4.2.5.1.2 Implementation of Bender's Decomposition

To increase the efficiency of the Bender's Decomposition algorithm, we introduce a path 0, corresponding to no route clearance missions and using the existing hazard values as the path's reduced hazard values, into the set of mission paths in the optimization problem. We then set the initial solution to the reduced master problem as $y_0^* = 1$, and $y_{\bar{\imath}}^* = 0, \quad \forall\, \bar{\imath} \neq 0$, which returns zero as the optimal objective value and solution to all $\boldsymbol{\gamma}'$ variables in the first iteration. Using this initial solution to run each of the sub-problems provides a set of constraints that relate the decision to include a path $\bar{\imath}$ in the solution to its specific effect on the optimal value for $\boldsymbol{\gamma}'$, given that no other missions are conducted. These constraints take the form:

$$\gamma'_a \geq \sum_{k \in \{1..K\}} \sum_{t \in \{1..T\}} {H'}_{a,t,k} - \sum_{i \in Y} \bar{c}_{\bar{\imath}} \times y_{\bar{\imath}},$$

where $\bar{c}_{\bar{\imath}} = -\mathbf{1}^{\mathrm{T}} \boldsymbol{\pi}_{\bar{\imath}}^{4^*}$. These constraints provide good starting points for determining γ'_a values in subsequent reduced master problem iterations.

Another technique to improve the efficiency of this algorithm involves arcs in the network that are never cleared by any of the mission paths under consideration. For an arc $\tilde{a}$ meeting this description, the constraint generated by the corresponding sub-problem in the initial iteration is:

$$\gamma'_{\tilde{a}} \geq \sum_{k \in \{1..K\}} \sum_{t \in \{1..T\}} {H'}_{a,t,k} - \sum_{\bar{\imath} \in Y} 0\, y_{\bar{\imath}}.$$

Decisions between different subsets of missions do not affect the residual hazard on arcs meeting this criterion, and future sub-problem solutions for this arc will not contribute new constraints to the problem. We can therefore identify these arcs in the initial iteration, introduce the resulting constraints into the reduced master problem, and then omit solving their associated sub-problems for the remaining iterations. This method can be extended so that during each iteration, only

sub-problems corresponding to arcs that are cleared by some path $\bar{\imath}$ for which $y_{\bar{\imath}}^* = 1$ (from the solution to the reduced master problem) and $\bar{c}_{\bar{\imath}} > 0$ are solved, where $\bar{c}_{\bar{\imath}} = -\mathbf{1}^{\mathrm{T}}\boldsymbol{\pi}_{\bar{\imath}}^{4^*}$ from the arcs' first sub-problem iteration.

4.2.5.1.3 Bender's Decomposition Algorithm

This section outlines the Bender's Decomposition Algorithm for the Path Optimization function. The reduced master problem and sub-problems are as discussed above and will not be illustrated again.

0. Set the solution to the reduced master problem to $y_0^* = 1,\ y_{\bar{\imath}}^* = 0\ \forall\ \bar{\imath} \neq 0, {\gamma'}_a^* = 0\ \forall\ a \in A$.
   Set $A_{in} = A$
   Solve each sub-problem in the dual.
   a. If the optimal objective value for sub-problem a is greater than ${z'}_a^*$,
      i. Add the constraint ${\gamma'}_a \geq \sum_{k\in\{1..K\}}\sum_{t\in\{1..T\}} {H'}_{a,t,k} - \sum_{i\in Y} -\mathbf{1}^{\mathrm{T}}\boldsymbol{\pi}_{\bar{\imath}}^{4^*} \times y_{\bar{\imath}}$ to the reduced master problem.
      ii. If $-\mathbf{1}^{\mathrm{T}}\boldsymbol{\pi}_{\bar{\imath}}^{4^*} = 0\ \forall\ \bar{\imath}$, remove arc a from A_in.
   b. If the optimal objective value for the sub-problem is equal to ${z'}_a^*$, remove arc a from A_in.
1. Solve the reduced master problem with the additional constraints. Re-set $\boldsymbol{y}^*$ and $\boldsymbol{\gamma'}^*$ to the new solution values.
2. For each arc in *A_in*, solve the corresponding sub-problem in the dual.
   If the optimal objective value for sub-problem a is greater than ${\gamma'}_a^*$, add the constraint ${\gamma'}_a \geq \mathbf{1}^{\mathrm{T}}\boldsymbol{\pi}^5 + \sum_{\bar{\imath}\in Y} \mathbf{1}^{\mathrm{T}}\boldsymbol{\pi}_{\bar{\imath}}^{4^*} \times y_{\bar{\imath}}$ to the reduced master problem.
3. If no constraints were added to the reduced master problem in step 2, terminate with optimal solution $\boldsymbol{y}^*$ and optimal residual hazard $\boldsymbol{\gamma}^* = \boldsymbol{\gamma'}^*$. Otherwise, return to step 1 for the next iteration.

This algorithm terminates with the optimal solution to the master problem. Initializing the solution to the master problem as directed in step 0 ensures feasibility for all of the sub-problems for all iterations. Although the reduced master problem grows with each iteration, it remains much smaller than the original master problem and, along with the sub-problems, is possible to solve efficiently.

4.2.5.2 Second Mixed Integer Formulation

While the reduced master problem and sub-problems in the first formulation do not take long to solve individually, there is no guarantee of fast convergence to an optimal solution. Furthermore, because the residual hazard values determined in each reduced master problem iteration are not adequately constrained before the algorithm reaches the optimal solution, they do not accurately portray the combined effects of the iteration's solution.

To address these two shortcomings, we present a second integer programming formulation that takes advantage of the structure of this problem. We begin with the goal of creating a problem of the following form:

maximize $\boldsymbol{c^T y}$

subject to

$\boldsymbol{Ay \leq b}$, (1,2)

$y_{\bar{\iota}} \in \{0,1\},\ \forall\, \bar{\iota} \in Y$ (6)

In other words, we seek a formulation in which the rewards assigned to each path $\bar{\iota}$ are fixed, so that our objective function is a linear combination of our path variables. The program determines the optimal combination of paths subject to the feasibility constraints, which are the same as they were in the previous formulation, but without constraints and variables set up to bound each residual hazard value. An intuitive choice for the reward value associated with path $\bar{\iota}$ is

$$c_{\bar{\iota}} = \sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\left(H'_{a,t,k} - \eta'^{\bar{\iota}}_{a,t,k}\right).$$

This value is the total difference between the reduced hazard for path $\bar{\iota}$ and the existing hazard inputs. It is a measure of overall hazard reduction associated with each path. The problem with it is that it does not account for situations in which two RCTs clear the same arc. Using our definition for residual hazard on an arc following multiple clearances, we have the reward for a feasible set of clearances $Y_{feasible}$:

$$\sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\max_{\bar{\iota}\in Y_{feasible}}\left(H'_{a,t,k} - \eta'^{\bar{\iota}}_{a,t,k}\right).$$

From this expression, it appears the reward vector $\boldsymbol{c}$ depend on the values assumed by the variables, significantly complicating the problem. To overcome this dependency, we introduce

additional *overlap* variables into the problem with their own associated costs. Like the path variables $y_{\bar{\imath}}$, these variables assume a value from the binary integer set {0,1}. As an example, suppose that two mission paths (1 and 2) under consideration each have an RCT clear arc $\tilde{a}$, at time $\tilde{t}-1$, in a problem in which we consider only one IED type. Also assume that:

$$H'_{\tilde{a},\tilde{t},1} = 1$$
$$\eta'^{1}_{\tilde{a},\tilde{t},1} = 0.5$$
$$\eta'^{2}_{\tilde{a},\tilde{t},1} = 0.2$$
$$H'_{a',t',1} = \eta'^{1}_{a',t',1} = \eta'^{2}_{a',t',1}, \forall\{(a',t')|(a',t') \neq (\tilde{a},\tilde{t}), a' \in A, t' \in \{1..T\}\}$$

From these values, we can compute the reward values (hazard reductions) associated with each path individually:

$$c_1 = \sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\left(H'_{a,t,k} - \eta'^{1}_{a,t,k}\right) = \left(H'_{\tilde{a},\tilde{t},1} - \eta'^{1}_{\tilde{a},\tilde{t},1}\right) = 0.5$$
$$c_2 = \sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\left(H'_{a,t,k} - \eta'^{2}_{a,t,k}\right) = \left(H'_{\tilde{a},\tilde{t},1} - \eta'^{2}_{\tilde{a},\tilde{t},1}\right) = 0.8$$

If our feasible solution set consists of these two missions, then we have as a total reward

$$c_1 + c_2 = 1.3$$

However, we know from our definition of residual hazard after multiple clearances that:

$$\sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\gamma_{a,t,k} = \sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\min_{\bar{\imath}\in Y_{feasible}}\left(\eta'^{\bar{\imath}}_{a,t,k}\right)$$
$$\sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}}\left(H'_{a,t,k} - \gamma_{a,t,k}\right) = H'_{\tilde{a},\tilde{t},k} - \gamma_{\tilde{a},\tilde{t},k} = 0.8$$

The realized reward for executing both paths is a reduction in hazard of 0.8. This is the same reduction we would have had if we had only executed the second route clearance mission and omitted the first. To account for this discrepancy, we introduce an overlap variable $z_{1\cap 2}$. This variable takes a value of 1 if both paths are included in the feasible solution set, i.e. variables $y_1 = y_2 = 1$. We assign a reward value $d^{1\cap 2}$ to this decision variable. In the example above, this value is -0.5.

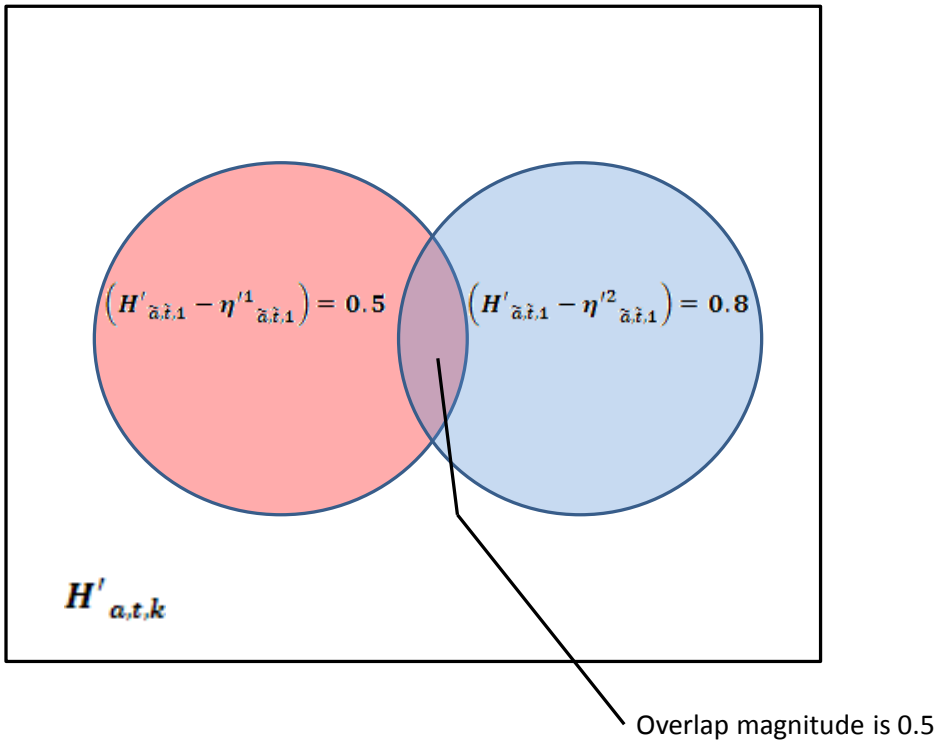

Figure 4-9: Hazard Reduction Overlap on Arc a, Stage t, for IED Type k

The idea of adding variables to account for hazard reduction overlaps can be visualized in a Venn Diagram, and explained using some basic concepts from probability theory. We relate the entire sample space to the existing hazard value for some arc, stage, and IED type. Each path has an associated hazard reduction for the arc, stage, and IED type under consideration, which represent different events that can occur. These events are not disjoint; they have overlap. Figure 4-9 shows a Venn diagram to illustrate this overlap in the given example. For two probabilistic events A_1 and A_2, the probability of their union is

$$\text{Prob}(A_1 \cup A_2) = \text{Prob}(A_1) + \text{Prob}(A_2) - \text{Prob}(A_1 \cap A_2).$$

The last term accounts for the probability associated with the intersection of the two events being included in each of the first two terms. Similarly, the hazard reduction on arc a, stage t, for IED type k resulting from clearing both paths 1 and 2 is

$$\left(H'_{a,t,k} - \eta'^{1}_{a,t,k}\right) + \left(H'_{a,t,k} - \eta'^{2}_{a,t,k}\right) + d^{1\cap 2}_{a,t,k}.$$

From our definition of residual hazard, we realize that the magnitude of this hazard reduction overlap can be defined as

$$\min_{\bar{\imath}\in\{1,2\}}\left(H'_{a,t,k}-\eta'^{\bar{\imath}}_{a,t,k}\right),$$

and therefore $d^{1\cap 2}_{a,t,k}$ is

$$d^{1\cap 2}_{a,t,k}:=-\min_{\bar{\imath}\in\{1,2\}}\left(H'_{a,t,k}-\eta'^{\bar{\imath}}_{a,t,k}\right).$$

We take the negative minimum to indicate that the overlap magnitude is ultimately subtracted when making hazard reduction union calculations.

Naturally, we must also consider cases in which more than two mission paths clear an arc in the same stage or stages close enough in time to result in overlapping path hazard reductions. Again we recall from probability that in the case of three events with overlapping outcome sets, the probability of their union is

$$\begin{aligned}\mathrm{Prob}(A_1\cup A_2\cup A_3)&\\ &=\mathrm{Prob}(A_1)+\mathrm{Prob}(A_2)+\mathrm{Prob}(A_3)-\mathrm{Prob}(A_1\cap A_2)-\mathrm{Prob}(A_1\cap A_3)\\ &\quad-\mathrm{Prob}(A_2\cap A_3)+\mathrm{Prob}(A_1\cap A_2\cap A_3).\end{aligned}$$

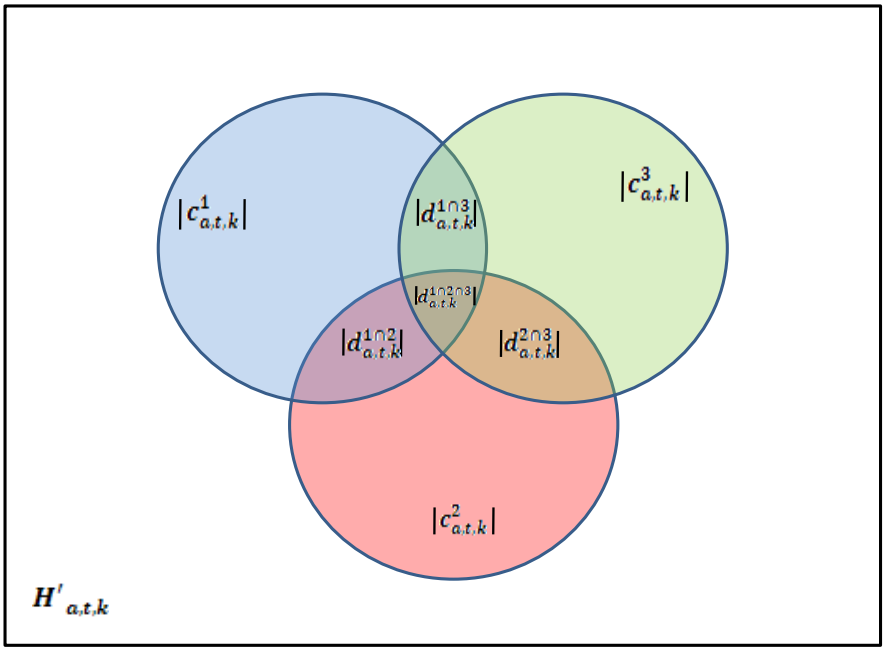

Figure 4-10: Three-Path Hazard Reduction Overlap

The development of this expression can be found by carefully examining the Venn diagram in Figure 4-10. By simply adding the probabilities associated with the three events, each two-event intersection is added twice, while the three-event intersection is added all three times. The expression given above ensures that the probability associated with each intersection is only counted once. Again translating this overlap concept to our definition of stage t residual hazard on arc a for IED type k, we find that the magnitude of the overlap between three paths (1, 2, and 3) is

$$\min_{\bar{\imath}\in\{1,2,3\}}\left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k}\right),$$

and the reward coefficient associated with this overlap is

$$d^{1\cap 2\cap 3}_{a,t,k} := \min_{\bar{\imath}\in\{1,2,3\}}\left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k}\right).$$

In this case, the reward coefficient takes a positive value, indicating that the magnitude of this overlap is added when calculating residual hazard resulting from this union of mission paths.

Generalizing, we find that for executing the feasible set of paths $Y_{feasible}$, the reward in terms of hazard reduction on arc a, stage t, for IED type k can be derived as:

$$H'_{a,t,k} - \gamma_{a,t,k} = H'_{a,t,k} - \min_{\bar{\imath} \in Y_{feasible}} \left(\eta'^{\bar{\imath}}_{a,t,k} \right)$$
$$= \sum_{\bar{\imath} \in Y_{feasible}} c^{\bar{\imath}}_{a,t,k} + \sum_{\tilde{Y} \subset Y_{feasible}, |\tilde{Y}| \geq 2} d^{\bigcap_{\bar{\jmath} \in \tilde{Y}} \bar{\jmath}}_{a,t,k},$$

in which

$$d^{\bigcap_{\bar{\jmath} \in \tilde{Y}} \bar{\jmath}}_{a,t,k} := (-1)^{|\tilde{Y}|-1} \min_{\bar{\jmath} \in \tilde{Y}} \left(H'_{a,t,k} - \eta'^{\bar{\jmath}}_{a,t,k} \right),$$
$$\left|\tilde{Y}\right| := [\text{The number of elements in } \tilde{Y}].$$

Finally, because these rewards are essentially hazard reductions, we can define reward values for paths and path intersections as the sum of these overlaps over all arcs, stages, and IED types. We have:

$$c_{\bar{\imath}} = \sum_{a \in A} \sum_{t \in \{1..T\}} \sum_{k \in \{1..K\}} \left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k} \right), \forall \bar{\imath} \in Y,$$
$$d_{\bigcap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} = (-1)^{|\tilde{Y}|-1} \min_{\bar{\imath} \in \tilde{Y}} \left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k} \right), \forall \{\tilde{Y} | \tilde{Y} \subset Y, \left|\tilde{Y}\right| \geq 2\}.$$

From these expressions we note that the *c* coefficients can be viewed as special cases, for which $\left|\tilde{Y}\right| = 1$, of the more generally defined *d* coefficients. However, we leave the path coefficients ($c_{\bar{\imath}}$) and variables ($y_{\bar{\imath}}$) separated from the overlap coefficients ($d_{\bigcap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}}$) and variables ($z_{\bigcap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}}$) for the purposes of writing constraints in our mixed integer program. For correctness, we also note that a more accurate illustration of hazard overlaps is shown in Figure 4-11. This figure shows smaller hazard reductions as being completely contained within larger hazard reductions, which is consistent with our modeling. The derivation of the expressions above can be visualized in this Venn diagram.

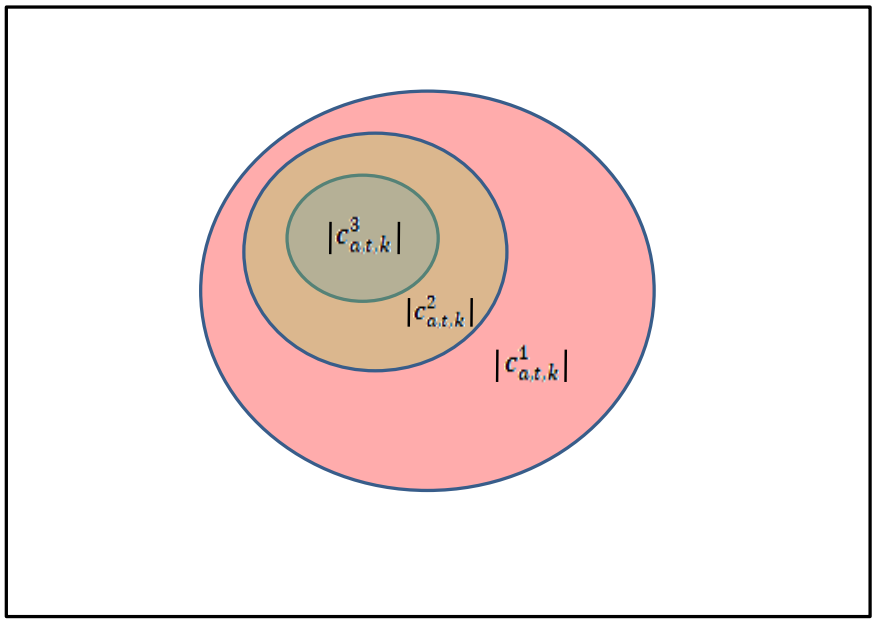

Figure 4-11: Hazard Overlap As Modeled

Introducing overlap variables and coefficients, we now re-write the problem:

maximize $\boldsymbol{c^T y + d^T z}$

subject to

$\boldsymbol{Ay \leq b}$, (1,2)

$\boldsymbol{By + Cz \leq \beta}$ (3,4,5)

$y_{\bar{\imath}} \in \{0,1\},\ \forall\, \bar{\imath} \in Y$ (6)

$z_{\bigcap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} \in \{0,1\},\ \forall \{\tilde{Y} | \tilde{Y} \subset Y, \left|\tilde{Y}\right| \geq 2\}$ (7)

Constraint sets (1,2) remains the same as their counterpart constraint sets (1) and (2) in the original mixed integer formulation. Constraint sets (3,4,5) have the purpose of setting overlap variables to 1, if all of its associated path variables are 1, i.e.,

$$z_{\bigcap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} = \begin{cases} 1, \text{if } y_{\bar{\imath}} = 1\ \forall\, \bar{\imath} \in \tilde{Y} \\ 0, \text{otherwise} \end{cases}.$$

We consider different constraints for two separate cases: first the case in which $\left|\tilde{Y}\right|$ is even and $d_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}} \leq 0$, and second the case in which $\left|\tilde{Y}\right|$ is odd and $d_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}} \geq 0$. In the first case, when the coefficient $d_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}}$ is non-positive, the associated decision variable $z_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}}$ will assume the value zero in the optimal solution if it is left unconstrained. We write the following constraints to "force" the variable to take value 1 if all its associated paths are in the feasible solution set, while otherwise allowing it to remain set at 0:

$$z_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}} \geq \sum_{\bar{\imath}\in\tilde{Y}} y_{\bar{\imath}} - \left|\tilde{Y}\right| + 1,$$
$$z_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}} \geq 0.$$

In the second case, in which the coefficient $d_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}}$ is non-negative, the associated decision variable $z_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}}$ will assume the value 1 in the optimal solution if it is left unconstrained. In this case, our constraint must "force" the variable to 0 as long as all of its associated overlapping paths are not included in the feasible solution set. We therefore write the constraints:

$$z_{\bigcap_{\bar{\imath}\in\tilde{Y}}\bar{\imath}} \leq y_{\bar{\imath}}, \forall \bar{\imath} \in \tilde{Y}.$$

Because in the mixed integer program the values for the z variables are bounded by the constraints given in both cases above, and will tend to seek those bounds in the optimum solution, we can remove the integrality constraints (7) on the z variables without changing the optimal solution to the problem. The integer constraints (6) on the y variables are still required and remain the same as constraint set (6) in the first mixed integer formulation.

The complete definition of the second mixed integer program formulation is:

$$\text{maximize} \sum_{\tilde{\imath} \in Y} c_{\tilde{\imath}} y_{\tilde{\imath}} + \sum_{\tilde{Y} \subset \mathrm{Y}, |\tilde{Y}| \geq 2} d_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} z_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}}$$

subject to

$$y_{\bar{\imath}} + y_{\bar{\jmath}} \leq 1, \ \forall \{(i,j) | \ \Omega'^{\bar{\imath}} = \Omega'^{\bar{\jmath}} = h, \left| t'^{\bar{\imath}}_0 - t'^{\bar{\jmath}}_0 \right| \leq T\_max_h + dwell_h\} \quad (1)$$

$$\sum_{\bar{\imath} \in Y_{h,d}} y_{\bar{\imath}} \leq Max\_day_h, \forall \ h \in \left\{1 \ldots \bar{h}\right\}, \forall d \in \{1, .., D\} \quad (2)$$

$$z_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} \geq \sum_{\bar{\imath} \in \tilde{Y}} y_{\bar{\imath}} - \left|\tilde{Y}\right| + 1, \forall \{\tilde{Y} | \tilde{Y} \subset \mathrm{Y}, \left|\tilde{Y}\right| \in \{2,4,6, \ldots\}\}, \quad (3)$$

$$z_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} \geq 0, \forall \{\tilde{Y} | \tilde{Y} \subset \mathrm{Y}, \left|\tilde{Y}\right| \in \{2,4,6, \ldots\}\}, \quad (4)$$

$$z_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} \leq y_{\bar{\imath}}, \forall \bar{\imath} \in \tilde{Y}, \forall \{\tilde{Y} | \tilde{Y} \subset \mathrm{Y}, \left|\tilde{Y}\right| \in \{1,3,5, \ldots\}\}. \quad (5)$$

$$y_{\bar{\imath}} \in \{0,1\}, \ \forall \ \bar{\imath} \in Y \quad (6)$$

In which:

$$c_{\bar{\imath}} = \sum_{a \in A} \sum_{t \in \{1..T\}} \sum_{k \in \{1..K\}} \left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k} \right), \forall \bar{\imath} \in Y,$$

$$d_{\cap_{\bar{\imath} \in \tilde{Y}} \bar{\imath}} = (-1)^{|\tilde{Y}|-1} \min_{\bar{\imath} \in \tilde{Y}} \left(H'_{a,t,k} - \eta'^{\bar{\imath}}_{a,t,k} \right), \forall \{\tilde{Y} | \tilde{Y} \subset Y, \left|\tilde{Y}\right| \geq 2\}.$$

The solution to this problem includes a feasible set of paths Y^* for which $y_{\bar{\imath}} = 1, \forall \bar{\imath} \in Y^*$. The optimal residual hazard values from this solution are easily obtained:

$$\gamma^*_{a,t,k} = \min_{\bar{\imath} \in Y^*} \left(\eta'^{\bar{\imath}}_{a,t,k} \right), \forall a \in A, t \in \{1..T\}, k \in \{1..K\}.$$

4.2.5.2.1 Preprocessing for the Second Mixed Integer Formulation

The "overlap" mixed integer formulation is a much smaller problem than the original mixed integer formulation, which we decomposed into constraint-generating sub-problems. The overlap formulation has no more than $2^{|Y|}$ variables, because this quantity represents the maximum number of existing subsets of Y. However, we can decrease this quantity substantially by not including overlaps existing between paths that cannot feasibly be executed in unison due to violations of constraint sets (1) and (2), because these paths cannot appear together in the optimal solution set. Overlap variables can also be omitted if there does not exist at least one common arc that is cleared in every mission path in the overlap set $\tilde{Y}$. Finally, we can exclude overlap variables if the clearances associated with the different paths in $\tilde{Y}$ are separated by

enough stages such that for each time stage at least one of the paths has a hazard reduction approximately equal to the input existing hazard value, i.e.

$$\forall a \in A, \forall t \in \{1..T\}, \forall \mathrm{k} \in \{1..K\}, \exists \bar{\imath} \in \tilde{Y} \text{ such that } {H'}_{a,t,k} - {\eta'}^{\bar{\imath}}_{a,t,k} \approx 0.$$

A user-defined tolerance can regulate how much of a difference between the existing hazard and each path's reduced hazard must exist on any arc, at any stage, for any IED type before an overlap variable is introduced into the problem. Small tolerances will result in more overlap variables and more preprocessing time, while larger tolerances will have the opposite effect.

Once an overlap variable is introduced, it requires either two constraints for overlaps involving even numbers of paths, or $\left|\tilde{Y}\right|$ constraints for overlaps involving odd numbers of paths. Identification of overlaps, computation of overlap coefficients, and formation of the constraints must all be completed in order to define and solve the mixed integer program. This preprocessing requires comparison of the reduced hazard values for all paths for each arc, stage, and IED-type combination, which can take a significant amount of time. A way to reduce the preprocessing time (other than increasing the tolerance) is to limit the overlaps considered to (feasible) path *pairs* only, making the assumption that the cumulative total of higher-order overlaps is not significant. For a set of paths in which all pair combinations are feasible, this approximation reduces the number of potential overlaps from a maximum of $2^{|Y|}$ to $\frac{|\tilde{Y}|(|\tilde{Y}|-1)}{2}$.

4.2.5.3 Path Optimization Control

The Path Optimization function returns the feasible subset of paths that minimizes residual hazard on the graph. As indicated in Figure 4-2, the function is used at two points in the algorithm, with a slightly different purpose in each case. We now outline the differences between the two uses of this function in our algorithm.

4.2.5.3.1 Path Optimization in the Path Set Production Sub-Process

The Path Optimization function's purpose within the Path Set Production sub-process is to limit the number of paths introduced into the problem during each iteration. The function inputs all of the paths generated by the Path Generation function in the current iteration for a single RCT. The number of such input paths is not constrained. The function outputs a feasible set of the

input paths that minimizes residual (conditional) hazard. The feasibility criteria, which come from the RCT's mission constraints, limit the number of paths outputted by the function. By selecting these output paths so as to minimize residual hazard, the function also ensures that they are good candidates for entry into the larger problem. The Path Generation function essentially serves as a filter, enabling the algorithm to achieve a good set of paths for each RCT during each iteration in the algorithm without having to retain a large set of generated paths.

The conditional hazard $\bar{\boldsymbol{H}}$, used as input to generate paths for the current RCT, also serves as the existing hazard input in the Path Optimization function. The reduced conditional hazard values $\bar{\boldsymbol{v}}^{\bar{\imath}}$ for each generated path ($\bar{\imath}$) constitute the reduced hazard inputs associated with each path, as described in paragraph 4.2.4.2. Using these inputs, the Path Optimization function finds the feasible combination of generated paths that results in the lowest residual conditional hazard, computed from the input existing hazard values $\bar{\boldsymbol{H}}$. This application of the Path Optimization function does not account for the original hazard values or the reduced original hazard values for each path. The optimal hazard outputs are used as the conditional hazard inputs into the next RCT's Path Generation.

4.2.5.3.2 Path Optimization in the Route Clearance Planning Algorithm

As opposed to its application in the Path Set Production sub-process, the Path Optimization function in the larger algorithm is not conditioned on any route clearance missions. It uses the original hazard values $\boldsymbol{H}$ as its existing hazard input and the reduced original hazard values $\boldsymbol{\eta}^{\bar{\imath}}$ as the associated path $\bar{\imath}$ reduced hazard inputs. This variation of the Path Optimization function inputs all paths included in the Path Set produced in the current iteration, as well as those from Path Sets produced in previous iterations. The function determines the feasible combination of these paths that results in the least residual hazard, as measured from the original hazard values. This combination of paths and the resulting residual hazard form this function's outputs.

4.2.6 Stopping Criteria

This step in the algorithm applies a simple test to the Path Optimization outputs to determine whether termination criteria have been met. These criteria can be based on the amount of improvement achieved in the current iteration (measured in hazard reduction from previous

iterations), the amount of computation time elapsed, the number of iterations, or a combination of these conditions. If the termination criteria are met, the process terminates with the optimal solution from the Path Optimization function as the best route clearance mission plan found. If the termination criteria are not met, the process uses the optimal residual hazard output from the Path Optimization function as the conditional hazard input into the next iteration of Path Set Production, and the algorithm continues.

4.3 Solution Process Summary

The functions described in section 4.2 are performed as depicted in Figure 4-2. The process is carried out as follows:

0. Determine the process inputs designated in Table 4-1 and execute the Original Hazard Computation function.
    a. Retrieve $\boldsymbol{H}$; $H_{a,t,k}, \forall a \in A, t \in \{1 \dots T\}, k \in \{1..K\}$.
    b. Set $\overline{\boldsymbol{H}} = \boldsymbol{H}$, i.e., $\overline{H}_{a,t,k} = H_{a,t,k}, \forall a \in A, t \in \{1 \dots T\}, k \in \{1..K\}$.
    c. Set $\bar{\imath} = 1$.
1. Execute Path Set Production.
   For each RCT $h \in \{1..\bar{h}\}$:
    a. Randomize the first start time. Uniformly pick a start time X on $(0, \psi]$ and determine which stage *t* this start time falls in. Set $\bar{t}_0^1$ equal to this stage, i.e., $\bar{t}_0^1 = [\mathrm{X}/\delta]$.
    b. Set $\bar{\jmath} = 1$.
    c. While $\bar{t}_0^{\bar{\jmath}} < T - \max[T\_max_h]$.
        i. Run the Path Generation function using the current RCT, $\bar{t}_0^{\bar{\jmath}}$, and conditional hazard values ($\overline{\boldsymbol{H}}$), along with the original hazard values ($\boldsymbol{H}$).
        ii. Set $\boldsymbol{\nu}^{\bar{\jmath}} = \boldsymbol{\xi}^*$, $\bar{\boldsymbol{\nu}}^{\bar{\jmath}} = \bar{\boldsymbol{\xi}}^*$, $\omega^{\bar{\jmath}} = h$, and $\varphi^{\bar{\jmath}} = \{\tilde{x}_0^*, \tilde{x}_{\tilde{t}_1}^*, \tilde{x}_{\tilde{t}_2}^* .. \tilde{x}_{\tau}^*\}$.
        iii. Re-set $\bar{\jmath} = \bar{\jmath} + 1$.
        iv. Set $\bar{t}^{\bar{\jmath}} = \bar{t}^{\bar{\jmath}-1} + \psi$.
    d. Set $\bar{J} = \bar{\jmath} - 1$.
    e. Run the Path Optimize function inputting $\overline{\boldsymbol{H}}$ as the existing hazard, $\bar{\boldsymbol{\nu}}^{\bar{\jmath}}$ $\forall \bar{\jmath} \in \{1..\bar{J}\}$ as the path reduced hazards, $\bar{t}_0^{\bar{\jmath}}$ $\forall \bar{\jmath} \in \{1..\bar{J}\}$ as the path start times, and $\omega^{\bar{\jmath}}$ $\forall \bar{\jmath} \in \{1..\bar{J}\}$ as the path RCTs.
    f. For all $\bar{\jmath}$ such that $x_{\bar{\jmath}} = 1$, i.e. for all paths included in the solution outputted by the Path Optimize function:
        i. Set $\boldsymbol{\eta}^{\bar{\imath}} = \boldsymbol{\nu}^{\bar{\jmath}}$.
        ii. Set $\Omega^{\bar{\imath}} = \omega^{\bar{\jmath}}$.

iii. Set $t_0^{\bar{\imath}} = \bar{t}_0^{\bar{\jmath}}$.

iv. Set $\Phi^{\bar{\imath}} = \varphi^{\bar{\jmath}}$

v. Re-set $\bar{\imath} = \bar{\imath} + 1$.

g. Set $\overline{\boldsymbol{H}} = \boldsymbol{\gamma}^*$, the optimal hazard output from the Path Optimization.

2. Run the Path Optimization function inputting $\boldsymbol{H}$ as the existing hazard, $\boldsymbol{\eta}^{\bar{\jmath}}\ \forall \bar{\jmath} \in \{1..\bar{\imath}-1\}$ as the path reduced hazards, $\bar{t}_0^{\bar{\jmath}}\ \forall \bar{\jmath} \in \{1..\bar{\imath}-1\}$ as the path start times, and $\omega^{\bar{\jmath}}\ \forall \bar{\jmath} \in \{1..\bar{\imath}-1\}$ as the path RCTs.
   a. Generate constraints that prevent incompatible missions and limit the number of missions per day for each RCT, as defined in paragraph 4.2.5.1.
   b. Determine overlap coefficients $d_{\bigcap_{\bar{\imath}\in\bar{Y}}\bar{\imath}}$, overlap variables $z_{\bigcap_{\bar{\imath}\in\bar{Y}}\bar{\imath}}$, and overlap constraints as defined in paragraph 4.2.5.2.
   c. Conduct mixed integer optimization as defined in paragraph 4.2.5.2.
   d. Output optimal solution vector $\boldsymbol{y}^*$ and optimal residual hazard values $\gamma_{a,t,k}^*\ \forall a \in A, t \in \{1 \dots T\}, k \in \{1..K\}$.
3. Review Stopping Criteria
   a. If Stopping Criteria are met, terminate with route clearance mission plans $\Phi^{\bar{\jmath}}$ for all $\bar{\jmath}$ such that $y_{\bar{\jmath}}^* = 1$.
   b. If Stopping Criteria are not met, set $\overline{\boldsymbol{H}} = \boldsymbol{\gamma}^*$, i.e., $\overline{H}_{a,t,k} = \gamma_{a,t,k}^*\ \forall a \in A, t \in \{1 \dots T\}, k \in \{1..K\}$, and return to step 1.

[This Page Intentionally Left Blank]

5 Experiments

In this chapter we describe the design, implementation, and results of three experiments with the goals of (1) providing insight into the route clearance planning problem and (2) demonstrating the applicability of the algorithm developed in Chapter 4. Experiment 1 seeks a correlation between the hazard reduction associated with a particular mission and the original hazard values on the graph during the stages corresponding to the mission. Experiment 2 examines the effect of partitioning the area of operations into unique sectors belonging to different RCTs versus planning for all RCTs to have access to the entire area of operations. Experiment 3 tests the value of having better foreknowledge of use rates on arcs. We use ©ILOG OPL Development Studio, Version 5.0 to execute the Path Optimization iterations in Experiments 2 and 3. All other processes and functions in the route clearance planning algorithm are executed and controlled using ©MATLAB 2007a. All experiments are carried out on a personal computer equipped with a 3.20 GHz processer, 1 GB of RAM, and a ©Microsoft Windows XP operating system.

5.1 Network Data and Unchanging Parameters

Some parameters in the model remain constant throughout all experimentation. These include the discrete time step increment (δ), which is set at 5 minutes, and the number of IEDs in the model (K), which we set at 2. Also, initial conditions are set at zero probability of IEDs of all types existing on each arc at the beginning of the planning horizon for each test. We carry out each test on one or both of two road networks in each experiment: an artificial network based on road connectivity in Cambridge, MA, and a network constructed using actual data on the roads in Utah.

5.1.1 Cambridge-Based Road Network

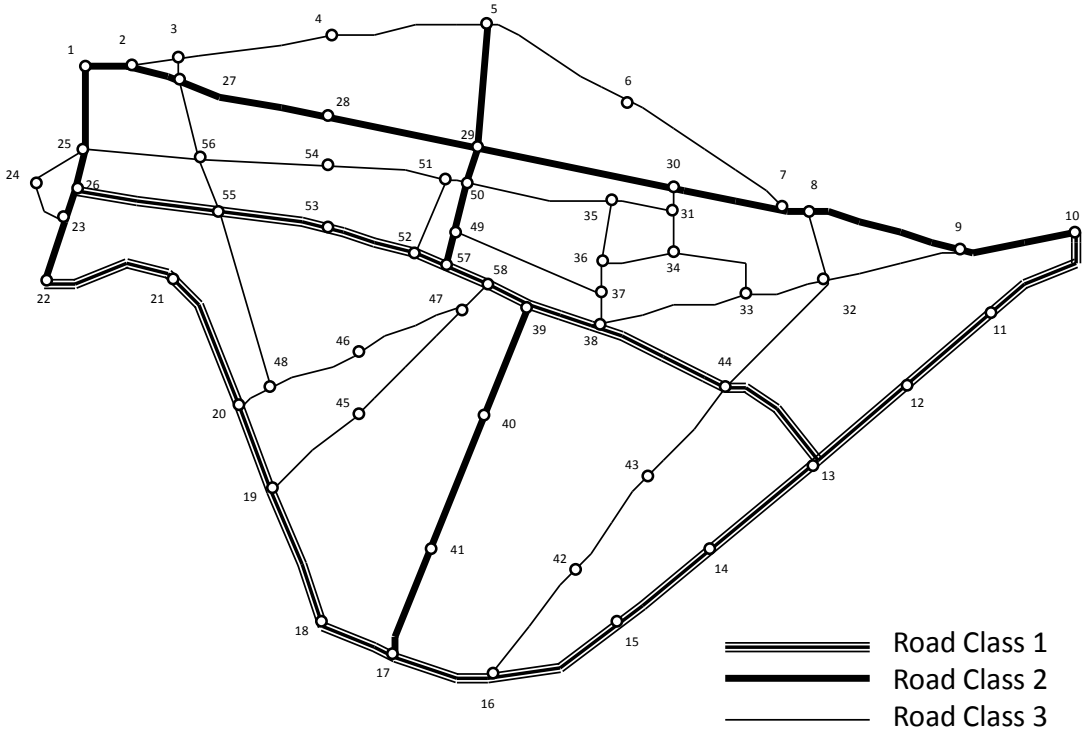

Figure 5-1: Cambridge-Based Road Network

The Cambridge-based road network consists of 58 nodes and 162 arcs, which are divided into three road-type classifications, as indicated. A graph representing this road network is shown in Figure 5-1, with each segment representing two directed arcs (one for each direction of travel). Lengths were arbitrarily estimated for each arc. RCT speeds in the Cambridge-based network are given in these length units per minute.

5.1.2 Utah Road Network

The Utah road network has 396 nodes and 918 arcs, divided into nine classifications. Each node in the network has associated latitude and longitude coordinates. The length for each arc is given in miles, and RCT speeds for this network are given in miles per minute. While it is not prudent

to display all nine road classifications, we give a graphical depiction of the Utah road network in Figure 5-2, with interstates plotted in red.

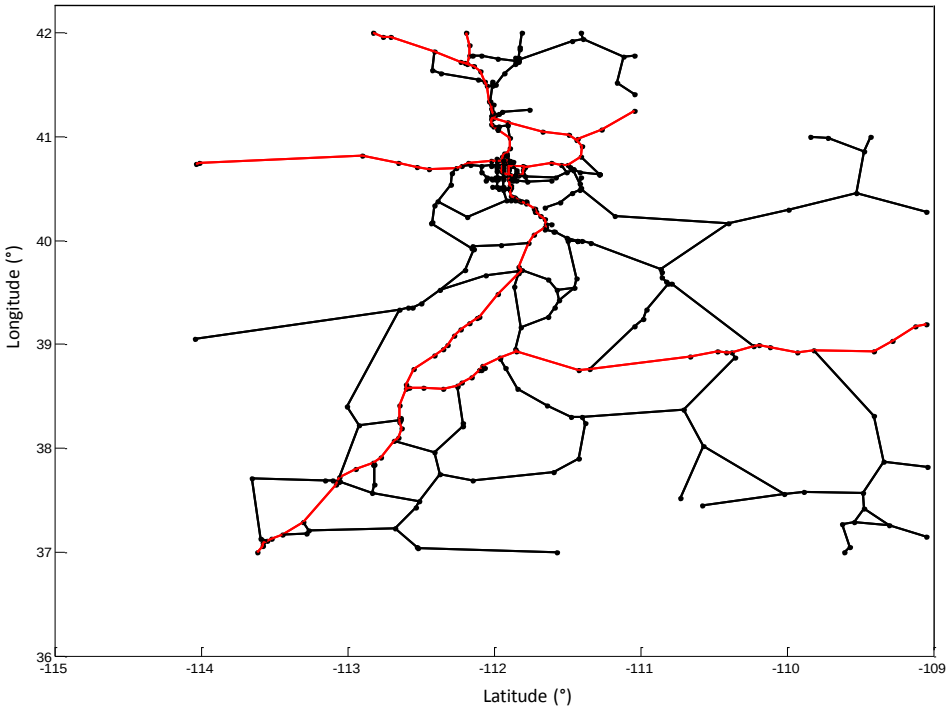

Figure 5-2: Utah Road Network

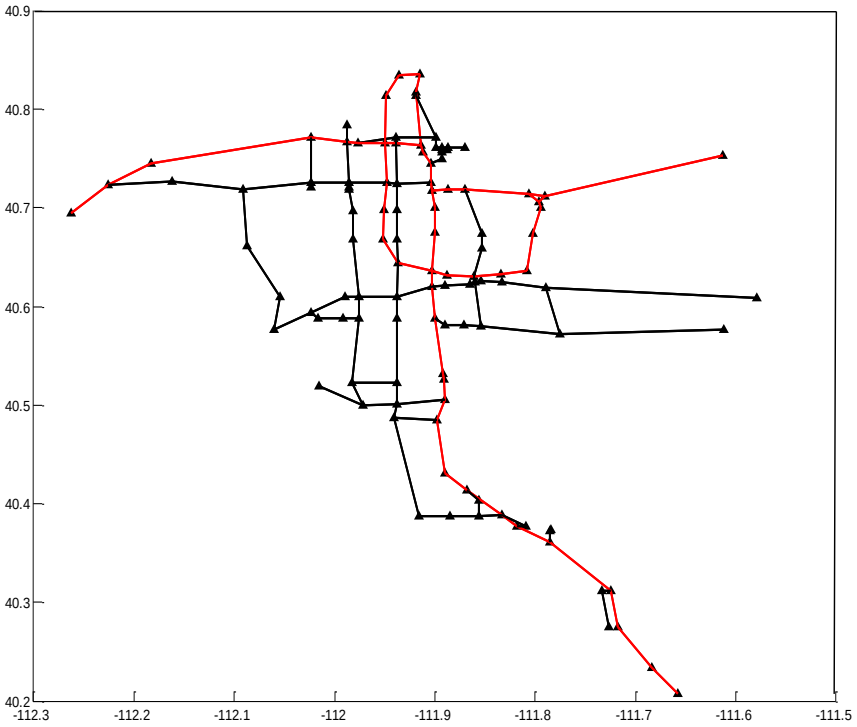

Figure 5-3: Salt Lake City Sub-Network

Unlike the Cambridge-based network, the Utah road network contains rural areas with few roads, as well as urban areas with higher road densities. In order to demonstrate the urban road density, Figure 5-3 shows only the Salt Lake City area within the Utah network.

5.2 Experiment 1

Our first experiment investigates the Path Generation function. We are specifically interested in how the total hazard reduction resulting from a generated mission path varies with the mission's start time, and whether the start times that result in the most hazard reduction can be predicted by analyzing the other inputs.

5.2.1 Conjecture 1A

We conjecture that the mission start time t_0 that results in the most conditional hazard reduction for a mission carried out by RCT h approximately coincides with the beginning of the period of T_max_h stages that has the maximum weighted conditional hazard sum. We weight the conditional hazard values by the detection capability of the RCT, so that hazard values relating to IEDs with a higher likelihood of detection contribute more to the sum. Essentially, we assert that the time period over which the RCT can have the greatest impact is the T_max_h-stage period over which the most "reducible" hazard exists on the subset of the graph that the RCT is capable of reaching.

The purpose of this experiment is to test whether the weighted hazard sum function provides any insight into good mission start times for Path Generation. If we can approximate the best mission start times for each RCT simply by examining the conditional hazard values, we might be able to produce a "good" mission path set using these start times without having to generate a large sample set of mission paths to select from, as discussed in chapter 4.

For this experiment, we define:

$A^h :=$ {The subset of arcs that comprise all feasible mission paths for RCT h},
$q(h) :=$ {The configuration type of RCT h},
$r(a) :=$ {The road classification of arc a},
$\overline{D}^d :=$ {The subset of stages that occur during day d within the planning horizon},

$\overline{H}_{a,t,k}$:= {Conditional hazard, as defined in Chapter 4}, and

$\bar{v}_{a,t,k}^{t_0}$:= {The Path Generation heuristic conditional reduced hazard output, given input RCT h starting at stage t_0, and input conditional hazard $\overline{H}_{a,t,k}$}.

We wish to test whether

$$\text{argmin}_{t_0 \in \{\overline{D}^d\}} \sum_{a \in A^h} \sum_{t \in \overline{D}^d} \sum_{k \in \{1 \ldots K\}} \bar{v}_{a,t,k}^{t_0,h} \approx$$

$$\text{argmax}_{s_0 \in \{\overline{D}^d\}} \sum_{a \in A^h} \sum_{t \in \{s_0 \ldots s_0 + T\_max_h\}} \sum_{k \in \{1 \ldots K\}} \overline{H}_{a,t,k} \times P\_detect_{q(h),k,r(a)}.$$

5.2.2 Conjecture 1B

We also use this experiment to compare the primary and alternate Path Generation heuristics. We hypothesize that the primary heuristic will perform worse than the alternate heuristic, but will take less time because of the more complete measure of reward.

5.2.3 Experiment 1 Design

We conduct a total of eight tests consisting of two trials each. For each test, we consider two RCTs (RCT A and RCT B) positioned at different base node locations (32 and 48, respectively) in the Cambridge-based road network. We execute the Path Generation heuristic starting at each stage in a one-day horizon, and compare the conditional hazard reductions to the weighted conditional hazard summed over the stages that comprise each mission. Table 5-1 shows the conditions for each of the six data sets, while the specific inputs for each data set can be found in Appendix D.

Test	Data Set	Trial A	Trial B	Special Conditions
1.1	11	RCT A	RCT B	Notional Data Set 1, no special conditions
1.2	12	RCT A	RCT B	Notional Data Set 2, no special conditions
1.2c	12	RCT A	RCT B	Data Set 2 with $\bar{\boldsymbol{H}} \neq \boldsymbol{H}$
1.2h2	12	RCT A	RCT B	Data Set 2, using alternate Path Generation Heuristic
1.3	13	RCT A	RCT B	Artificial Data Set 3, time-invariant use rates
1.4	14	RCT A	RCT B	Artificial Data Set 4, time-invariant IED emplacement rates
1.5	15	RCT A	RCT B	Artificial Data Set 5, time-invariant IED emplacement and use rates
1.6	16	RCT A	RCT B	Artificial Data Set 6, asynchronous IED emplacement activities and arc usage

Table 5-1: Design of Experiment 1

In this experiment, we do not carry out the route clearance planning algorithm as shown in Figure 4-2. Rather, these trials only involve numerous iterations of Path Generation. Tests 1.1 and 1.2 carry out Path Generation on notional data sets with no difference between conditional and original hazard values. Test 1.2c uses the same data set as test 1.2, but carries out Path Generation with conditional hazard values created by assuming some route clearance missions have already been conducted. Test 1.2h2 uses the same data set as test 1.2, but executes the alternate Path Generation heuristic. Tests 1.3, 1.4, 1.5, and 1.6 all use artificial data to provide more insight into the problem of determining good start times from the input data.

We use the following measurements to analyze and compare results for each trial:

$$\bar{W}^h(t_0) := \sum_{a\in A^h} \sum_{t\in\{t_0 \ldots t_0+T\_max_h\}} \sum_{k\in\{1\ldots K\}} \bar{H}_{a,t,k} \times P\_detect_{q(h),k,r(a)},$$

$$\bar{H}^t(t_0) := \sum_{a\in A^h} \sum_{k\in\{1\ldots K\}} \bar{H}_{a,t,k},$$

$$\bar{N}^h(t_0) := \sum_{a\in A^h} \sum_{t\in\bar{D}^d} \sum_{k\in\{1\ldots K\}} \left(\bar{H}_{a,t,k} - \bar{v}_{a,t,k}^{t_0,h}\right), \text{ and}$$

$$runtime := \{\text{The time elapsed in generating all 576 paths for each test}\}.$$

$\bar{W}^h(t_0)$ is the weighted conditional reduced hazard sum, as defined previously. $\bar{H}^t(t_0)$ is the *instantaneous* conditional hazard on the feasible sub-graph, summed over all arcs and IED types at each stage t_0. $\bar{N}^h(t_0)$ is the total reduction in conditional hazard resulting from the Path Generation for RCT h conducting a mission starting at stage t_0. Finally, we compare the runtimes in each case, with particular emphasis on comparisons between the different Path Generation heuristics.

Because we use the Cambridge-based road network, all arcs are within range of each RCT in each test, i.e., $A^h = A \; \forall h \in \{\text{A}, \text{B}\}$.

5.2.4 Experiment 1 Results and Analysis

We now discuss the results for each of the tests conducted in Experiment 1.

5.2.4.1 Test 1.1

Figures 5-4A and 5-4B show plots of $\overline{W}^h(t_0)$, $\overline{H}^t(t_0)$, and $\overline{N}^h(t_0)$, for trials A and B respectively, with some scaling for display purposes. The runtime for this test is 47.88 minutes.

We notice in both cases that the missions that give the most reduction in hazard occur about 25 stages prior to the peaks in both the instantaneous conditional hazard, $\overline{H}^t(t_0)$, and the weighted conditional hazard sum, $\overline{W}^h(t_0)$. These stages do not correspond to relatively high values in the weighted conditional hazard sum, but they do correspond to points in which the weighted conditional hazard summation function has a positive slope. We also note that the stages that maximize $\overline{W}^h(t_0)$ do not approximate the best mission start times. Finally, we observe the erratic nature of the path hazard reduction, $\overline{N}^h(t_0)$, coming from the Path Generation heuristic. Small changes in mission start times can cause significant fluctuation in the performance of the heuristic.

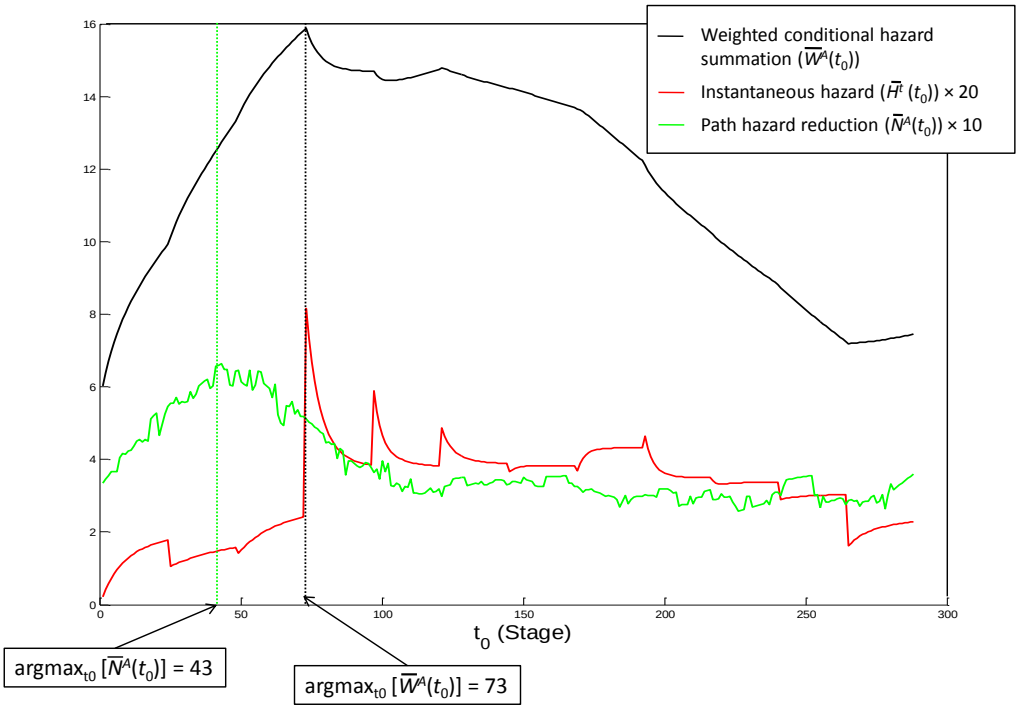

Figure 5-4A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.1).

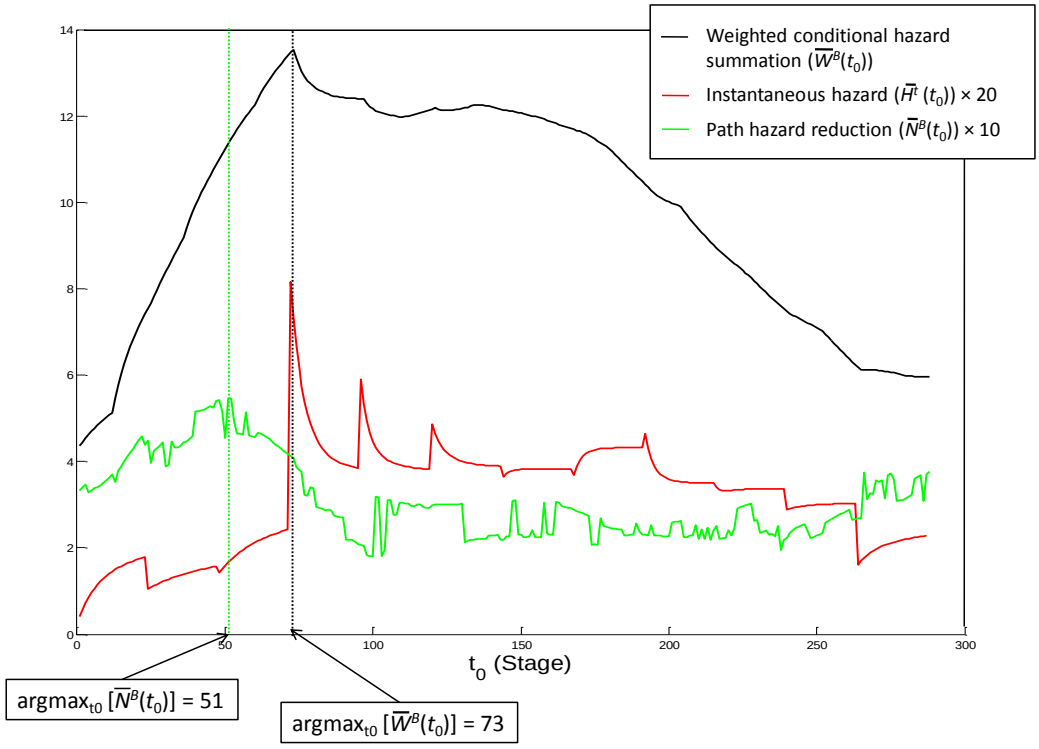

Figure 5-4B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.1).

5.2.4.2 Test 1.2

Figures 5-5A and 5-5B show plots of the output data for test 1.2.

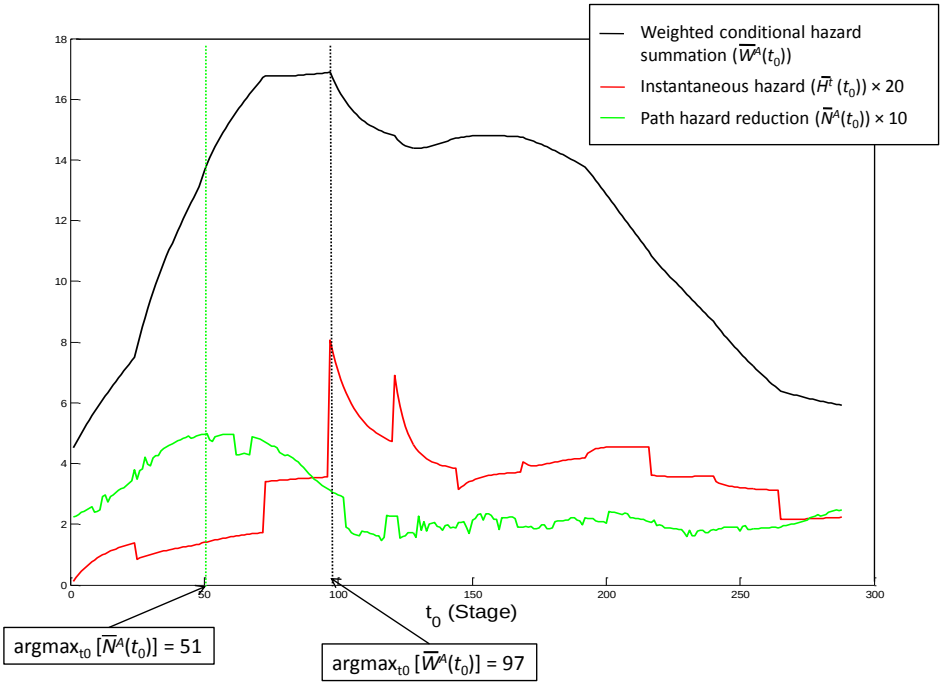

Figure 5-5A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.2).

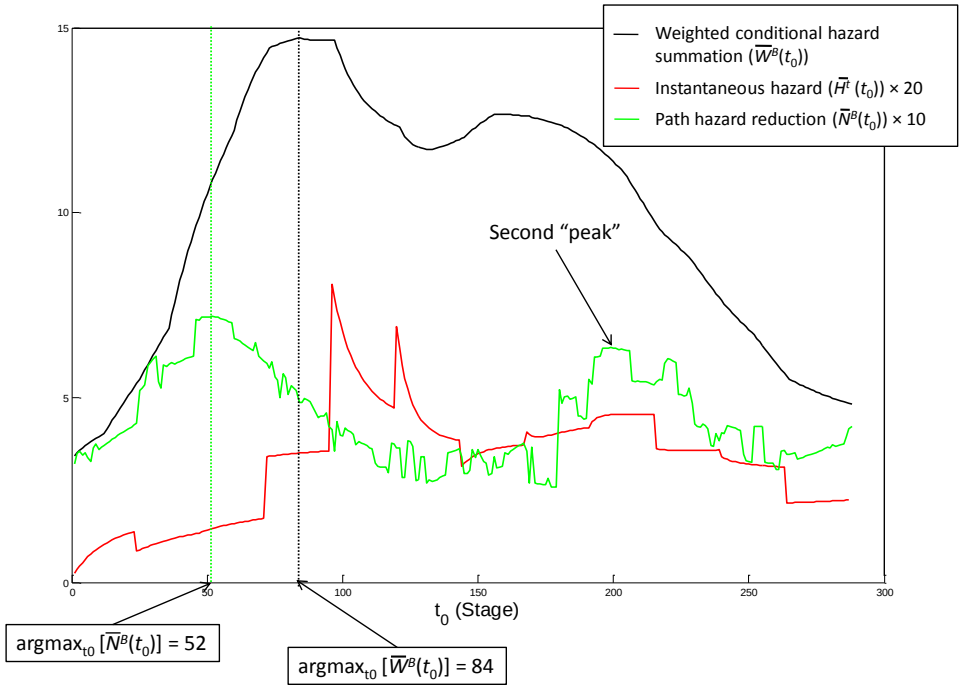

Figure 5-5B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.2).

The runtime for this test is 47.81 minutes.

These results have many of the same properties as the results from test 1.1. In this test, however, we observe even more erratic behavior, especially in the paths generated for RCT B. We also note that for RCT B, the best mission start time corresponds with a relatively low point on the weighted conditional hazard summation line. Finally, we note the second peak in the hazard reduction function for RCT B. This peak seems out of place, given the instantaneous conditional hazard and the weighted conditional hazard sum functions are both decreasing in the stages that correspond to this peak. In tests 1.3-1.6, we aim to investigate more closely the conditions that result in larger hazard reductions.

5.2.4.3 Test 1.2c

This is the only test for which we use different original hazard and conditional hazard values in the Path Generation heuristic. The missions chosen to generate these conditional hazard values were the best mission paths generated (resulting in the most hazard reduction) for each RCT in test 1.2. These missions began at stage 51, for RCT A, and stage 52, for RCT B. Both missions were applied to arrive at the conditional hazard input for this test. The results from this test 1.2c are plotted in Figures 5.6A and 5.6B.

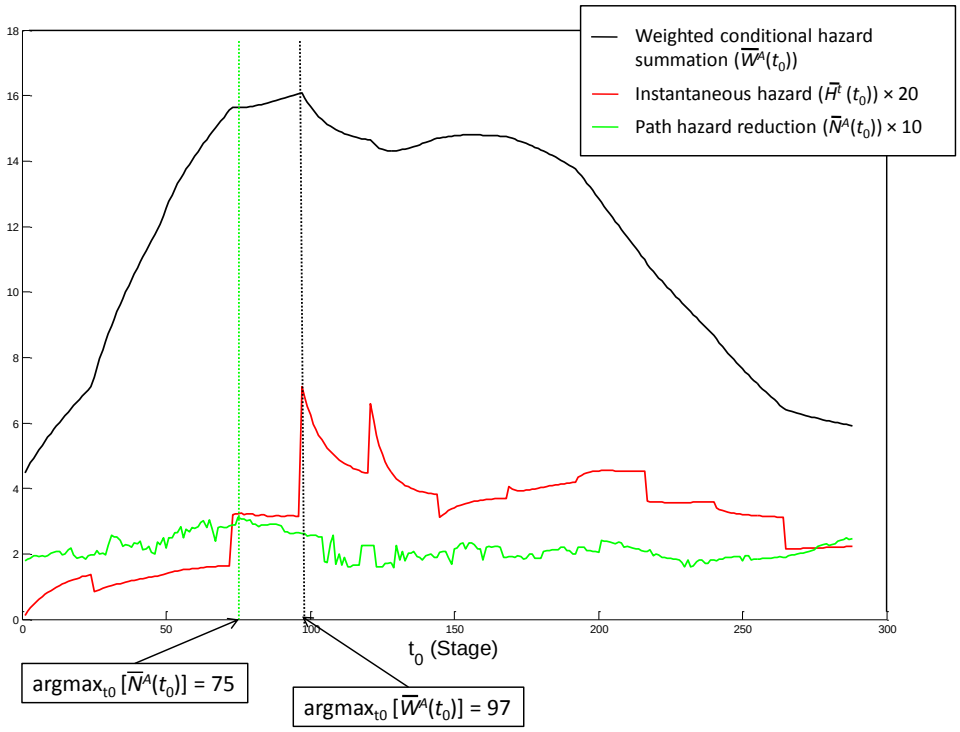

Figure 5-6A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.2c).

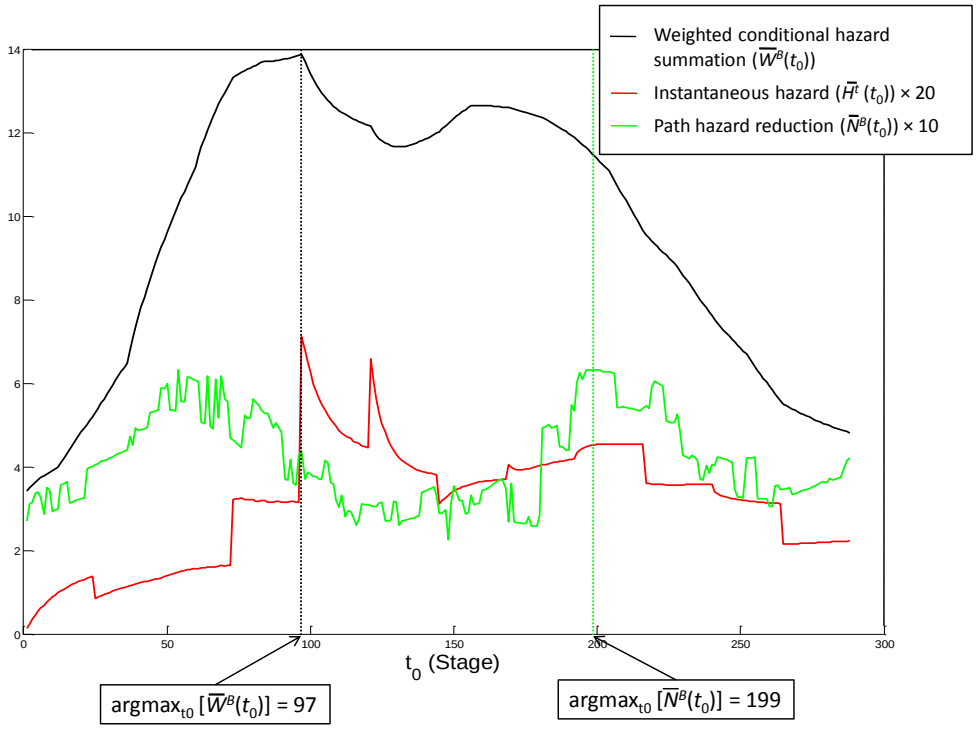

Figure 5-6B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.2c).

The runtime for generating these paths, including the time spent computing conditional hazard values, is 48.00 minutes.

RCT A's hazard reductions have almost flattened out when plotted for different start times due to this conditional hazard input. The case of RCT B is more interesting. The Path Generation heuristic returns paths with relatively high hazard reductions at approximately the same stages as the two missions that have been selected and used to determine conditional hazard values. This output indicates that there are multiple mission paths in the network for RCT B with a start time on or about the 50th stage that result in significant hazard reductions. Furthermore, the second peak in RCT B's reduced hazard function now exceeds the height of the first, with the mission beginning at stage 199 resulting in the most hazard reduction. Discouragingly, the shapes of the weighted conditional hazard summation function and the instantaneous conditional hazard function have hardly changed from their original shapes in test 1.2, and appear to tell us little about the change in the best mission start times.

5.2.4.4 Test 1.2h2

Figures 5-7A and 5-7 B show plots of the results of test 1.2h2. We introduce the notation $\overline{N}^{h}_{(x)}(t_0)$, in which x refers to the heuristic used in obtaining the mission path and resulting reduction in conditional hazard. The values $\overline{N}^{h}_{(1)}(t_0)$ (hazard reduction using the primary Path Generation heuristic) are the same as the hazard reductions in test 1.2, and were not computed again for this test. The alternate heuristic in this test computed arc clearance hazard reductions ahead from the current stage to stage $\bar{t}$ for each label-setting iteration, where

$$\bar{t} = \min[t_0 + T\_max_h + dwell_h, T].$$

The runtime for this test is 699.72 minutes. This time only includes running the alternate Path Generation heuristic to generate each of the 576 paths for this test.

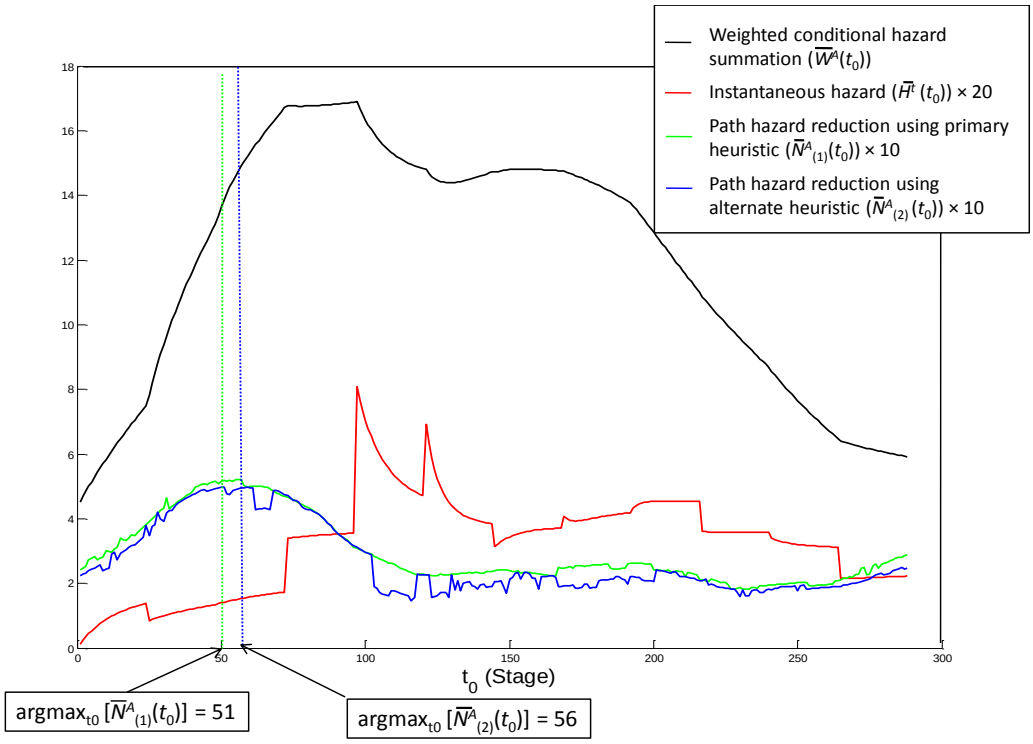

Figure 5-7A: Comparison of Primary and Alternate Path Generation Heuristics for RCT A (Test 1.2h).

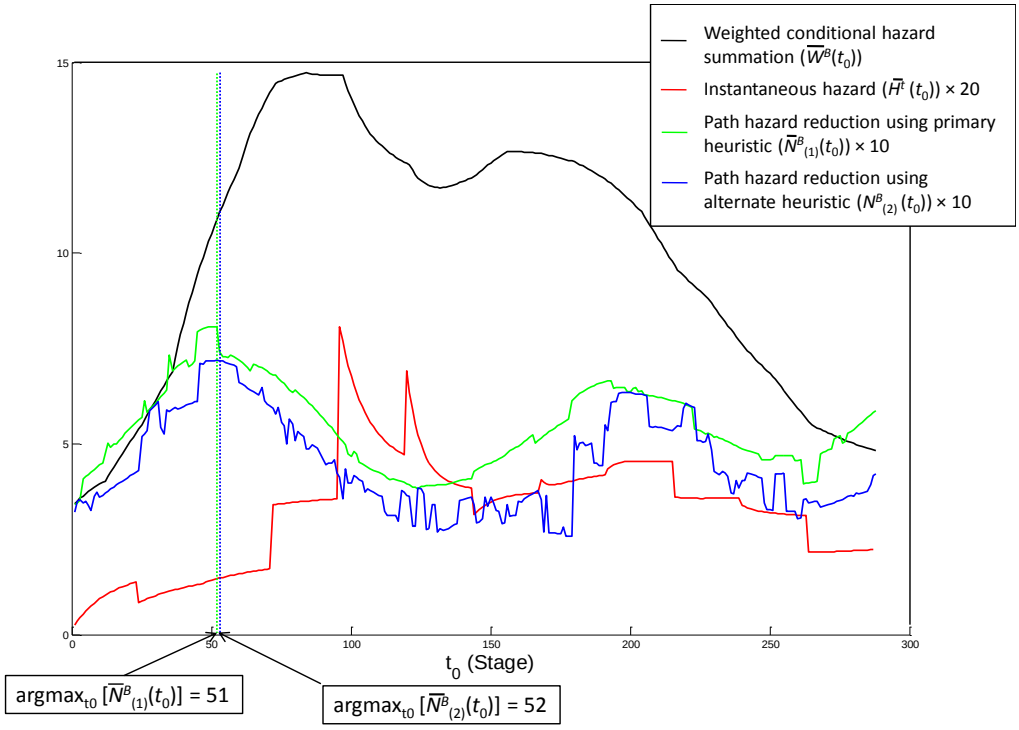

Figure 5-7B: Comparison of Primary and Alternate Path Generation Heuristics for RCT B (Test 1.2h).

We find that the alternate Path Generation heuristic (2) performs better than the primary heuristic in almost all cases. There are some exceptions, but in these situations the outcomes from the two heuristics are approximately the same. The alternate heuristic also appears to be more consistent than the primary heuristic, generating a smoother curve. Both heuristics follow roughly the same contour; their minimum and maximum hazard reductions occur with approximately the same mission start times. The alternate heuristic is much less computationally efficient than the primary heuristic, most likely due to the many iterative cost computations it must compute for each label-setting operation. The amount of time it takes to generate paths using the alternate heuristic is too long for efficient use in the route clearance planning algorithm.

5.2.4.5 Test 1.3

In this test we do not allow use rates to vary with time, while emplacement rates do vary. The results of this test are depicted in Figures 5-8A and 5-8B.

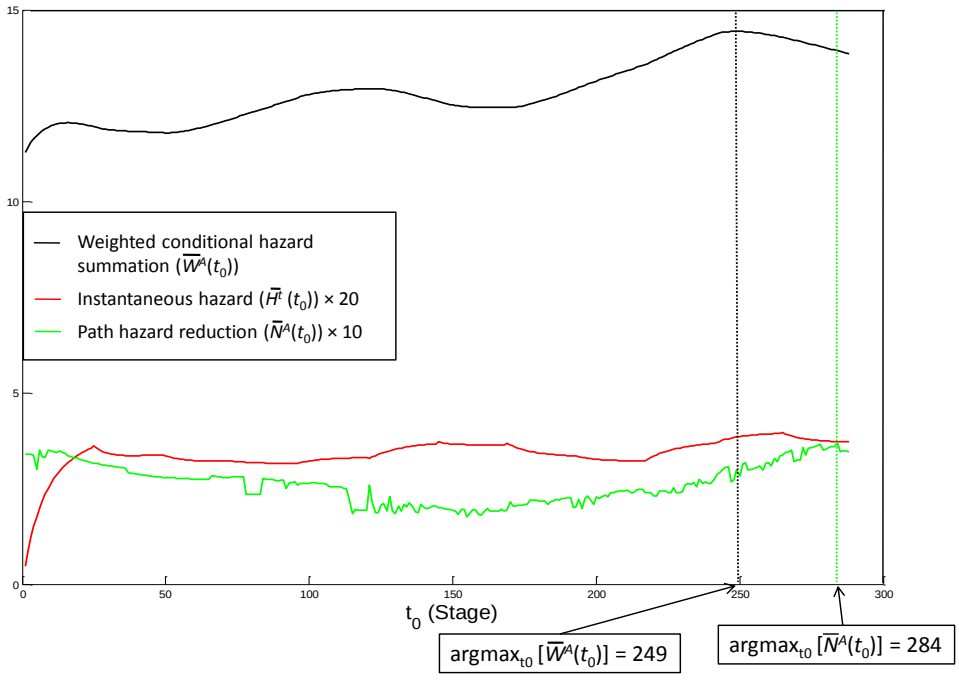

Figure 5-8A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.3).

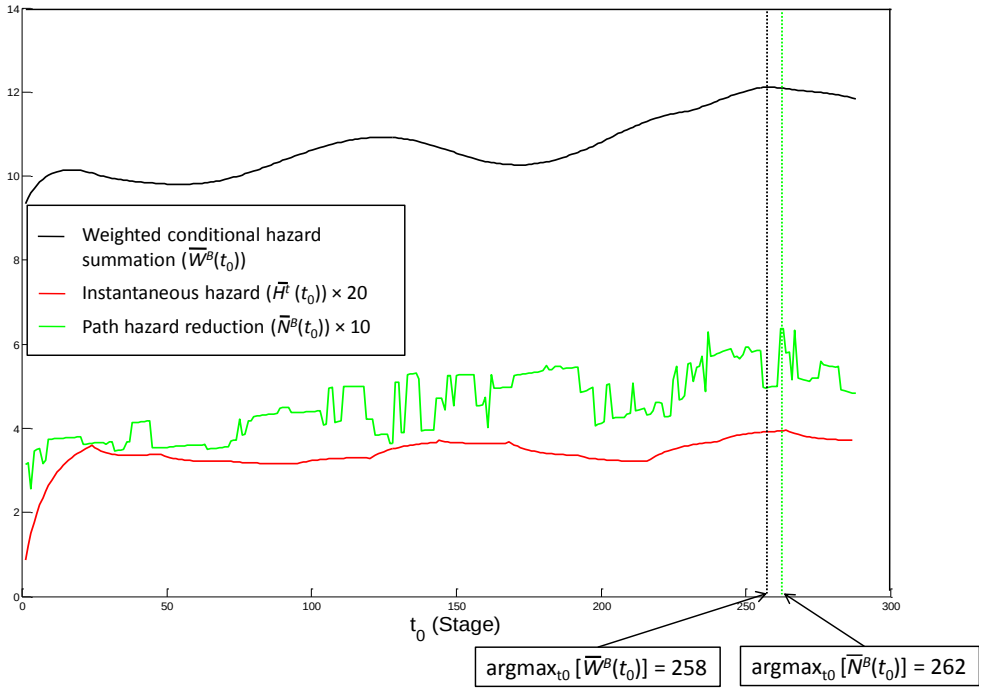

Figure 5-8B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.3).

The runtime for this test is 47.86 minutes.

We see that when use rates are held constant, instantaneous conditional hazard and weighted conditional hazard summation functions are smooth curves, even though the emplacement rates are step functions. For RCT A, we observe that the best mission paths begin in the early stages or the late stages of the day, with the missions resulting in the least conditional hazard reductions occurring in the middle of the day. This result is not immediately evident in the instantaneous conditional hazard curve or the weighted conditional hazard summation curve. For RCT B, there is a slight upward trend in the hazard reduction as a function of mission start time, which appears to emulate a similar upward trend in the weighted conditional hazard summation function. In this test we observe that the best mission start time approximately coincides with the weighted conditional hazard summation maximum.

5.2.4.6 Test 1.4

In this test, emplacement rates are held constant while use rates vary with time. The results are plotted in Figures 5-9A and 5-9B. The runtime for this test is 47.87 minutes.

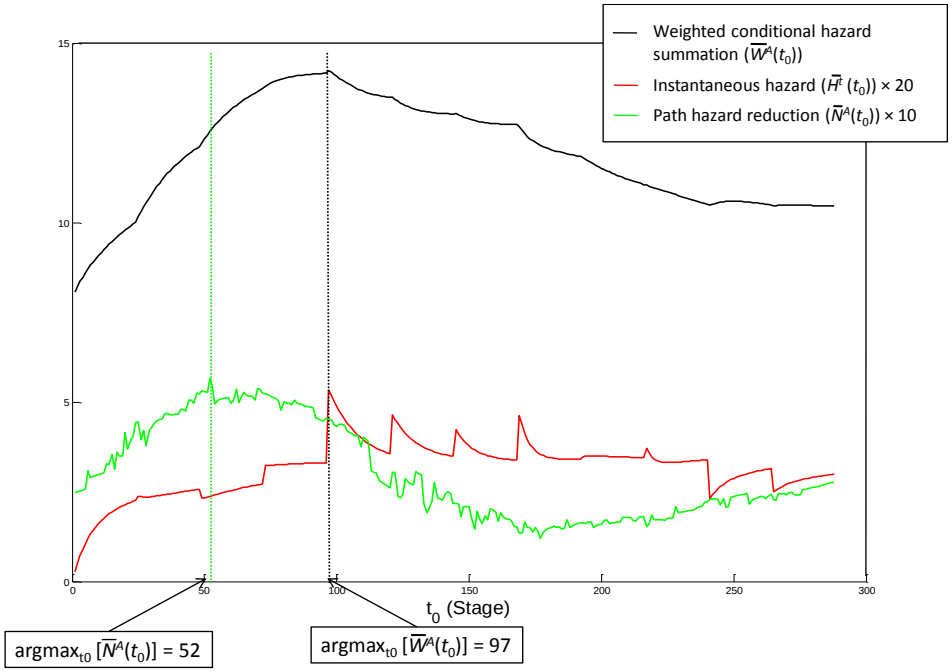

Figure 5-9A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.4).

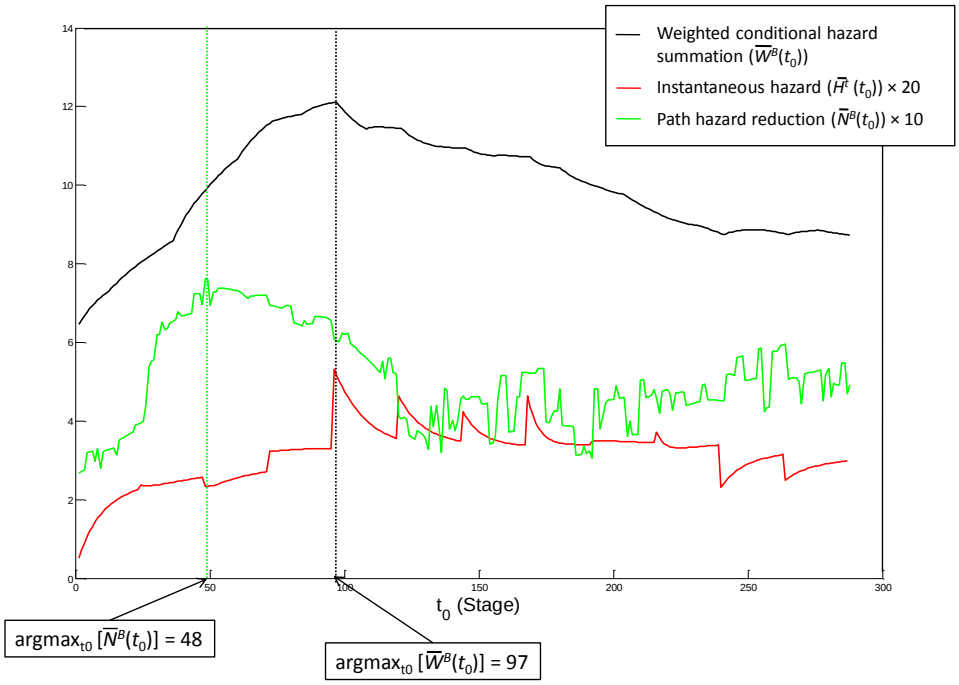

Figure 5-9B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.4).

Again we see the erratic nature of the primary Path Generation heuristic. For each RCT in this test, the best mission paths generated begin about 50 stages prior to the maxima in the instantaneous conditional hazard function and in the weighted conditional hazard summation function. The instantaneous conditional hazard function is no longer a smooth curve, with steps corresponding to increases or decreases in use rates. These steps appear because use rate is an input into the hazard function. Recall from Chapter 4 that:

$$\Xi(a,t,k) = -\log(1-\theta_{a,t,k}), \text{ in which}$$

$$\theta_{a,t,k} = \rho_{a,t,k} \times P\_use_{a,t} \times P\_eff_k.$$

While the probability of IED emplacement affects the evolution of the probability of an IED existing on an arc ($\rho_{a,t,k}$ for a type *k* IED on arc *a* at stage *t*), fluctuations in the use probabilities have a much more direct and immediate impact on hazard values, as shown in the figures.

5.2.4.7 Test 1.5

The results of this test are depicted in Figures 5-10A and 5-10B. For this test, both use rates and emplacement rates are held constant with time. The runtime for this test is 47.91 minutes.

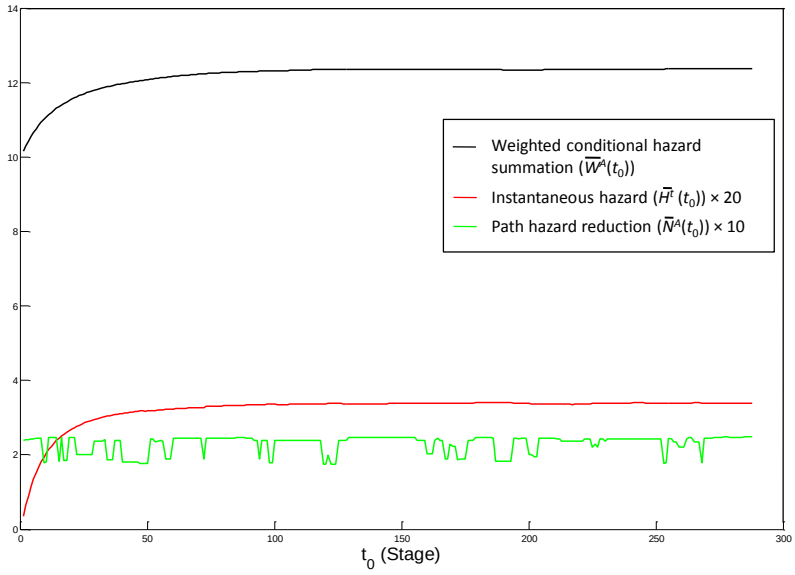

Figure 5-10A: Plots of Path Hazard Reduction and Weighted Hazard Summation for RCT A (Test 1.5).

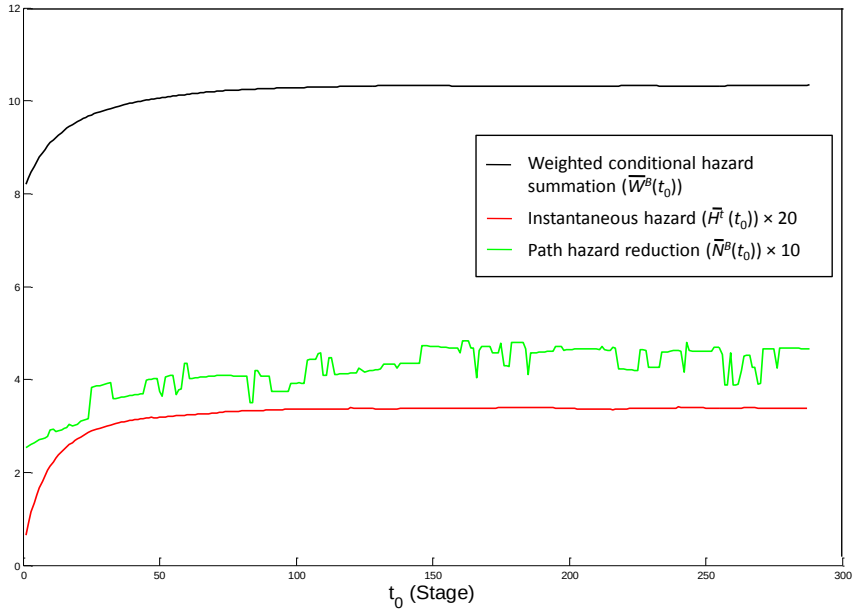

Figure 5-10B: Plots of Path Hazard Reduction and Weighted Hazard Summation for RCT B (Test 1.5).

This test is interesting because we see that conditional hazard (which is the original hazard in all Experiment 1 tests except test 1.2c) increases from its initial condition, which is zero, toward some steady state condition. For some arcs the evolution to steady state is faster than for others based on the emplacement and use rates assigned to each arc. Even as the conditional hazard approaches steady state, we still observe some erratic behavior in the hazard reduction output from the (primary) Path Generation heuristic. The fluctuations in hazard reduction over periods in which there appears to be very little change in conditional hazard indicates the sensitivity of the heuristic. It is possible that a small change in hazard might cause a good path to be pruned early, in favor of another path that ultimately results in less hazard reduction. There is naturally no information on optimal mission start times in the instantaneous conditional hazard and weighted conditional hazard summation information for this test.

5.2.4.8 Test 1.6

Test 1.6 uses inputs that reflect a spike in arc usage early in the day, followed some time later by an IED emplacement spike late in the day. Otherwise, activity is held at constant, low levels. Figures 5-11A and 5-11B depict the results of this test.

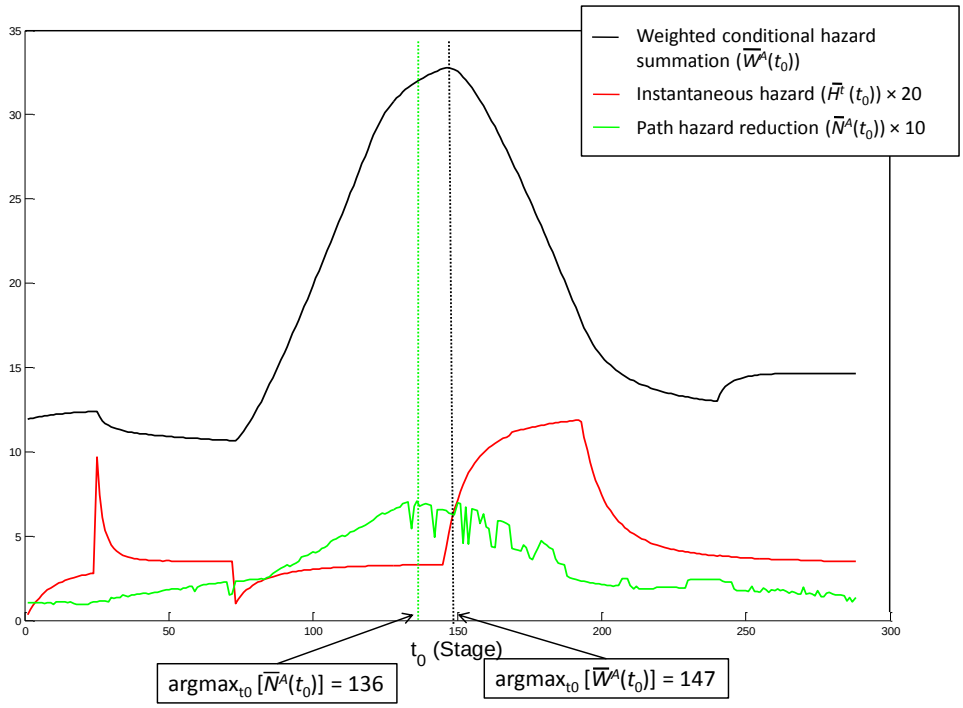

Figure 5-11A: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT A (Test 1.6).

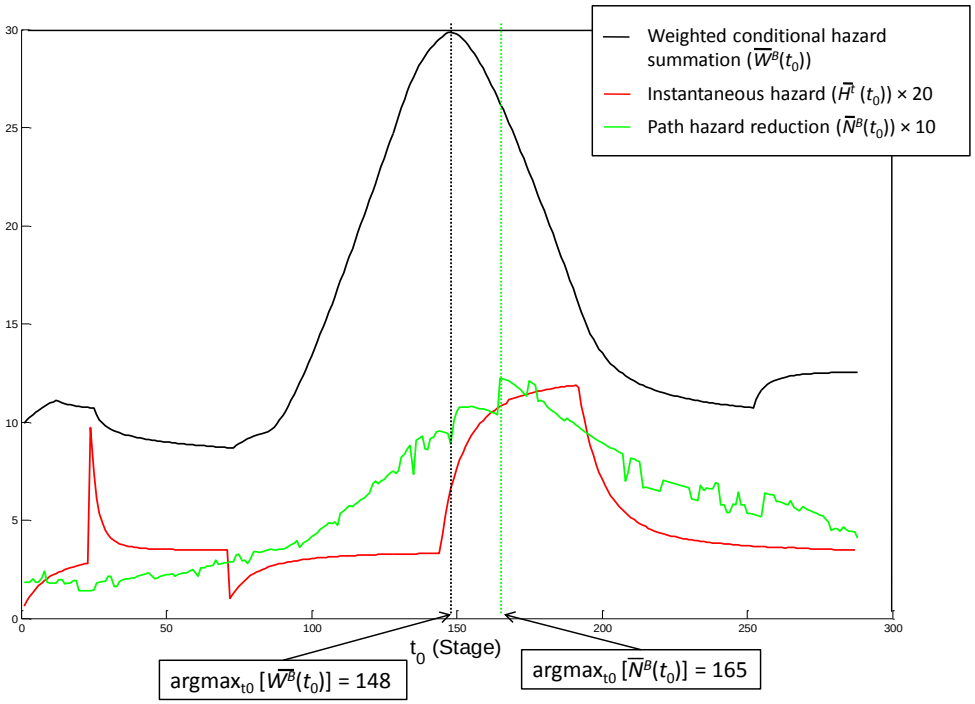

Figure 5-11B: Results Showing Best Mission Start Time and Weighted Hazard Summation Maximum for RCT B (Test 1.6).

The runtime for this test is 47.81 minutes.

We can see in the instantaneous conditional hazard function the early step increase corresponding to the increased use on the arcs. Because the state probabilities corresponding to IEDs in place ($\rho_{a,t,k}$) have not built up to high levels prior to this increase in arc usage, the magnitude of this step increase is limited. We also notice that the high usage rate effects a fast decay in conditional hazard because any emplaced IEDs would quickly be expended during periods of such increased usage. The second increase in conditional hazard occurs later in the day, beginning at stage 144. This increase is more gradual because it corresponds to higher IED emplacement rates. These higher rates gradually increase the "IED in place" state probabilities ($\rho_{a,t,k}$) on each arc, resulting in the gradual increase in conditional hazard. Because use rates are lower during these stages, this increase in conditional hazard decays more slowly than it did in the earlier case once the emplacement rates return to their original values.

We find that the best route clearance missions in this case correspond with the period of increased IED emplacement rates. The figures indicate the differences between the two RCTs.

RCT A has a longer maximum mission duration, and we observe that its best missions begin earlier than those of RCT B. However, RCT B is more effective than RCT A, which is apparent in the greater reductions in conditional hazard values resulting from RCT B's missions.

5.2.5 Experiment 1 Summary and Conclusions

We have not found enough evidence to support Conjecture 1A (mission start times that result in the most hazard reduction could be approximated from the weighted conditional hazard summation). Looking at the data sets in Appendix D and the conditional hazard reductions for each test, we find that the best times to conduct route clearance missions are (1) immediately before periods of increased arc usage, and (2) during periods of increased emplacement rates. Unfortunately, test 1.6 is not applicable in supporting this claim because the period of increased arc usage is so early in the day that it is not possible for either RCT to complete a mission prior to it. From other tests, however, it is evident that missions that occur over stages that fall during or after high-emplacement periods, and before high-use periods, are the most effective. Intuitively, this rule makes sense: RCTs must detect IEDs after they are emplaced but before they are employed in order to be effective.

Conjecture 1B (the alternate Path Generation heuristic returns better paths than the primary heuristic, measured by reduction in hazard, but requires more time to run) is supported by the results, although we find that the alternate Path Generation heuristic performs slightly worse than the primary heuristic in a very small number of cases. In the large majority of cases, however, we find that the alternate heuristic outperforms the primary heuristic, resulting in an average 17% increase in performance (measured in total hazard reduction) from the primary heuristic. As expected, we also find that the alternate heuristic requires considerably more time to execute than the primary heuristic.

5.3 Experiment 2

In this experiment we investigate the effects of partitioning the road network into disjoint subsets of arcs assigned to different RCTs.

5.3.1 Conjecture 2A

In this experiment, we consider the value in partitioning the graph into sub-graphs assigned to each RCT. We hypothesize that by partitioning the area of operations, we will arrive at solutions faster, but they will result in less overall hazard reduction than the solutions obtained without partitioning the graph. The projected increase in computation speed comes from two factors: (1) the Path Generation heuristic is run on a subset of the original graph, requiring fewer computations and label-setting operations, and (2) the partition will significantly reduce the number of overlapping paths, resulting in a decrease in the amount of preprocessing effort required in the Path Optimization function. Essentially, partitioning has the effect of decomposing the problem into smaller sub-problems.

5.3.2 Conjecture 2B

In addition to examining the effects of partitioning, we also study the effects of different control methods applied within the Path Set Production sub-process. Specifically, we conjecture that by selecting only a limited number of paths generated for each RCT in the Path Set Production sub-process, and by reconditioning the (conditional) hazard values on these paths before generating paths for the next RCT, we can arrive at a solution in less time with only marginal loss in overall performance. We select generated paths for inclusion in the path set based on some criteria that look for maximal conditional hazard reduction (to ensure the algorithm performs well) and minimal overlap (to decrease computational effort in the Path Optimization function).

5.3.3 Experiment 2 Design

We conduct four tests, which are each run on two data sets (Data Sets 21 and 22), for a total of eight trials. The conditions for the four tests are depicted in Table 5-2. The data sets and each of these conditions are explained in more detail in the subsequent paragraphs.

	Partitioned Area of Operations	Combined Area of Operations
Unselective Path Set Production	Test 2.1	Test 2.2
Selective Path Set Production	Test 2.3	Test 2.4

Table 5-2: Design of Experiment 2

5.3.3.1 Data Sets and Parameters

For data sets 21 and 22, we use the Cambridge-based road network and position two RCTs of identical capability at node 32. All other data, such as data pertaining to IED emplacement rates and arc use rates, is exactly the same as in data sets 11 and 12, respectively. The specific information pertaining to these data sets is in Appendix D.

In this experiment we carry out the route clearance planning algorithm over a two-day planning horizon. Within the Path Set Production sub-process, paths are generated with start times at one hour increments, with a randomized initial start time, as shown in step 1 in the Path Set Production algorithm presented in section 4.2.4.2. Once these paths have been generated for each RCT, additional paths are generated by selecting the start time that results in the most hazard reduction for each day (from the set of mission paths already generated), and running the Path Generation heuristic using start times at 15 minute increments over the 4-hour interval centered on each of these selected start times. (This step is an augmentation to the control algorithm in Section 4.2.4.2, occurring immediately following step 1.c.) For each RCT, we therefore expect a total of approximately 40 paths generated for each day in the planning horizon, in every iteration of Path Set Production.

The stopping criteria used in this experiment are: less than 0.1% improvement in hazard reduction from the last iteration, elapsed computation time of more than two hours, or three iterations completed.

5.3.3.2 Partitioning the Area of Operations

Figure 5-12 shows the Cambridge-based road network partitioning. For the cases in which we partition the area of operations, we assign one RCT to clear only the arcs associated with the red

links in the graph, while the other RCT clears only the arcs associated with the blue links. The Path Generation heuristic in each case builds and sets labels on a mission network consisting only of arcs in each respective RCT's partition. In the trials in which the area of operations is not partitioned, the entire graph is considered when generating mission paths for each RCT.

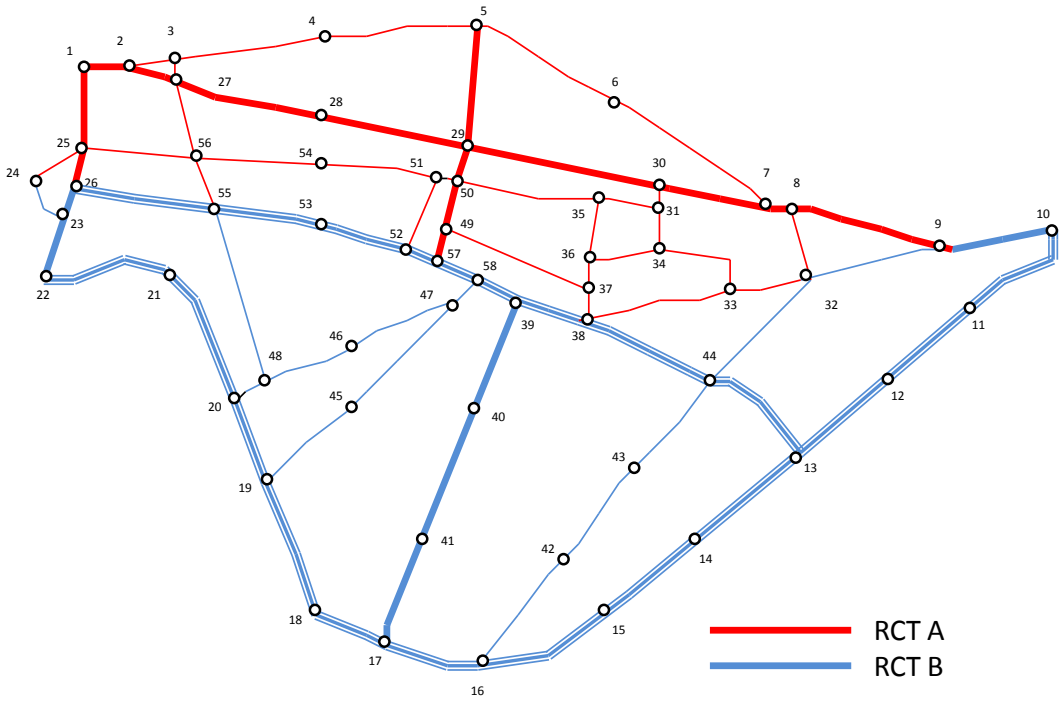

Figure 5-12: Cambridge-Based Road Network Partitioning

5.3.3.3 Path Set Production Controls

5.3.3.3.1 Unselective Path Set Production

For the trials in which we do not use a selection procedure when producing the Path Set, we execute the modified route clearance planning algorithm depicted in Figure 5-13. All mission paths generated for each RCT are simply included in the path set. Additionally, conditional hazard is not recomputed after paths are generated for each RCT. From the Path Set Production control algorithm given in section 4.2.4.2, these changes translate to omitting steps d, e, and g,

and applying the actions in step f to all generated paths. Aside from these changes, the route clearance planning algorithm remains the same as it was presented in Chapter 4.

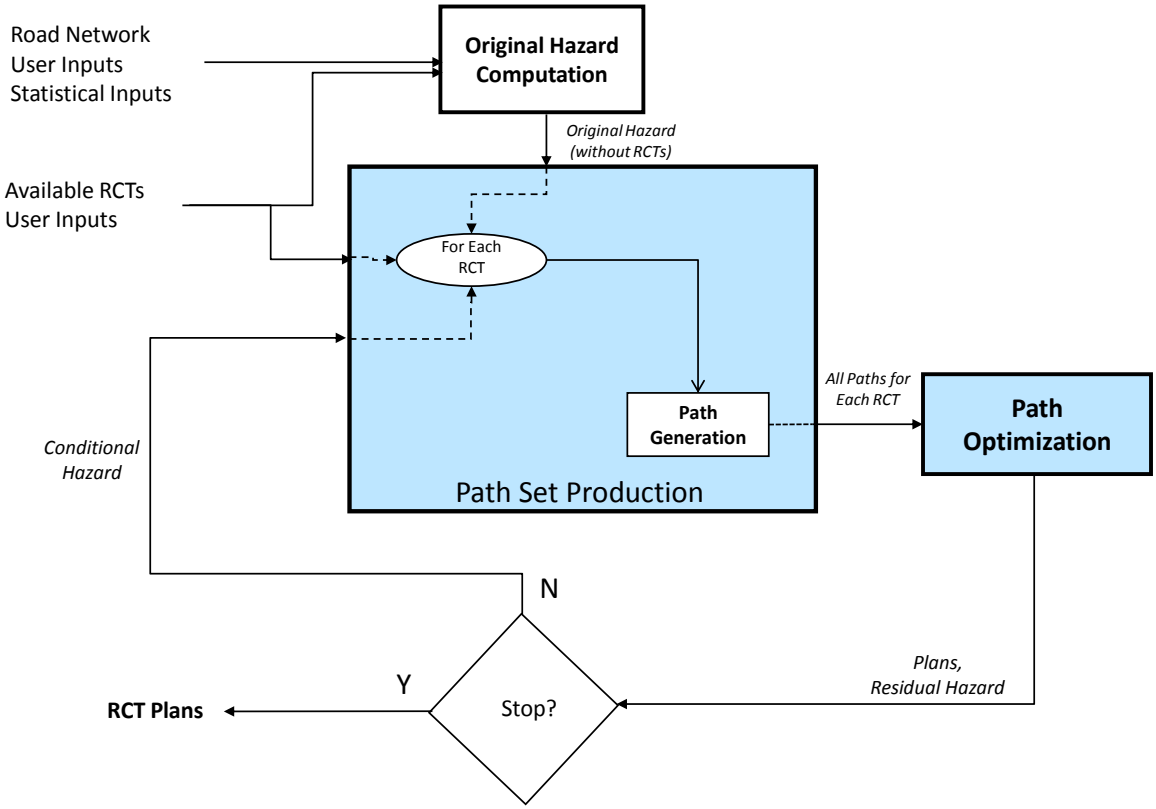

Figure 5-13: Route Clearance Planning Algorithm With Unselective Path Set Production

5.3.3.3.2 Selective Path Set Production

For the trials using selective Path Set Production controls, we use the algorithm presented in Chapter 4, depicted in Figure 4-2, with the Path Set Production control algorithm as it is presented in section 4.2.4.2. We use the Path Optimization function to identify a combination of paths for each RCT to meet the requirements set forth in Conjecture 2B: (1) relatively large reductions in conditional hazard and (2) minimal overlaps. The paths in the solution to the Path Optimization function have relatively few overlapping effects because they are constrained to be separated by enough time to satisfy feasibility constraints. Because the effects of route clearance on an arc decay quickly (as depicted in Figure 4-8), missions that are separated in time by the

feasibility constraints are much less likely to have significant overlapping hazard reductions than missions occurring simultaneously. The Path Optimization function also chooses paths to maximize reduction in conditional hazard (i.e., to minimize conditional reduced hazard), so the paths in the solution are also going to meet the second criterion relating to mission effectiveness.

Finally, we recondition the conditional hazard values to include the paths outputted by the Path Optimization function before generating paths for the next RCT (step g in the Path Set Production control algorithm in section 4.2.4.2). This step decreases the rewards associated with the arc clearances that comprise the output paths, causing subsequent iterations of the Path Generation heuristic (for other RCTs) to find mission paths that do not involve the same arc clearances. Not only does this conditional hazard update result in fewer overlaps, it also ensures a diversity of paths entering the Path Set in each iteration.

5.3.3.4 Metrics for Analysis and Comparison

We consider the following metrics for each trial:

$\#Paths^{(I)}$:={The total number of mission paths in the *I*th algorithm iteration, including the artificial path corresponding to no route clearance introduced in the first iteration},

$\Gamma^{(I)}$:= {The total hazard reduction resulting from the *I*th algorithm iteration's solution}

$runtime^{(I)}$:={The computation time elapsed in running all iterations up to and including iteration *I*}.

$\Gamma_{\%}^{(I)} := \frac{\Gamma^{(I)}}{\sum_{a\in A}\sum_{t\in\{1..T\}}\sum_{k\in\{1..K\}} H_{a,t,k}}$ ={The percent hazard reduction (from original hazard) resulting from the *I*th algorithm iteration's solution}.

5.3.4 Experiment 2 Results

Table 5-3 summarizes the initial results for Experiment 2. Most trials did not complete three iterations because some stopping criterion was met in a previous iteration.

		Data Set 21			Data Set 22		
Test	Variable	Iteration 1	Iteration 2	Iteration 3	Iteration 1	Iteration 2	Iteration 3
2.1	$\#Paths^{(I)}$	154	301	(Stop)	157	311	(Stop)
	$\Gamma^{(I)}$	4.1420	4.1420		5.2343	5.2507	
	$\Gamma_{\%}^{(I)}$	4.11%	4.11%		5.19%	5.21%	
	$runtime^{(I)}$	59.1 min	310.2 min		59.0 min	288.4 min	
2.2	$\#Paths^{(I)}$	(Intractable)			(Intractable)		
	$\Gamma^{(I)}$						
	$\Gamma_{\%}^{(I)}$						
	$runtime^{(I)}$						
2.3	$\#Paths^{(I)}$	9	17	(Stop)	9	17	(Stop)
	$\Gamma^{(I)}$	4.1359	4.1400		5.1633	5.1633	
	$\Gamma_{\%}^{(I)}$	4.11%	4.11%		5.12%	5.12%	
	$runtime^{(I)}$	10.9 min	20.9 min		10.1 min	19.3 min	
2.4	$\#Paths^{(I)}$	9	17	(Stop)	9	17	(Stop)
	$\Gamma^{(I)}$	4.5084	4.5084		5.5140	5.5140	
	$\Gamma_{\%}^{(I)}$	4.48%	4.48%		5.47%	5.47%	
	$runtime^{(I)}$	21.1 min	41.6 min		20.0 min	39.6 min	

Table 5-3: Experiment 2 Primary Results

We examine each conjecture separately.

5.3.4.1 Effects of Partitioning the Area of Operations

When running the route clearance planning algorithm without the selective Path Set Production, we find that partitioning the area of operations makes the problem tractable. The two trials in test 2.2, in which the graph was not partitioned and there was no path selection procedure, could not complete the overlap preprocessing in the Path Optimization function in less than eight hours of processing time. Given the identical conditions for the two RCTs in this problem, the Path Set Production sub-process created nearly identical sets of paths for each RCT in each iteration. These paths had too many overlapping effects for the algorithm to identify and process efficiently.

With the selective Path Set Production sub-process the algorithm finds solutions for both the partitioned and combined tests (2.3 and 2.4). We see that each iteration in test 2.4 (combined

area of operations) takes roughly twice the amount of computation time as its counterpart in test 2.3 (partitioned area of operations), even though the same number of paths is generated in each case. We also observe that this improvement in efficiency comes at a cost of about 7% of the total hazard reduction in each case.

5.3.4.2 Effects of Including Path Selection Controls in Path Set Production

For the partitioned trials (tests 2.1 and 2.3), we find that applying the selection controls to the Path Set Production sub-process significantly reduces computation time by limiting the number of new paths introduced into the algorithm in each iteration. For both data sets, it took test 2.1 (no path selection controls) about 15 times longer to complete two iterations than it took for test 2.3 (path selection controls). When using Data Set 21, the overall hazard reduction resulting from the two tests are nearly identical. For the trials using Data Set 22, we find that the solution in test 2.3 (with the selection controls) yields about 98% of the hazard reduction achieved in test 2.1 (without the selection controls).

Without partitioning we find the problem is intractable when the number of paths introduced into the algorithm in each iteration is not limited. In addition to providing this limiting function, the selection controls also update the conditional hazard after selecting paths for each RCT, decreasing the number of overlapping paths likely to emerge in the resulting Path Set. Controlling the number of overlapping paths going into the Path Optimization function significantly improves the efficiency of the algorithm.

5.3.5 Experiment 2 Follow-On Tests

In order to obtain results for test 2.2, we reduce the overlap preprocessing effort required by identifying and assigning variables to path *pair* overlaps only, as suggested in paragraph 4.2.5.2.1.

5.3.5.1 Conjecture 2C

We estimate that by identifying pair overlaps only, we will increase algorithm speed without significant loss of performance (i.e., hazard reduction).

5.3.5.2 Tests 2.1p and 2.2p Design

We define tests 2.1p and 2.2p as the re-execution of tests 2.1 and 2.2, respectively, substituting the "pair-only" overlaps approximation for the full overlaps method applied in the original tests.

5.3.5.3 Tests 2.1p and 2.2p Results

		Data Set 21			Data Set 22		
Test	Variable	Iteration 1	Iteration 2	Iteration 3	Iteration 1	Iteration 2	Iteration 3
2.1p	$\#Paths^{(I)}$	154	301	(Stop)	157	311	(Stop)
	$\Gamma^{(I)}$	4.142	4.142		5.2343	5.2507	
	$\Gamma_{\%}^{(I)}$	4.11%	4.11%		5.19%	5.21%	
	$runtime^{(I)}$	13.1 min	77.28 min		11.7 min	54.6min	
2.2p	$\#Paths^{(I)}$	156	(Stop)	(Stop)	156	(Stop)	(Stop)
	$\Gamma^{(I)}$	4.4682			5.3515		
	$\Gamma_{\%}^{(I)}$	4.44%			5.31%		
	$runtime^{(I)}$	401.8 min			295.3 min		

Table 5-4: Experiment 2 Subsequent Results

Table 5-4 summarizes the results of tests 2.1p and 2.2p. We find that the total hazard reductions for tests 2.1p and 2.1 are the same. However, computation time is significantly decreased, with each iteration in test 2.1p taking approximately a quarter of the computational time required in test 2.1. The "pairs-only" overlaps approximation increased the algorithm's efficiency with no loss in performance in this case. We also observe that the algorithm completes one iteration for test 2.2p before terminating due to the amount of runtime elapsed. The solutions achieved in each single-iteration trial are comparable with the results obtained after two iterations in test 2.4 (no partitioning with path selection controls). However, the trials in test 2.4 required much less computational effort to complete two iterations than the single-iteration trials in test 2.2p.

5.3.6 Experiment 2 Summary and Conclusions

In Experiment 2, we examined several methods of running the algorithm to maximize both performance (in terms of original hazard reduction) and efficiency. Each method examined focused on reducing overlap preprocessing time in the Path Optimization function, either by

controlling the numbers of overlaps in the Path Sets produced, or by omitting most of the overlap computations altogether. The test results support Conjectures 2A, 2B, and 2C.

Partitioning the graph has the effect of decreasing the performance of the algorithm, but also decreases computation time. We expect this result because the partition is only an additional constraint that reduces the size of the set of all feasible solutions. This result has an analogy in military operations, in which the terrain is divided and subdivided into areas of operations for different units. If the senior commander could personally, optimally allocate every asset within his control to accomplish the mission, the partitioning of the terrain would not be necessary. However, the number of military assets makes this impossible for large military units. The partition serves as a decomposition of the problem into smaller, more manageable components. These components might be further decomposed by subordinate commanders following a similar procedure.

The results also indicate that limiting the number of paths introduced into the route clearance planning algorithm in each iteration by applying good selection criteria (maximize conditional hazard reduction while minimizing arc clearance overlap) considerably increases computational efficiency with a small cost in performance. In these trials, applying the path selection controls reduced computation time by over 90% while performing to within 2% of the solution obtained without the controls. However, most of the computation time in this algorithm is spent generating paths that are ultimately not introduced into the Path Set for optimization.

We also found that computing only the overlap coefficients for feasible pair-combinations of overlapping paths in the problem provides a very good approximation for the Path Optimization function to obtain optimal or near-optimal solutions. By restricting the number of potential overlaps considered in the preprocessing step, this approximation significantly reduces the computation effort required in the algorithm, making otherwise intractable problems solvable.

Finally, we note two additional observations. First, the algorithm is relatively consistent for these two data sets, yielding similar results in each case. Secondly, stopping criteria are met after two iterations in all tests (excluding tests 2.2 and 2.2p) because the second iteration yields only marginal improvement from the first iteration's solution. This final result indicates that the algorithm is most efficient in the first iteration, with subsequent iterations requiring similar (additional) processing times but resulting in less improvement to the solution. Under time-

constrained planning conditions, running the algorithm for one iteration can result in a solution nearly as good as one obtained from multiple iterations of the algorithm.

5.4 Experiment 3

In Experiment 3, we test the value of having access to future patrol and convoy schedules in addition to statistical arc use probabilities when conducting route clearance planning.

5.4.1 Conjecture 3A

In this experiment, we explore the value of having definitive information on arc usage, such as a schedule of all planned convoys and military movements, when planning route clearance missions. We presume that better foreknowledge of friendly operations will result in more hazard reduction when conducting optimization-based route clearance planning.

5.4.2 Conjecture 3B

Our second conjecture for this experiment is that, given accurate information about all military movements scheduled to occur in the planning horizon, the route clearance planning algorithm will perform considerably better (estimate more than 17% observed in Experiment 1) when using the alternate Path Generation heuristic instead of the primary heuristic. This conjecture is based on the alternate heuristic's consideration of the enduring effects of route clearance, which we estimate will be more important in cases in which more information is available.

5.4.3 Experiment 3 Design

We conduct four tests in this experiment, each over a seven day planning horizon. In these tests, we run the route clearance planning algorithm on two sets of data with different amounts of information about future movement plans. Table 5-5 summarizes the conditions for each test.

Test	Arc Usage Information	Path Generation Heuristic
Test 3.1	No deterministic information (Probabilistic only)	Primary
Test 3.2	Partial information	Primary
Test 3.3	Perfect information	Primary
Test 3.3h2	Perfect information	Alternate

Table 5-5: Design of Experiment 3

5.4.3.1 Data Sets and Parameters

For this experiment, we conduct each test on two basic data sets (Data Sets 31 and 32). Data Set 32 uses the partitioned Cambridge-based road network, IED emplacement rates, and route clearance team parameters from Data Set 2 in Experiment 2 (see Figure 5-12). Data Set 31 uses the Utah-based road network with sixteen RCTs. This road network is partitioned into twelve mutually exclusive areas of operation, each having one or two assigned RCTs. Additionally, two of the sixteen RCTs are assigned to clear interstate highways only, but are not confined to a single area of operations. Figure 5-2 depicts the Utah road network with arcs representing interstates indicated in red. Figure 5-3 shows the Salt Lake City sub-network, again with interstates indicated by red arcs, which serves as one of the twelve areas of operation in this experiment. Finally, we make each RCT unavailable for one of the seven days in the planning horizon in both data sets. Appendix D offers more details on each of these data sets.

We use the route clearance planning algorithm depicted in Figure 4-2 and described in Chapter 4. In the Path Set Production sub-process, paths are generated using start times spaced at 3-hour increments, beginning with a random initial start time as described in Section 4.2.4.2. Once all of these paths are generated, we generate additional paths at 45-min increments over the 3-hour interval centered on the start time corresponding to the best (most hazard reducing) path for each day. This augmentation to the Path Set Production controls, similar to the one described in Paragraph 5.3.3.1, results in the addition of about four generated paths for each day. From this method of Path Generation we expect approximately 12 paths generated for each available day in the planning horizon, for each RCT. Other than this minor augmentation, the Path Set

Production sub-process follows the algorithm given in Section 4.2.4.2, using the Path Optimization function to select a promising subset of generated paths and updated conditional hazard values iteratively for each RCT.

Justified by the results from Experiment 2, we limit overlap preprocessing to pair overlaps only in the Path Set Production sub-process. Full overlap preprocessing is still executed in the Path Optimization step in the route clearance planning algorithm.

The stopping criteria used in Experiment 3 are the stopping criteria used in Experiment 2 without the computation time limit.

5.4.3.2 Test 3.1: No Deterministic Information

In test 3.1 we use probabilistic arc usage data based on statistically determined rates and a Poisson approximation, as described in Chapter 4. This test is essentially the control case, in which we assume we have no knowledge of future convoys beyond what can be inferred from statistical information.

5.4.3.3 Test 3.2: Partial Information

For test 3.2, we produce some information on scheduled convoys over the seven-day planning horizon, allowing us to determine when certain arcs will be traversed. We assume that this information is not complete; convoys of which we have no foreknowledge will still occur at rates we can approximate statistically. To include both probabilistic convoy movements and known convoy plans, we generate arc use probabilities based on the statistical information and then update arc usage values 1 for stages that correspond to times that convoys are scheduled to traverse a particular arc. Use probability thus takes the following form:

$$P\_use_{a,t} = \max\left[1 - e^{-\mu_{a,t}\delta}, \Phi_{a,t}\right],$$

$$\text{for } \Phi_{a,t} := \begin{cases} 1, \text{if a scheduled convoy traverses arc } a \text{ at stage } t \\ 0, \text{otherwise.} \end{cases}$$

5.4.3.4 Test 3.3: Perfect Information

Test 3.3 uses perfect convoy information, meaning there are no convoys or movements that we do not know about in advance, to set the arc use probabilities. In this case we again generate a

convoy schedule and use it to determine when each arc will be traversed. We then set use probabilities to 1 for stages on arcs that correspond to planned convoy traversal times. All other use probabilities are set to $0 + \varepsilon$, because there are no other movements to consider when planning, i.e.,

$$P\_use_{a,t} = \Phi_{a,t},$$

$$\text{for } \Phi_{a,t} := \begin{cases} 1, \text{if a scheduled convoy traverses arc } a \text{ at stage } t \\ 0 + \varepsilon, \text{otherwise.} \end{cases}$$

We set ε to be arbitrarily small. We cannot allow usage probabilities to reach zero because the recursions for hazard computation have $P\_use_{a,t}$ in the denominator (see Section 4.2.3). Note that $P\_use_{a,t}$ does not depend on the statistical usage rate from the underlying data set in the perfect information case. See Appendix E for a description of the method used to generate convoy plans for the partial and perfect information scenarios.

5.4.3.5 Test 3.3h2: Perfect Information Using the Alternate Heuristic

Test 3.3h2 uses the perfect information used in test 3.3 and runs the route clearance planning algorithm using the alternate Path Generation Heuristic to generate paths for inclusion in the Path Set only. This version of the algorithm uses the primary Path Generation heuristic, because of its speed, to generate paths for selection in the Path Set Production sub-process. Instead of retaining the selected paths outputted by the Path Optimization function, this method *regenerates* the selected paths using the same set of inputs (start time, conditional hazard, etc.) in the alternate heuristic. This method is justified by the findings from Experiment 1, in which we observed the similarity in contour when comparing the primary Path Generation heuristic and the alternate Path Generation heuristic plotted for different start times.

5.4.3.6 Metrics for Analysis and Comparison

We use the same metrics applied in Experiment 2 (see Paragraph 5.3.3.4). Unlike in Experiment 2, original hazard values in Experiment 3 vary among the different trials and tests. Because of this variance, absolute hazard reductions are not a reliable tool for comparison. We therefore use percent hazard reductions for comparisons in this experiment.

5.4.4 Experiment 3 Algorithm Modification

We find that the Path Set Production sub-process requires too much memory to execute over a seven-day horizon as set up in this experiment on a machine constrained by 1GB of RAM. In order to complete this experiment, we modified the sub-process to conserve memory. Instead of generating paths over the entire planning horizon before optimizing, updating conditional hazard, and moving on to the next RCT, we set up the algorithm to generate paths one day at a time for each RCT.

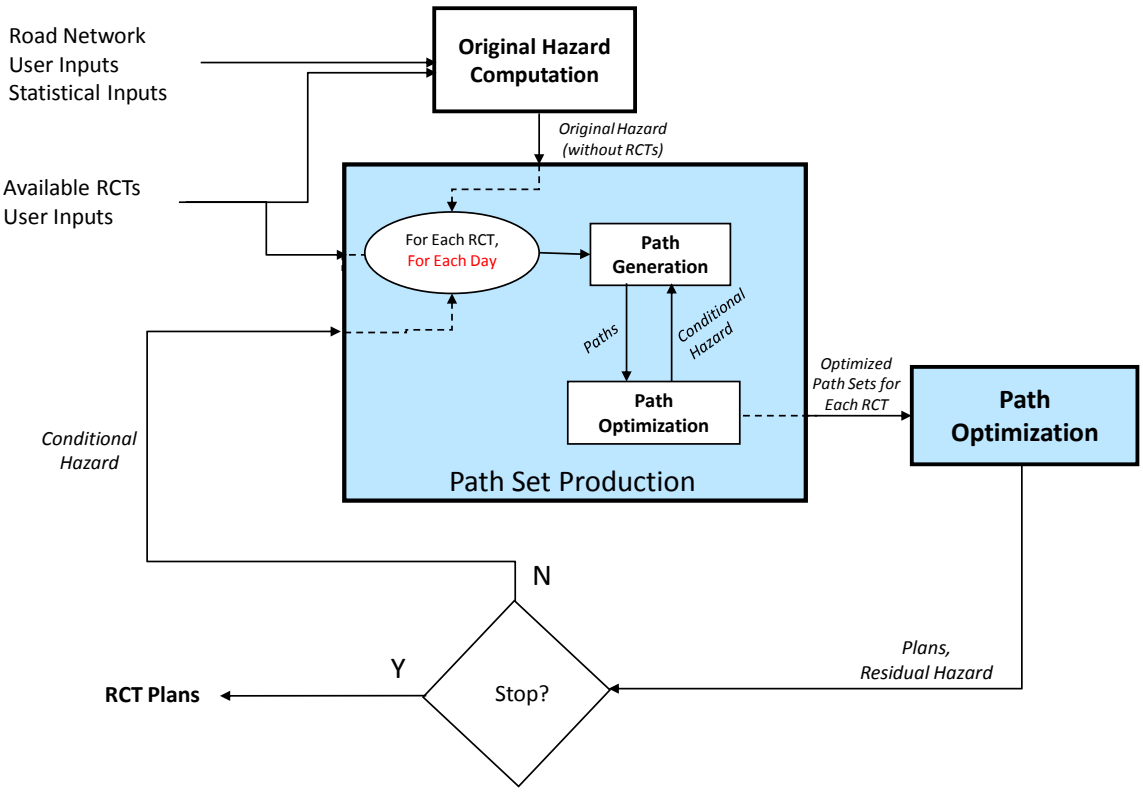

Figure 5-14: Route Clearance Planning Algorithm Modified for Experiment 3

The modified Path Set Production sub-process generates paths for a RCT for a particular day, selects a promising subset of these generated paths using the Path Optimization function, updates conditional hazard values based on the selection, then begins Path Generation for the next day in the planning horizon for the same RCT. Upon completing this process for each day in the planning horizon, the modified Path Set Production sub-process begins anew for the next RCT.

This modification requires the Path Optimization function to execute seven times for each RCT in the Path Set Production sub-process, as opposed to only once per RCT in the original sub-process definition. More frequent use of the Path Optimization function as a filter to select good paths for retention and discard the others prevents the algorithm from filling a computer's memory by storing too many paths, along with their associated reduced conditional hazard values and reduced original hazard values, simultaneously. Figure 5-14 depicts this modified algorithm, with the change in red for emphasis.

5.4.5 Experiment 3 Results

Table 5-6 displays the results for all four tests. We find that performance actually decreases slightly from the case with no deterministic information (test 3.1) when partial information is introduced (test 3.2). This apparent discrepancy from our conjecture requires further examination.

		Data Set 32 (Cambridge-based)			Data Set 31 (Utah)		
Test	Variable	Iteration 1	Iteration 2	Iteration 3	Iteration 1	Iteration 2	Iteration 3
3.1	$\#Paths^{(I)}$	25	49	73	197	393	(Stop)
	$\Gamma^{(I)}$	18.6365	20.1286	20.1407	29.6739	30.6544	
	$\Gamma_{\%}^{(I)}$	3.6	3.89	3.89	2.95	3.05	
	$runtime^{(I)}$	9.9 min	20.4 min	34.3 min	135.4 min	413.8min	
3.2	$\#Paths^{(I)}$	25	49	73	197	392	588
	$\Gamma^{(I)}$	16.4581	18.7044	18.7044	26.5285	28.5340	28.7178
	$\Gamma_{\%}^{(I)}$	3.16	3.59	3.59	2.63	2.83	2.84
	$runtime^{(I)}$	9.6 min	19.9	32.3 min	122.9 min	344.5 min	882.0 min
3.3	$\#Paths^{(I)}$	25	49	73	190	380	564
	$\Gamma^{(I)}$	56.3862	65.3222	66.9538	92.6768	105.9078	109.6351
	$\Gamma_{\%}^{(I)}$	12.5	14.5	14.9	12.8	14.6	15.1
	$runtime^{(I)}$	8.5 min	17.3 min	26.1 min	99.3 min	210.9 min	608.0 min
3.3h2	$\#Paths^{(I)}$	24	48	72	191	375	560
	$\Gamma^{(I)}$	62.1624	69.2828	69.3824	118.6400	127.0845	129.2374
	$\Gamma_{\%}^{(I)}$	13.8	15.4	15.4	16.3	17.5	17.8
	$runtime^{(I)}$	18.6 min	36.8 min	55.3 min	349.5 min	672.3 min	1029.0 min

Table 5-6: Experiment 3 Results

For the perfect information cases on the Utah road network, we initially noted extremely long computation time in the third iteration's Path Optimization step. To enable the mixed integer optimization to converge more quickly, we increased the gap tolerance from 0.01% to 1%.

5.4.5.1 Problem with Partial Information

Using deterministic and probabilistic use inputs in this model does not work well. Specifically, the alternation in the use probabilities between 1 and some other positive probability complicates the dynamics of the hazard recursion. When the two state Markov chain depicted in Figure 4-3 undergoes a transition with $P\_use_{a,t} = 1$ (i.e., when we know a convoy will use arc *a* at stage *t*), the next stage state probability mass shifts drastically to the state in which there are no type *k* IEDs on the arc. Because we assume that the convoy interacts with any and all emplaced IEDs, the only event that could result in a type *k* IED existing on the arc at stage $t + 1$ is an IED emplacement occurring immediately after the convoy passes. This event, in which Poisson IED emplacement occurs in a short time interval, has a low probability under our assumptions. In effect, the passing convoy clears the arc better than the RCT in the model, while at the same time incurring a large immediate cost in terms of hazard.

We have already seen that the reduction in hazard resulting from arc clearance decays over time. The transition probabilities (use and emplacement probabilities) drive the state probabilities toward some steady state (which is a function of the transition probabilities) with each transition. We refer to the decaying hazard reductions as the *enduring* benefits of clearing an arc. If an arc is cleared by a RCT immediately before a known convoy traversal, there is an immediate, significant reduction in hazard. However, the reduced hazard decays completely in the following stage and there are no enduring benefits. The reason for the instantaneous decay is that, regardless of what route clearance activities occur on the arc in any stages prior to the convoy's traversal, we know that the arc is clear after the convoy uses it, given our assumptions.

Figure 5-15 shows a plot of the original hazard values and the residual hazard values versus time (i.e., stage) for one arc from the test 3.2 solution for the Cambridge-based network. Values are summed over all IED types. The interval plotted consists of the stages comprising the second day: stages 289 to 576. We see that this arc is cleared five times over the course of the day. The earlier three times are during a period of increased probabilistic usage. The later two clearances

take place several stages before a scheduled convoy passes on the arc. Figure 5-16 shows the arc use function corresponding to the same arc over the same time interval.

The decay of the enduring effects can be seen in this plot. The goal of the route clearance planning algorithm is to minimize residual hazard, which is equivalent to maximizing the area between the two curves. We observe that the first three clearances have a significant effect on separating residual hazard from the original hazard on this arc. The later two clearances have effects that endure only until the scheduled convoy traverses the arc. At this stage, there is a significant reduction in hazard, but for following stages the two curves follow the same trajectory.

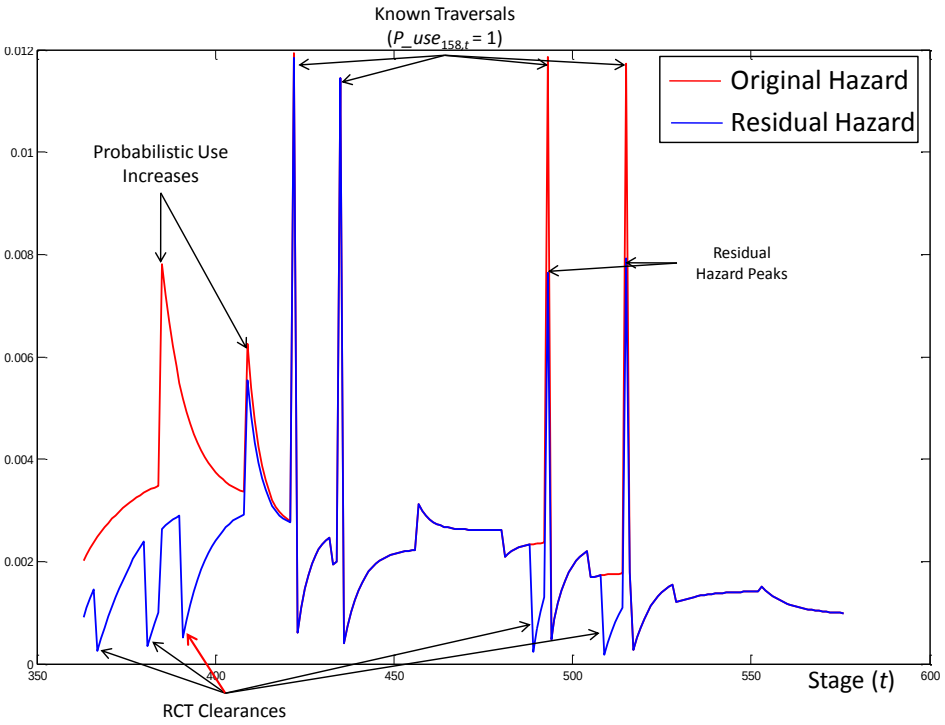

Figure 5-15: Cambridge-Based Network Arc 158, Day 2 Hazard Results for Test 3.2

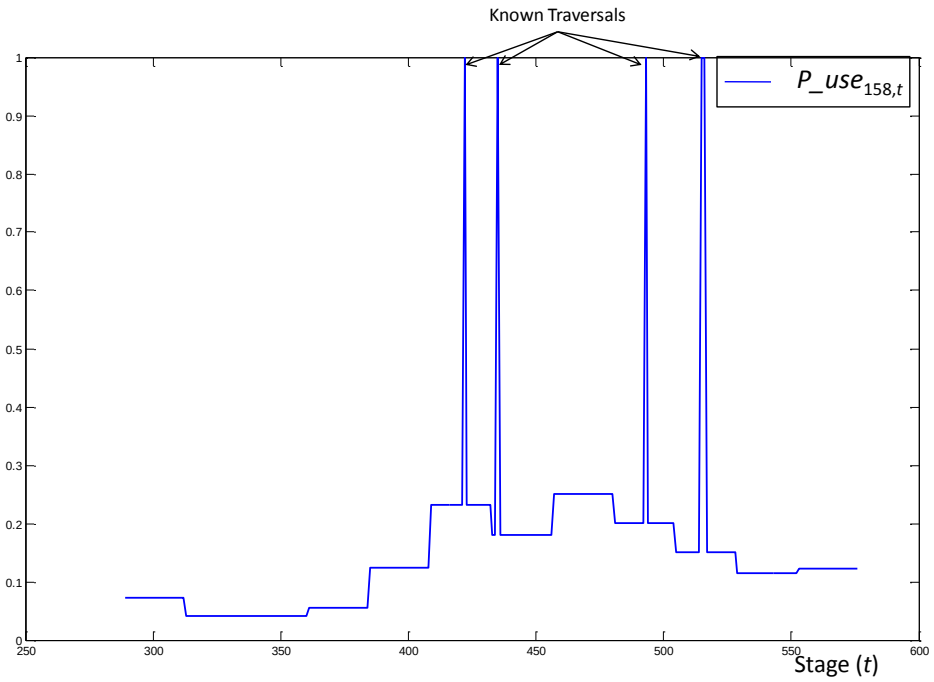

Figure 5-16: Cambridge-Based Network Arc 158, Day 2 Use Probabilities

The total effect (immediate and enduring) of the third arc clearance in the plot, which occurs at stage 390 and is indicated by the red arrow, is a reduction in hazard of 0.036. We can compare this value to the total effects of the later arc clearances, which are 0.0104, for the clearance at stage 488, and 0.0102, for the clearance at stage 508. In essence, having scheduled convoy information decreases the effectiveness of route clearance by wiping out any enduring benefits.

Looking to maximize total reduction in hazard, this model finds mission paths that are probabilistically more successful (under the Poisson assumptions for arrivals of IEDs and convoys on arcs) than they are in a deterministic model. We conclude that combining the deterministic convoy scheduling into the Poisson model for IED emplacement and arc usage rates violates its underlying assumptions and does not provide meaningful results.

5.4.5.2 Perfect Information

In the case of perfect information, we no longer rely on a Poisson approximation for convoy arrivals, so we do not have a conflict between probabilistic and deterministic inputs in the model. In this case, the IED emplacements are still modeled as a Poisson process. When a convoy is

scheduled to traverse an arc, the IED probability on that arc directly determines the hazard. In following stages, the IED probability on the arc is reset and begins to accrue again over time subject to the IED emplacement rates. The resulting evolution of hazard will yield the highest rewards (hazard reductions) for arc clearances immediately before scheduled convoys, or clusters of scheduled convoys, on arcs that have built up "IED in place" probability over a period of no usage, and there are no enduring effects in this model.

We observe that the route clearance planning algorithm, run with the purely deterministic arc usage inputs, results in over twice the hazard reduction as the probabilistic model.

5.4.6 Experiment 3 Summary and Conclusions

This experiment demonstrates the complexity involved with modeling IED and counter-IED operations, including some of the advantages and limitations involved with the methods we employ. Conjecture 3A is supported by results from test 3.3. Knowing when convoys are going to traverse arcs enables for much more efficient route clearance planning, and much more hazard reduction. Test 3.2 does not support these results, but is difficult to interpret because of the inconsistency in the hazard modeling used. The analysis discussed in Paragraph 5.4.5.1 reveals that different assumptions on the arc usage inputs can significantly change the nature of the IED hazard dynamics, and can conflict if applied within the same model. Additional testing, using alternate modeling methods and comparison metrics, would give more insight into the partial information scenario.

We can see from the results for tests 3.3 and 3.3h2 in Table 5-6 that the alternate Path Generation heuristic outperformed the primary heuristic, but not by enough to confirm that deterministic arc usage favors the alternate heuristic more than the case of probabilistic information. The alternate heuristic performed 3.6% better on the Cambridge-based network, and 17.9% better on the Utah network (measured in hazard reduction). Because the alternate heuristic performed an average of 17% better than the primary heuristic in Experiment 1 using probabilistic use data, we would expect at least that much improvement if the alternate heuristic was better suited to deterministic information. It is possible that the optimization processes in the algorithm somehow dilute the outperformance in Path Generation achieved by the alternate heuristic. This conjecture could serve as a conjecture for future experimentation.

5.5 Summary of Experimentation

The experimentation covered in this chapter verified some of our conjectures and demonstrated the algorithm's capability in producing some results. It also demonstrated the complexity of the problem and identified some of the limitations of the modeling approach. We present some ideas for improvement and recommendations for further research in the next chapter.

[This Page Intentionally Left Blank]

6 Future Research and Application as a Decision Support Tool

In the first part of this chapter, we present ideas and suggestions for future research and testing of the route clearance planning algorithm's models and functions. The second part of this chapter discusses the algorithm's potential applications in military planning and similar functions.

6.1 Future Research

The research presented in the preceding chapters has much potential for follow-on exploration. In this section we present ideas for additional experimentation, changes or additions to the modeling approach, and modifications to the algorithm's processes and functions that could provide additional insight and discover better solutions to the route clearance planning problem.

6.1.1 Additional Experimentation

We propose two experiments that would serve to validate the route clearance planning algorithm's applicability.

6.1.1.1 Comparison with Human and Other Heuristic Solutions

We recommend this algorithm be tested against other solutions obtained using the same inputs and hazard metrics. These other solutions can be human-generated solutions. For an ideal test, the input data to the route clearance planning algorithm can be included as information in an operations order during a unit training exercise involving IEDs and route clearance. As part of the MDMP, the unit staff would plan route clearance missions for available RCTs. The resulting plan could then be compared to the output of the algorithm and evaluated using the IED hazard metrics (these could even be weighted to support the training unit's objectives). Evaluation could also include the time spent conducting route clearance planning versus the algorithm computation time, and the actual outcome when the unit executes its plans in the training environment.

A smaller version of this experiment could be run in a more controlled environment focusing on route clearance planning only. One or more willing and interested participants, such as a military officer, would receive the input information on the road network, road segment use

rates, IED emplacement rates, and RCT availability. They would then devise a route clearance plan (or plans), presumably following the steps of the MDMP. This plan, and the time required for the participants to create it, could then be compared to the solution to the algorithm using the IED hazard metrics and the runtime.

A final method of producing alternate solutions for comparison is to choose various heuristics that seem intuitive (e.g., send the RCT on the most heavily used routes before an increase in use). The solutions to these heuristics and the time it takes to run them can then be compared to the performance of the algorithm.

One difficulty in any experiment involving a human planner is the representation of the inputs. The tables of values used in this experiment can be difficult for a person to comprehend because of the large, multi-dimensional arrays of data. To obtain a good human solution, the input data must be presented to the planner in a way that is intelligible. Figures, such as those used in summarizing emplacement and use rates in Appendix C, are good tools for displaying this information.

The results from this experiment would provide measures of algorithm effectiveness and efficiency.

6.1.1.2 Use of Real Data

In our research we have not closely examined the route clearance paths that comprise the solution set to the algorithm. As long as they are a function of notional data, there is not much we can glean from these paths, other than testing their metrics against those of other solutions as suggested in Paragraph 6.1.1.1. By using actual IED emplacement rates and road segment use rates from ongoing military campaigns, the solutions can be compared to actual operations as well as to actual IED events as they unfold.

We note two difficulties in accomplishing this test. The first is security; countries conducting ongoing military operations are not likely to release operational information (such as IED incidents, road use information, or mission plans) for anybody to research. Proper clearances must be obtained and the research must be sanctioned by an appropriate governmental agency. The second difficulty involves determining emplacement rates, use rates, detection rates, effectiveness rates, and all other probabilistic inputs from the available data. This step could

require a significant statistical effort, depending on what kind of actual data is available, and how reliable it is.

6.1.2 Changes/Additions to the Data Modeling Approach

We now suggest modeling ideas that have not yet been considered, but could prove useful in more accurately characterizing route clearance operations.

6.1.2.1 Modeling Terrain

We begin by presenting several variations to our terrain modeling approach.

6.1.2.1.1 Model Time-Dependent Terrain

We assumed in Chapter 4 that some parameters, such as arc classification, were not dependent on time. We suggest introducing time-dependence into these parameters so that we no longer need this assumption. In particular, detection probabilities will likely vary with different light conditions. Weather and light might also affect route classification; rain can puddle on roads allowing for more IED concealment, and darkness can produce a change in maximum clearance speed. These differences could be accounted for by varying the appropriate parameters as a function of stage, according to known weather and light data in the planning horizon. The drawbacks to making this change are that it requires more detailed input from the user and more computational memory.

6.1.2.1.2 Discrete Time Step Length

We conducted all experiments with a 5-minute discrete time step (or stage). Increasing the length of the stage will decrease computational requirements, but also decrease the level of precision in the solution. Decreasing the length of the stage will have the opposite effects. Quantifying the trade-offs is worth investigating.

6.1.2.1.3 Homogenous Arc Lengths

With some arc lengths exceeding 60 miles, the Utah road network used in Experiment 3 violated our assumption on reasonably short arc lengths. Not only do the long arcs violate our

assumptions, they also require an excessive time commitment to clear because RCTs must clear an entire arc before going to another arc in our model. When modeling the terrain, we suggest building a road network with homogenous arc lengths or homogenous arc clearance times (which does not allow for RCTs with different clearance speeds). Homogenous arc lengths would enable us to ensure the assumptions of reasonably short arcs (so that simultaneous IED incidents are negligibly likely), and would serve to normalize the hazard reductions. Creating a network with homogenous arc clearance times serves the same purposes, but does not allow for different RCT configurations to travel at different speeds. The additional benefit to having homogenous arc clearance times is that it simplifies the Path Generation label-setting heuristic to having only single-stage transitions.

Constructing a network with perfectly homogenous arc lengths or clearance times can be difficult because intersections and changes in terrain, along with emplacement and use rates, do not always occur at regular intervals. To compensate for this difficulty, arc lengths could be set at some very small value, which would require a very short discrete time increment. These effects would greatly increase the number of arcs and stages in the model, which in turn increases the memory and computational requirements of the route clearance planning algorithm exponentially. Finding "good" arc lengths that are approximately homogenous and short enough to satisfy our assumptions but long enough to allow for efficient computation is suggested for further research.

6.1.2.2 Modeling the Enemy

We have chosen to model the enemy by statistical IED emplacement rates. This somewhat static model of enemy activity could be replaced or appended to include enemy reactions to route clearance plans. A model that uses game theory or some other method to anticipate enemy reaction to route clearance missions over time is worth considering. An example of such a model is described in [Was06].

6.1.2.3 Convoy Planning and Routing to Minimize Hazard

In this research, route clearance missions are planned to reduce IED hazard to military convoys that occur in some probabilistic fashion or are determined in advance. We have not considered the convoy and patrol schedules as decision variables. We suggest investigation of iterative

optimization-based methods of scheduling convoys and RCTs to arrive at better, more "synchronized" solutions.

6.1.2.4 Synchronization with Other Efforts to Defeat the Device

We discussed in Chapter 2 other methods of defeating the device, such as the employment of observation posts. There are two ways we can include these other counter-IED efforts in our model. The first way is to plan them in advance, determine their effects on hazard, and update the hazard values to reflect their execution. The second way is to include them as available assets in the model and assign decision variables to them. This manner of incorporating other assets into the model is much more involved than planning them separately, requiring the generation of new sets of constraints and effects on hazard. However, it would allow for the algorithm to synchronize all of these assets, including RCTs. One of the characteristics of the solution route clearance mission paths from the experiments in this thesis is that they tend to concentrate on clearing a few high-value arcs numerous times. Given the data inputs, these redundant paths might be the best course of action. However, the high-value arcs could also be ideal locations for a static counter-IED effort, such as an observation post, which would free up the RCT to clear other arcs in the network.

An algorithm for allocating force protection assets, including counter-IED assets, in support of scheduled logistics convoys in a hostile, asymmetric environment has been developed in detail in [DeG07]. This algorithm does not include planning of route clearance missions but does include the employment of other "global" theater assets, such as aircraft-mounted electromagnetic jamming equipment, as well as "local" assets assigned to specific convoys, such as armored escorts or helicopter support. Coordinating the employment of the force protection resource allocation algorithm and the route clearance planning algorithm to protect military convoys and patrols in an asymmetric operational environment could be a first step in synchronizing the methods developed in this thesis with other counter-IED efforts.

6.1.2.5 RCT Modeling

In the following paragraphs we present several potential modifications or additions to our RCT modeling approach.

6.1.2.5.1 Modeling Risk to RCTs

IED risk to RCTs is not directly represented in our route clearance planning algorithm. How to separately model hazard to RCTs and how it should affect RCT mission plans is a potential subject of future research.

6.1.2.5.2 RCT Moving Without Clearing

There might exist situations when IED risk around a RCT's base location is relatively low, while more distant arcs have high IED hazard values. In such a situation, a RCT could be more effective by driving directly to the area of increased risk before beginning clearing operations. Our model can be amended to allow for RCT "driving" by introducing a set of duplicate arcs on the network. The duplicate arcs would have the same lengths, origin nodes, and destination nodes as their original counterparts, but could be given different classifications allowing for increased RCT speeds but decreased clearance effects. The hazard values on these arcs would be set at zero or linked to the hazard values on the original arc, depending on the modeling approach used. There could also be a cost associated with RCT driving in terms of IED hazard to the RCT as mentioned in the above paragraph. Introducing a duplicate network will multiply the memory and computational requirements involved in running the algorithm, but will also provide a simple way to expand its capabilities.

6.1.2.5.3 Mission Times

We have assumed in our modeling approach that route clearance missions are planned to meet certain fixed time constraints. Another approach would be to allow mission times to vary and assess the trade-off between planning one or two long missions versus many short missions per day.

6.1.2.6 IED Hazard Modeling

The IED hazard model created for this research presents ample opportunity for future research and improvements. Below we present a few ideas for further investigation.

6.1.2.6.1 Weighting the Hazard Values

The IED hazard function is determined solely from statistical inputs in our model. In Figure 3-2, we recognize that there are other inputs into the *decide* targeting activity that are not as quantifiable, such as commander's priorities, incoming intelligence data, and planned missions. To account for this additional information, we suggest including *hazard multipliers* in the model. These values would multiply by their associated hazard values to increase or decrease them as desired. The multipliers would be used, for example, to increase the hazard values at locations assessed by intelligence personnel to be five times more dangerous than other places in the network, based on specific information received or new political, social, or economic developments. Or, a "high-value" individual (e.g., a politician), might be traveling on certain arcs at certain times. A commander might assess that the strategic effects are 100 times higher if the high-value individual is injured or killed and weight the associated hazard values accordingly.

6.1.2.6.2 Employment Probability

We make the assumption that emplaced IEDs are always employed on a passing convoy, or found by it. We could abandon this assumption (and our justification of it) and introduce a probability of employment that would allow for the possibility of an IED remaining in place on an arc after a convoy has passed. This change would not significantly affect our original formulation with probabilistic arc use inputs, but might achieve more accurate modeling of the partial information and perfect information cases tested in Experiment 3. It would be interesting to see how allowing for this possibility affects the performance of the algorithm in the partial information case. A natural difficulty in employing this approach is in determining how often this event (a convoy passing an emplaced IED without incident) actually occurs.

6.1.2.6.3 IED's Effects on Non-Military Entities

We have only considered IED threats to military movements for which statistical data is available. Expanding the IED hazard function to include other types of targets would not require significant modifications to the existing algorithm. There would have to be a method of determining the non-military targets' arc use rates. Different hazard weights could be introduced for different types of targets (see Paragraph 6.1.2.6.1), as well as different probabilities of employment (as discussed in Paragraph 6.1.2.6.2).

6.1.2.6.4 IED Hazard as a Partially Observable Markov Decision Process

Another possibility is to model the entire road network as a Partially Observable Markov Decision Process (POMDP). The method we employed in this thesis contains many of the characteristics of a POMDP, such as the Markov property, but does not explicitly define the problem as one. Exploration of this modeling approach and its differences from the one we employ offers potential further research into this problem.

6.1.3 Path Set Production Sub-Process

We now discuss some opportunities for further investigation and improvement of the functions, heuristics, and sub-processes employed in the route clearance planning algorithm, beginning with the Path Set Production sub-process.

6.1.3.1 Path Set Production Controls

We developed controls to manage the inputs and select paths from the outputs of the Path Generation function. These controls had the purpose of creating a good Path Set with diverse start times for Path Optimization. We now propose alternate approaches to controlling the Path Set Production sub-process to achieve the same goals.

6.1.3.1.1 Finding Good Mission Start Times Without Generating Many Paths

As suggested in Chapter 5, it would be more efficient if we could somehow determine which start time inputs into the Path Generation function would result in "good" mission paths, instead of generating many paths, which takes time, and then selecting "good" paths from those generated. Relating these "good" mission start times to the conditional hazard function was our aim in Experiment 1, and was not successful. However, we did not conclude that such a method does not exist. We suggest further investigation into relationships between the problem inputs and the best start times for each RCT.

There still exists the possibility of a relationship between a weighted, summed conditional hazard function and the best mission start times, if the weights are appropriately selected. We did not include arc distance from the RCT's base location in our weighting criteria for Experiment 1, but this distance might be important. Closer arcs could be assigned higher weights, because they

cost less mission time for the RCT to clear than arcs that are further away. Instead of simply guessing at the weights, statistical methods could be employed to relate conditional hazard values to hazard reduction on multiple data sets. It might also be helpful to examine separately the arc use rates, because these appeared to have a large impact on hazard evolution and best start times in Experiment 1.

Another method to reduce the number of paths generated in search of good mission paths is to store some information about the shape of the conditional hazard reduction curve as a function of mission start time (as plotted for the results of Experiment 1 in Chapter 5, Figures 5-4 through 5-11). We observed in Test 1.2c that hazard values did not always reduce significantly as a result of conditioning on some missions being conducted. Conditional hazard reduction as a function of start time can be found by generating many paths in the first iteration, and then assumed to be relatively unchanged in follow-on iterations, excepting certain start times that coincide with conditionally scheduled missions. Re-computing conditional hazard reduction only for missions with these start times would cut down on computation time and memory usage, and might enable "good" path selection without the need for many iterations of Path Generation in each iteration.

6.1.3.1.2 Criteria for Path Selection

The Path Optimization function used to select good paths in the Path Set Production sub-process does not need to have the same set of constraints as the Path Optimization function used in the larger route clearance planning algorithm. These constraints can be relaxed to allow for more paths to be chosen, or to allow for combinations of paths to be chosen that cannot be feasibly executed simultaneously. The purpose of this function is to choose a good subset of the generated paths to add to the Path Set in the current iteration. The feasibility constraints were used to limit the number of paths selected and to ensure some distribution in the mission start times. Other path selection criteria could be applied that would accomplish the same goals, but allow for more diversity in the paths selected.

6.1.3.1.3 Order of RCTs

In each iteration of the Path Set Production sub-process, paths were generated, and optimized, and conditional hazard values were updated for the RCTs in the same order as in previous

iterations. Changing the order of the RCTs for each iteration would likely increase the diversity of the paths being generated and is a good subject of further inquiry in this algorithm.

6.1.3.1.4 Constraining Start Times

Generating mission paths with diverse start times is important in the performance of this algorithm. For example, if for each iteration, each path introduced into the Path Set for a specific RCT had the same start time, the RCT would never be able to conduct more than one mission in the solution because of its feasibility constraints. Introducing additional constraints to ensure that each RCT has enough diversity in mission start times in the Path Set can help with performance. These types of constraints are more important when Path Generation is carried out separately for each day, as it was in Experiment 3 over a seven-day planning horizon. Without these constraints, it is possible that some RCT resources will go unused in the algorithm's solution because no feasible allocation of mission paths was generated.

6.1.3.1.5 Another Method for Updating Conditional Hazard

In all iterations after the first route clearance planning algorithm iteration, the conditional hazard values are computed based on a full allocation of all RCTs (assuming start times are appropriately distributed if Path Generation is conducted by day, as discussed in the above paragraph). It might make sense, therefore, to re-compute conditional hazard for each RCT's Path Generations without including any hazard reductions resulting from any of the RCT's missions. The only problem with this approach is that it might result in the same paths being generated in each iteration, especially if the area of operations is partitioned so that there is not overlap between RCTs. In this case, removing an RCT's hazard reductions from the conditional hazard values resets the conditional hazard values back to the original hazard values for all arcs in the RCT's sector.

6.1.3.2 Heuristics

In this section, we present some ideas for further investigation into the heuristics used to generate paths in the route clearance planning algorithm, and suggest opportunities for potential improvement.

6.1.3.2.1 Discount Hazard Reduction for Arcs Previously Cleared

We have mentioned previously the tendency for the route clearance planning algorithm to produce solutions that focus on clearing several high-risk arcs repeatedly. This course of action might not be intuitive, but it could be the best way to reduce IED hazard given the notional data sets. However, there might be a compelling reason that is not captured in the input data that makes it undesirable to clear a few arcs repetitively. To account for this possibility, we suggest introducing discounting factors for multiple arc clearances. These factors would be multipliers between zero and one, and would be a function of the number of times an arc has been cleared in a path being constructed by the Path Generation label-setting heuristic. Arcs that have been cleared several times would have discount factors close to zero. These factors would multiply with hazard reduction, essentially decreasing the RCT's effectiveness on the arc and causing the Path Generation heuristic to choose other paths.

There might be discounting factors that improve the Path Generation heuristics in some cases by effecting better path pruning.

6.1.3.2.2 Cost-To-Go and Cost-Per-Stage Approximations

All reduced conditional hazard computations carried out by the Path Generation heuristics are computed exactly. In the alternate heuristic, these computations require time-consuming recursions to find the enduring hazard reductions. Both heuristics, particularly the alternate heuristic, would benefit from an approximation for future-stage reduced conditional hazard values on arcs as a function of current-stage reduced conditional hazard and input usage and emplacement rates, eliminating the need for the recursive calculations.

6.1.3.2.3 Cost-Incurred Versus Cost-Per-Stage

In the primary Path Generation heuristic, the label-setting method seeks to minimize cost-per-stage, which is the summed reduced conditional hazard for the stage corresponding to a particular state. We choose this method because the instantaneous conditional reduced hazard contains enduring hazard reductions from all previous decisions. However, we have not compared this label function with one that minimizes total cost incurred, summed over all previous stages, as suggested in Paragraph 4.2.4.1.2.8. Using the reduced conditional hazard

values summed over all previous stages as the labeling function for each state might perform better than the labeling function used in this thesis, and is worth further investigation.

6.1.4 Path Optimization Model

Finally, we present some opportunities for further development and improvement of the Path Optimization mixed integer program, especially the "overlaps" preprocessing and integration.

6.1.4.1 Pair Overlaps Versus Full Overlaps

In Experiment 2, we found that by identifying and introducing overlap variables and coefficients for overlapping path pairs only, we could speed up computation time significantly without changing the optimal solution. Further analysis of this idea would be useful in justifying its use and in finding pathological cases in which searching for pair overlaps does not provide a useful approximation of the exact mixed integer program.

6.1.4.2 Overlap Tolerance

Increasing the magnitude of the difference between the input conditional hazard values and the reduced conditional hazard values on an arc required to generate an overlap variable also significantly reduced the computation time. We did not conduct experiments with different values. Additional analysis of how large tolerances can be made before performance is adversely affected would be useful in making the algorithm more efficient.

6.1.4.3 Overlap Generation

Another idea pertaining to overlaps is to omit any overlap generation prior to optimization. The Path Optimization function would be run in this case with only the total hazard reduction coefficients, integer path variables, and path feasibility constraints. Once a solution is found, overlap variables, coefficients, and constraints would be identified and introduced into the problem for the paths in the solution set only. The mixed integer program would then be re-run with the additional variables, coefficients, and constraints. This iterative method is likely to converge to the optimal solution, because introducing overlap variables, constraints, and coefficients into the problem will only reduce the objective value (recall that this is a

maximization problem). If this technique converges in a small number of iterations, it could prove faster than finding all feasible path combination overlaps in preprocessing.

6.1.4.4 Best Integer—Linear Relaxation Solution Gap.

We indicated in Chapter 5 that, in order to speed the Path Optimization mixed integer problem processing time, we increased the gap tolerance between the best integer solution and the lower bound from the linear relaxation for some trials. The effect of this gap increase was not assessed, but by increasing the gap we traded performance (in terms of hazard reduction) for computational speed. Further experimentation using different values for this parameter could provide more insight into values for this parameter that result in the best combination of performance and efficiency for different numbers of paths being optimized.

6.2 Application as a Decision Aid

The route clearance planning algorithm has a potential application as a decision aid for military planners faced with creating and scheduling route clearance missions. In this section, we propose a way to integrate the algorithm into the military planning process and discuss some of the human-interface requirements.

6.2.1 Integration into the Route Clearance Targeting Process and MDMP

The route clearance planning algorithm presented in this thesis could be used to assist planners in both COA development and COA analysis during the *decide* targeting activity, as depicted in Figure 6-1. Users could use the algorithm to generate multiple COAs by manipulating the algorithm's inputs. For example, hazard weighting values could be changed to reflect the intelligence officer's assessment of the enemy's most likely and most dangerous courses of action. Another possibility would be to run the algorithm under different sets of user assumptions on IED effectiveness (by type) or arc use rates. Each COA generated by the algorithm consists of a schedule of route clearance missions and associated hazard reduction metrics. The benefit of employing the route clearance planning algorithm as a decision aid is that it can produce multiple courses of action while staff planners are free to conduct other tasks. For example algorithm-generated COAs could be produced in parallel with human-generated COAs.

The hazard reduction metrics provided as part of the algorithm's output are quantitative measures of each COA's effectiveness, and serve as a useful COA analysis tool. With little effort, human-generated route clearance COAs could be inputted into the algorithm and analyzed using the same hazard reduction metrics. Military planners would likely have other metrics to consider as well. Some of these metrics might be possible to retrieve from the algorithm outputs (e.g., total length of roadway cleared), while others might be more qualitative in nature and would require some human analysis of each COA (e.g., disruption to society, or measures of competing demands for logistical support). The advantages of using the hazard reduction metrics as an analysis tool are: (1) they can be computed very quickly, and (2) their quantitative nature makes them useful for COA comparisons. Once all COAs are analyzed, they are compared by the staff. The staff could compare hazard reduction metrics only, or could make more general comparisons based on a combination of the aspects analyzed for each COA.

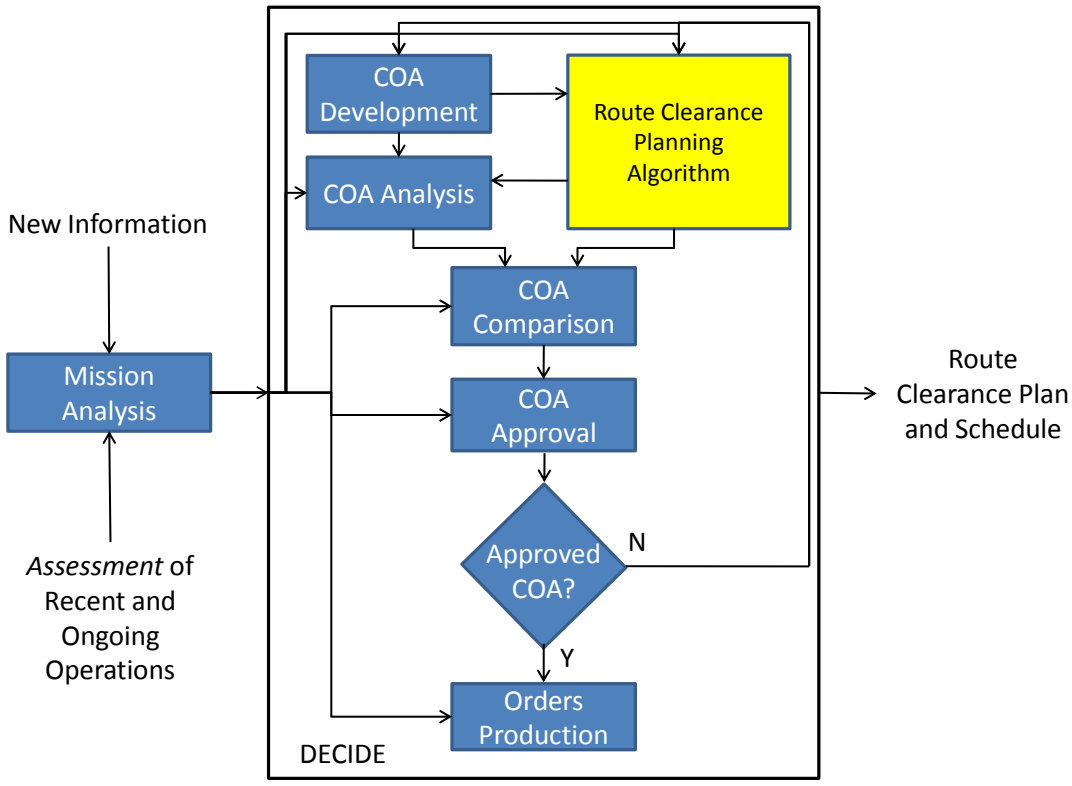

Figure 6-1: Route Clearance Planning Algorithm as a Decision Aid in the Targeting Process

6.2.2 Interface Requirements

In order for the algorithm to function effectively as a decision support tool, it should have a user interface that allows for easy manipulation of inputs, processes, and outputs, making it a *human-guided* algorithm. Human-guided algorithms allow users to control the algorithmic search for high quality solutions, capitalizing on the strengths of both humans and computers to improve performance and efficiency [Tho07]. We use the rest of this chapter to discuss some of the controls the user should have in the route clearance planning algorithm, and to suggest some properties of a useful user-interface.

6.2.2.1 Manipulation of the Inputs

The statistical data could be updated automatically from databases containing IED incident reports and route usage information. However, other information might exist that is not evident in the statistics. Religious celebrations, cultural events, economic changes, or political activities can affect IED operations or temporarily make certain locations much more strategically important than other locations. Users can account for this type of information when using the route clearance planning algorithm by weighting the hazard values or by changing specific input probabilities. A graphical user interface could help enable the user to have this level of control. The interface would display the road network overlaid on map data and allow the user to select and group arcs, then view and modify the associated input parameters. User-define arc groups could be used to partition the area of operations. IED types and effectiveness and RCT parameters could also be modified in table formats.

6.2.2.2 Manipulation of Processes

In order to use the route clearance algorithm to meet the diverse needs of military planners, users should have some control over the algorithm's processes. Parameters that control the algorithm's speed and performance, including number of iterations, stopping criteria, and controls on the number of paths generated in each iteration should have default values that can be adjusted to meet the needs of the user. Also, users should be able to run the algorithm to obtain partial solutions for a subset of RCTs or a subset of days in the planning horizon.

6.2.2.3 Solution Modification and Creation

Planners should be able to view and make modifications to the solutions coming from the route clearance planning algorithm using the graphical user interface. The display should also include hazard reduction metrics that can be compared when modifying a solution. The user should also be able to run additional iterations of the algorithm using user-modified or user-generated solutions (created using the graphical user interface) as inputs.

Incorporating all of these algorithm controls into a user interface would require a significant software development effort, but would increase the algorithm's utility by enabling the user to orient the algorithm to meet specific planning needs.

7 Summary and Conclusions

The purpose of this chapter is to summarize the problem, the modeling approach, and the experimentation that comprises this thesis. We end this thesis with some general conclusions drawn from our modeling techniques and experimental results.

7.1 Summary

In this thesis, we identified the problem of planning and scheduling route clearance teams. In Chapters 2 and 3, we conducted an in-depth analysis of this problem from the perspective of United States Army units conducting stability operations in a contemporary operating environment in which IEDs are employed. First, we looked at the route clearance problem in the larger contexts of asymmetric warfare, counter-insurgency, and assured mobility operations. Next, we used the activities of the Army targeting process to decompose the route clearance planning problem into three functions: a mission planning (i.e. *decide*) function, a mission execution (i.e. *detect* and *deliver*) function, and an operational *assessment* function.

After analyzing the route clearance planning problem, we developed a technical model in Chapter 4. The first step in the model development was the creation of a data model. We employed a graph, consisting of sets of arcs and nodes, to model the road network as the terrain over which IED events occur. We approximated IED events and convoy traversals of arcs as independent Poisson arrival processes, and we applied a Markov model to approximate IED dynamics on each arc in the graph. We also determined our objective function, and from it built a "Hazard" function that converts state probabilities from our Markov model to additive measures of IED risk for each arc, which we refer to as *hazard* values. We estimated the effects of route clearance efforts on these hazard values and then applied two approximate dynamic programming methods to search for route clearance paths in the graph that resulted in the most hazard reduction. The paths generated by the dynamic programs were then optimized using a mixed integer program, which determined the most hazard-reducing combination of paths that could be executed, given certain feasibility constraints on the RCTs. In addition to describing each of these methods, we discussed their evolution and relationships to other methods that were not employed because of their computational time requirements.

In Chapter 5 we conducted three experiments. The first experiment sought a way to estimate good route clearance mission start times based on the input data alone, without having to generate mission paths for each start time. We were not successful in finding a useful estimation. The second experiment evaluated the effects of allocating a unique subset of the road network graph to each RCT in the model, versus assigning all RCTs to the entire area of operations. We found that the algorithm required much less computational time in the partitioned case, but did not obtain as much hazard reduction as the un-partitioned case. We also explored two methods for increasing computational efficiency in the second experiment: path selection and pair-overlap only preprocessing. We found that both techniques significantly improved computational efficiency with minimal loss of performance. In the third experiment, we examined the value of having foreknowledge of route usage times in the area of operations. We found that in the case of perfect information, the route clearance planning algorithm performed much better than in the case of probabilistic information only. The partial information test was inconclusive and requires further experimentation.

Finally, in Chapter 6 we presented many opportunities for further research into the route clearance planning problem, including changing modeling assumptions, applying different techniques and modeling approaches, and improving solution procedures. We also discuss the algorithm's applicability as a decision support tool and consider some of the user interface requirements.

7.2 Conclusions

7.2.1 Modeling IED Warfare

Modeling IED activities as a two-state Markov process (essentially a queuing model) as developed in Section 4.2.3 is a valuable contribution to developing algorithms related to IED defeat. This model has the following desirable properties:

- Probability of an IED existing on an arc at a certain stage depends on (1) the probability an IED existed on the arc during the previous stage, (2) the IED emplacement probability, and (3) the arc use probability.
- IED hazard on an arc at a specific stage can be quantified as a function of the probability an IED is in place on an arc and the probability the arc is traversed by a convoy.

- Effects of route clearance on IED hazard diminish over time at a rate determined by the IED emplacement probabilities.
- The queuing process enables the computation of many interesting values, such as the expected rate of IED attacks on each arc and the expected amount of time an IED is emplaced prior to detonation on each arc.

The first property is desirable because it is intuitively realistic. IED existence on an arc is the union of two events: an IED existed on the arc in the previous stage that was not used, or an IED was just emplaced on the arc. The second property also intuitively makes sense because there is no risk of an effective IED attack unless both the IED and the target are present at the attack location. The third property is one that is acknowledged in Army doctrine; essentially a route does not remain clear unless it is secured and observed after a route clearance mission [FM109]. This property quantifies the diminishing effects of route clearance. The fourth property was not used directly in this thesis, but would be useful in further analyzing the evolution of IED operations. The expected time until detonation for IEDs might also exhibit an interesting relationship with the best times to clear arcs. We conclude that the two-state queuing model for IED activities is valid and has potential beyond the work carried out in this thesis.

7.2.2 Path Generation—Path Optimization Methodology

The solution space for this problem is too large for exact optimization methods. The technique we have presented uses heuristic methods to generate potential route clearance missions, followed by exact optimization methods over the set of paths generated. This procedure is similar to column generation, making the problem more tractable.

7.2.3 Value of Information

Our experimentation demonstrated that having information ahead of time about convoys and patrols allows for more effective planning and scheduling of route clearance teams. We showed this result with convoys occurring randomly according Poisson processes. If convoys and patrols were scheduled in a way that maximized the use of routes recently cleared, the results could prove better. One way commanders could implement this type of scheduling to increase the benefits of route clearance would be to restrict movement on routes to certain time intervals each day, which would be chosen in advance each day to handle the amount of traffic required to

move. Route clearance teams could clear routes immediately prior to their "opening," and the ensuing convoy density would discourage IED emplacement during the unrestricted interval.

7.2.4 Model Applicability

The algorithm we have developed is applicable to counter IED (route clearance) operations, but can also extend to routing and scheduling problems outside of military operations, such as the planning of police patrols when there is a known crime distribution. Adding a graphical user interface that gives a human user easy control and interaction with the algorithm is a key to increasing its usefulness. It has the potential to benefit decision makers by processing large amounts of available information and generating courses of action quickly. By using numerical optimization methods and heuristics, it ensures that the route clearance plans produced are driven by some objective, such as minimizing the effects of IEDs, and provides metrics for comparison. Its implementation can contribute to diminishing the strategic effects of IEDs, including casualties, in the contemporary operating environment.

Appendix A: Abbreviations and Acronyms

Abbreviation/ Acronym	Term
ACR	Armored Cavalry Regiment
ADCOORD	Air Defense Coordinator
AG	Adjutant General
ALO	Air Liaison Officer
AMO	Aviation Medical Officer
AO	Area of Operations
ASOC	Air Support Operations Center
AVN	Aviation Officer
BCT	Brigade Combat Team
CCIR	Commander's Critical Information Requirements
CDR	Commander
CH	Chaplain
CML	Chemical Officer
COA	Course of Action
COE	Contemporary Operating Environment
CofS	Chief of Staff
COMDT	Commandant
CP	Command Post
CSM	Command Sergeant Major
DEP	Deputy
EFP	Explosively Formed Penetrator
ENGR	Engineer
EOD	Explosive Ordnance Disposal
FIN	Finance Officer
FSCOORD	Fire Support Coordinator
G1	Personnel Officer
G2	Intelligence Officer
G3	Operations Officer
G4	Logistics Officer
G5	Civil Affairs Officer
HIST	Historian
HQ	Headquarters
IED	Improvised Explosive Device
IG	Inspector General
IPB	Intelligence Preparation of the Battlefield
LO	Liaison Officer
MDMP	Military Decision Making Process
NGLO	Naval Gunfire Liaison Officer
OPLAN	Operational Plan
OPORD	Operations Order
PAO	Public Affairs Officer
PM	Provost Marshal
RAM	Random Access Memory
RCT	Route Clearance Team
SGS	Secretary of the General Staff
SIG	Signal Officer

SJA	Staff Judge Advocate
SURG	Surgeon
SWO	Staff Weather Officer
T-72	Soviet Tank
US	United States
VBIED	Vehicle-Borne Improvised Explosive Device
WARNO	Warning Order

Appendix B: Assumptions and Their Implications

This appendix provides a list of assumptions, along with their implications and justifications. We group assumptions into two categories: data modeling assumptions and RCT assumptions. We employ data modeling assumptions to obtain desirable properties in our model, including simplicity and validity. The RCT assumptions are made to enable RCT modeling, and are more flexible in our technical approach.

B.1 Data Modeling Assumptions

B.1.1 IED Assumptions

B.1.1.1 IED Effectiveness Availability

Assumption: Given a set of criteria that define an "effective" IED (e.g. IED causes a casualty), IED effectiveness rates, conditioned on interaction with military forces (e.g. a passing convoy), can be determined for each type of IED modeled.
Implication: This assumption enables the prioritization of route clearance assets based on which IEDs are the most dangerous. It implies that data exists from which these IED effectiveness rates can be determined.
Justification: Records are kept during military operations that include all reports of IED incidents and their outcomes. These records could be queried to determine IED effectiveness rates.
Effect if removed: Without this assumption, we would not be able to distinguish IEDs by type in the model and would have to assume that all are equally dangerous.
References: Paragraph 4.1.1 discusses this assumption.

B.1.1.2 Existence of IED Emplacement Patterns

Assumption: Insurgent forces employing IEDs do so in statistically observable patterns or in response to environmental variables (such as ongoing military operations, civil operations, cultural or religious observances, social demographics, etc.).
Implication: Approximate IED emplacement probabilities can be determined with some level of accuracy.
Justification: According to US Army doctrine, operational patterns are common to all insurgencies [FM109]. Using pattern analysis to predict future IED likelihood is taught at the Army National Training Center (NTC) in California [Mag05] and in Army manuals [FM109].
Effect if removed: Without this assumption, we would not have a way to determine if IEDs emplacements were more probable at some locations and times than others. A route clearance planning algorithm could still be developed, but it would have to rely only on arc use information.
References: Paragraph 4.1.4 discusses this assumption.

B.1.1.3 Employment of IEDs

Assumption: Non-route clearance military movement always interacts with an existing IED on an arc it traverses.
Implication: If a convoy moves on an arc with an IED emplaced, it incurs casualties with probability P_eff_k. The IED is always removed from the arc as a result of the convoy traversing it. When using probabilistic arc traversals, this assumption simplifies the model without loss of generality (arc use probability could be interpreted as probability a convoy uses the arc and interacts with the IED). When deterministic arc use data is introduced, the implication is stronger because this interpretation can no longer be applied.
Justification: Insurgents allowing IEDs to remain in place while convoys pass by risk being discovered and killed, so it is in their interest to use an IED that is already in place.
Effect if removed: A probability of detonation could be introduced into the problem without considerable effect on computational effort or processes. Accurate determination of this probability would be very difficult to determine.
References: Paragraph 4.1.5.2 discusses this assumption. Paragraph 5.4.5.1 addresses its effects in Experiment 3 and Paragraph 6.1.2.6.2 discusses models based on alternate assumptions.

B.1.1.4 Independence of IED Incidents

Assumption: IED incidents are independent events for different arcs and different times.
Implication: Probabilities can be multiplied to compute intersections. This assumption allows the derivation of our objective function.
Justification: This assumption supports the structure of our technical approach and enables the development of a model that has the desirable properties given in Section 7.2.1. In reality, IEDs are not stochastic events; they are carefully planned and executed operations.
Effect if removed: Introducing IED dependencies makes the problem more difficult mathematically. Determining IED dependencies, if any exist, also presents a significant challenge. Without assuming dependence or independence, the probabilistic modeling approach does not work.
References: Paragraph 4.2.2 discusses this assumption.

B.1.1.5 IED Emplacements as a Poisson Process

Assumption: IED incidents are approximated by a time-varying Poisson process.
Implication: IED emplacements have a "memoryless" property. The probability of an emplacement at any time does not depend on the amount of time that has elapsed since a previous emplacement.
Justification: This assumption simplifies the computation of emplacement probabilities and supports our employment of a two-state Markov chain to model IED emplacements and detonations on each arc.
Effect if removed: Modeling the evolution of IED probabilities across the road network becomes more complicated using other assumptions on the IED emplacement process.
References: Paragraph 4.1.4 and 4.2.3 discuss this assumption.

B.1.1.6 Markov Property

Assumption: IED existence on an arc at each stage depends only on the probability an IED existed on the arc in the previous stage and the probabilities of IED emplacement and arc usage.
Implication: Probabilities of IEDs existing on arcs develop according to Markov processes.
Justification: This property follows from our previous assumptions. Intuitively, the only two ways for an IED to exist on an arc at a certain time is for it to be emplaced at that time, or for it to have been emplaced at an earlier time and not yet employed.
Effect if removed: Another method of modeling the evolution of IED "existence" probabilities would have to be devised.
References: Paragraph 4.2.3 discusses this assumption.

B.1.2 Terrain Assumptions

B.1.2.1 Static Terrain

Assumption: Arc length and arc characteristics do not change with time.
Implication: Weather and light data are not considered in our model. IED detection probabilities and RCT clearance speeds are the same for a given arc for all stages in the problem.
Justification: This is a simplifying assumption. Arc length does not change as a function of time. Road characteristics could change with time, depending on how they are defined by the user.
Effect if removed: Additional data model inputs, such as RCT speeds and detection probabilities for night and day on each road type, would be required if this assumption was not made. However, it would not significantly alter the technological approach or increase the computational effort required.
References: Paragraph 4.1.2.1 discusses this assumption.

B.1.2.2 Short Discrete Time Increments

Assumption: Discrete time increments used to model stages are sufficiently short so as to marginalize the probability of multiple events (i.e., IED emplacements and arc traversals) occurring during the same stage.
Implication: This assumption supports the two-stage Markov model developed to characterize the evolution of IED activity.
Justification: We can set the time increment at any value we desire, but smaller values increase computational effort required.
Effect if removed: Without this assumption, another model of the evolution of IED activities would have to be developed.
References: Paragraph 4.1.2.2 discusses this assumption.

B.1.3 Arc Use Assumptions

B.1.3.1 Arc Use as a Poisson Process

Assumption: Similar to IED emplacements, convoy traversals on an arc are approximated by a time-varying Poisson process.
Implication: Arc traversals (or uses) have a "memoryless" property. The probability of traversal at any time does not depend on the amount of time that has elapsed since a previous traversal.
Justification: This assumption simplifies the computation of arc use probabilities and supports our employment of a two-state Markov chain to model IED emplacements and detonations on each arc. Convoys from different elements are normally planned and executed by different individuals acting independently.
Effect if removed: The evolution of IED "existence" probabilities across the road network changes according to the method used to model arc traversals. Using more general stochastic processes to model convoys could increase the complexity of the problem. Using deterministic convoy arrival data is possible, but requires reconsideration of assumption B.1.1.3.
References: Paragraph 4.1.5.2 and 4.2.3 discuss this assumption.

B.1.3.2 Homogenous Convoys

Assumption: All convoys interact equally with IEDs. IEDs are equally effective on each convoy.
Implication: We do not need to account for differences in convoy tactics or composition when considering whether a convoy will interact with an IED or whether an IED will be effective when employed against a convoy.
Justification: We use this assumption only to simplify our model of the evolution of IED activities on each arc.
Effect if removed: Allowing for differences in convoy types and compositions would require a significant amount of information. IED effectiveness rates and arc use rates would be required for each convoy classification. The data model would have to be altered slightly to account for different convoys. The increased data requirements and the modification to the data model would result in needing more memory and computational effort in the route clearance planning algorithm.
References: Paragraph 4.1.5.2 discusses this assumption.

B.1.3.3 Arc Use Independence

Assumption: The probability of a convoy traversing an arc during a given stage is not affected by conditioning on whether an IED exists on the arc during the same stage.
Implication: State transitions in the two-state Markov process do not affect each other. The probability of convoy interaction with an IED on an arc can be computed as the product of the probability an IED is in place on the arc and the probability a convoy traverses the arc.
Justification: Presumably, an IED in place on an arc is unknown to military convoys until interaction occurs, resulting in detection or detonation and IED removal.
Effect if removed: We would have to determine what dependence, if any, exists between arc use and IED existence on the arc, in order to calculate IED hazard. Defining this dependence would be a difficult task.
References: Paragraph 4.2.3 discusses this assumption.

B.2 Route Clearance Assumptions

B.2.1 Successful RCTs

Assumption: RCTs are reasonably capable of detecting the types of IEDs they are configured to detect.
Implication: Reduction of IED hazard resulting from route clearance activities can be modeled.
Justification: This assumption is implicit in the military's motivation to plan and execute route clearance operations.
Effect if removed: Without this assumption, the route clearance problem addressed in this thesis is moot.
References: Paragraph 4.1.5.1 discusses this assumption.

B.2.2 RCT Bases

Assumption: Route clearance missions originate from and end at the same predetermined location.
Implication: The set of possible mission paths for a RCT can be constrained to only paths beginning from and ending at the RCT's base location.
Justification: Logistical and security requirements drive the need to house equipment and soldiers at fixed installations during deployments. Allowing route clearance missions to begin and end at different locations would be unsustainable in practice.
Effect if removed: The number of possible mission paths becomes larger, but the Path Generation heuristics would still function at approximately the same speed. Planned directional missions, constrained to originate from one specified base and end at another, could be introduced into the algorithm without considerable effort or changes to the methodology.
References: Paragraph 4.1.5.1 discusses this assumption.

B.2.3 RCT Mission Duration Constraints

Assumption: Planners develop maximum and minimum mission durations as constraints when planning route clearance missions.
Implication: The set of possible mission paths for a RCT can be constrained to include only paths that fall within specified mission duration limits.
Justification: Continuous, unconstrained route clearance operations would be ideal, but are not feasible because of logistical requirements, including fuel, maintenance, and crew rest. These requirements limit the amount of time a RCT can be employed before it must return to its base. However, the effort required to prepare and execute a route clearance mission makes it inefficient not to make use of all or most of the available mission time.
Effect if removed: Allowing for variable mission durations increases the size of the solution set and makes this problem much harder to solve. It could also lead to better solutions.
References: Paragraph 4.1.3.1 and 4.2.4.1 discuss this assumption. Paragraph 6.1.2.5.3 discusses removing this assumption from the model.

B.2.4 Continuous Route Clearance

Assumption: Route clearance missions are planned for continuous arc clearance without pausing.
Implication: Path Generation can be modeled as a network flow problem, in which an RCT entering a node in one stage must immediately depart it in the next stage. RCTs also do not drive without clearing arcs.
Justification: In order to make best use of available mission time, RCTs should continuously clear arcs. Sitting idle does not result in IED hazard reduction. Driving without clearing arcs also does not reduce hazard.
Effect if removed: The number of mission paths that are feasible becomes larger. However, it is very unlikely that a heuristic solution that includes RCT pauses at nodes will significantly outperform a similar heuristic solution that does not.
References: Paragraph 4.2.4.1 discusses this assumption. Paragraph 6.1.2.5.2 discusses amending the model to allow RCTs to move without clearing arcs.

B.2.5 Decay of Route Clearance Effects

Assumption: IED Hazard following route clearance of a particular arc quickly resumes its pre-clearance levels.
Implication: Considering only hazard reductions caused by a RCT *during* its mission gives a reasonable approximation of the total hazard reductions resulting from the route clearance mission. In other words, enduring effects of the mission that exist in stages following mission completion can be neglected when comparing potential mission paths.
Justification: We observe the decay of enduring effects in Figures 4-8 and 5-15. Army doctrine acknowledges that routes are not considered "clear" unless they have been constantly secured and observed following route clearance [FM109].
Effect if removed: Path Generation takes longer. The alternate Path Generation heuristic does not rely on this assumption.
References: Paragraph 4.2.4.1.2.7 discusses this assumption.

B.2.6 Independence of IED Detections

Assumption: Arc clearances, by the same RCT or by different RCTs, are *not* independent of each other with respect to the probability of finding an emplaced IED.
Implication: A RCT cannot clear one arc repeatedly, achieving the same magnitude of hazard reduction with each clearance.
Justification: If an IED exists on an arc that is not found by a passing RCT, then it is less likely to be detected on a subsequent pass by the same RCT or by another RCT with similar capabilities. The IED is sufficiently concealed to defeat the RCTs detection capabilities.
Effect if removed: RCTs could clear an arc repeatedly to effectively reduce the hazard to zero. This effect seems unrealistic; there is always the possibility that an IED is concealed so well it defeats the detection capabilities of a RCT.
References: Paragraph 4.2.4.1.1.1 discusses this assumption.

Appendix C: Comparison of Approximate and Exact Path Generation Methods

In this appendix we compare exact and approximate dynamic programming methods, including the primary Path Generation heuristic, for generating 13-stage mission paths on a small, simple network.

C.1 Network and Inputs Summary

We present the results of five trials using the network graph depicted in Figure C-1, a horizon of 13 stages, only one IED type (allowing us to omit IED effectiveness from calculations), and a 100% effective RCT. For these trials, the values for $\lambda_{a,t}$ and $\mu_{a,t}$ are kept constant for over all stages $\{1..\tau\}$, allowing us to drop the "time" subscript, although varying use and emplacement rates would not increase computation effort required. Both rates are measured in occurrences per stage. We do not consider a longer planning horizon, i.e., $\tau = T$, $t_0 = 0$. Node 1 serves as the RCT base, and the values of μ_a for each arc are picked from the range (0.05,0.35). These values remain fixed for all five trials. Different values of λ_a are used in each trial, each coming from a non-standard distribution on the range (0,0.004). To initialize the state probabilities for each arc for each trial, we take the final state probabilities from the previous iteration and carry out 10 transitions according to the Markov chain depicted in Figure 4.3, using the transition probabilities for the new trial. Finally, we set the clearance speed of the RCT so that each arc takes one stage to clear. The values and distribution summaries for the μ and λ parameters for each trial are given in paragraph C.5.

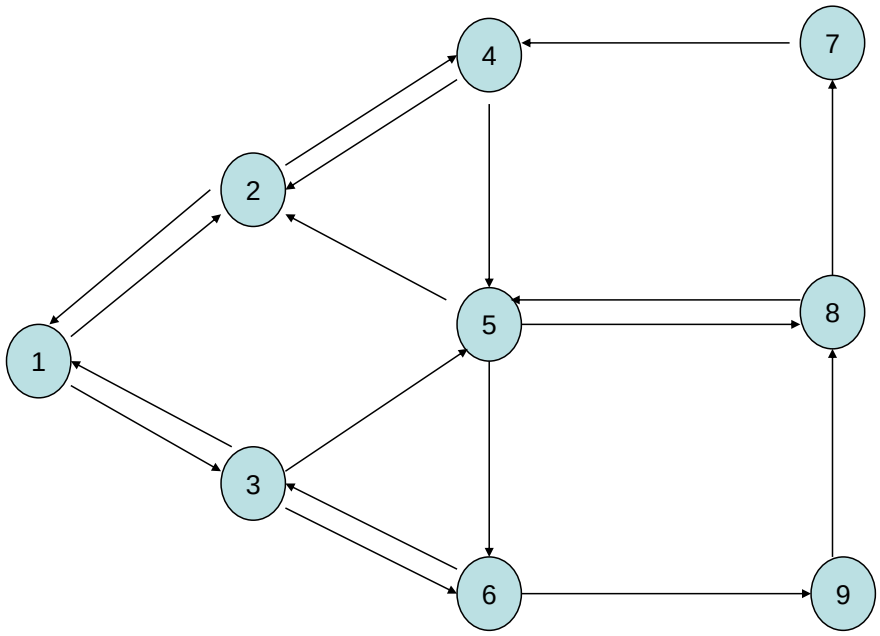

Figure C-1: Path Generation Trial Network

C.2 Methods tested

We test four methods of Path Generation: an exact solution method, a greedy heuristic, a greedy-rollout heuristic, and the label-setting heuristic given as the primary Path Generation heuristic in Chapter 4.

C.2.1 Exact Solution Method

The algorithm we use for finding the exact solution comes directly from the forward dynamic programming method given in Chapter 4, but is simpler because of the simplifying assumptions we have made. The state definition, state transition function, and cost function all remain as defined.

0. Set $J_0(\boldsymbol{x}_0) = 0$,
1. For $t = \{0, .. \tau - 1\}$,
2. Calculate $J_{t+1}(\boldsymbol{x}_{t+1}) = \min_{u_t \in U(x_t)} \left[g_t(\boldsymbol{x}_t, u_t) + J_t\left(f(\boldsymbol{x}_t, u_t)\right) \right], \forall \boldsymbol{x}_{t+1} \in S^{t+1}$
3. Set $\varphi_{t+1}(\boldsymbol{x}_{t+1}) = \text{argmin}_{u_t \in U(x_t)} \left[g_t(\boldsymbol{x}_t, u_t) + J_t\left(f(\boldsymbol{x}_t, u_t)\right) \right]$

4. Select the state $\boldsymbol{x}_\tau^*$ for which $J_\tau(\boldsymbol{x}_\tau^*) \leq J_\tau(\boldsymbol{x}_\tau), \forall \boldsymbol{x}_\tau \in S^\tau$, as well as its corresponding mission path as the optimal solution

As discussed in Chapter 4 and depicted in Figure 4-6, steps 1a and 1b do not involve minimization over more than one value because for each state in S^{t+1}, there is only one state in S^t from which transition is possible.

C.2.2 Greedy Algorithm

The greedy algorithm easy to implement. We begin at the origin and at each stage select the feasible arc that results in the lowest cost for that stage. This heuristic requires relatively little computational effort with cost calculations for at most $n \times T$ states.

C.2.3 Greedy-Rollout Algorithm

We apply the rollout algorithm [Ber05] (one-step look ahead) to the greedy heuristic. We use the same state space, cost-per-stage function, and transition function as defined in the original dynamic programming formulation, but we define

$$\bar{J}_t(\boldsymbol{x}_t) := \{\text{The cost of the greedy heuristic from state } \boldsymbol{x}_t \text{ to the destination node}\}$$

for any state $\boldsymbol{x}_t \in S^t$. Beginning from $\boldsymbol{x}_0 = (1, \boldsymbol{H}_0)$, we successively choose policies that minimize the current stage cost plus the cost-to-go approximation given by the greedy algorithm, i.e., we iterate for $t = \{0..\tau - 1\}$:

1. $\varphi_t(\boldsymbol{x}_t) = u_t^* = \text{argmin}_{u_t \in U(\boldsymbol{x}_t)}\left[g_t(\boldsymbol{x}_t, u_t) + \bar{J}_{t+1}\left(f(\boldsymbol{x}_t, u_t)\right)\right]$
2. $\boldsymbol{x}_{t+1} = f\left(\boldsymbol{x}_t, \mu_t(\boldsymbol{x}_t)\right)$

This algorithm returns a path $(\varphi_0, \varphi_1, .. \varphi_{\tau-1})$, with the cost computed at each iteration.

Because the costs resulting from the greedy heuristic satisfy the criteria

$$\min_{u_t \in U(\boldsymbol{x}_t)}\left[g_t(\boldsymbol{x}_t, u_t) + \bar{J}_t\left(f(\boldsymbol{x}_t, u_t)\right)\right] \leq \bar{J}_t(\boldsymbol{x}_t),$$

we know the greedy-rollout algorithm will perform at least as well as the greedy algorithm [Ber05]. We also expect it to take longer, as it must run the greedy algorithm up to *n* times for each stage.

C.2.4 Label-Setting Heuristic

The heuristic used in these trials is the Path Generation heuristic described in detail in section 4.2.4.1.4 of this thesis.

C.3 Results

Figure C-2 shows a plot of the costs of each of the above methods for each trial.

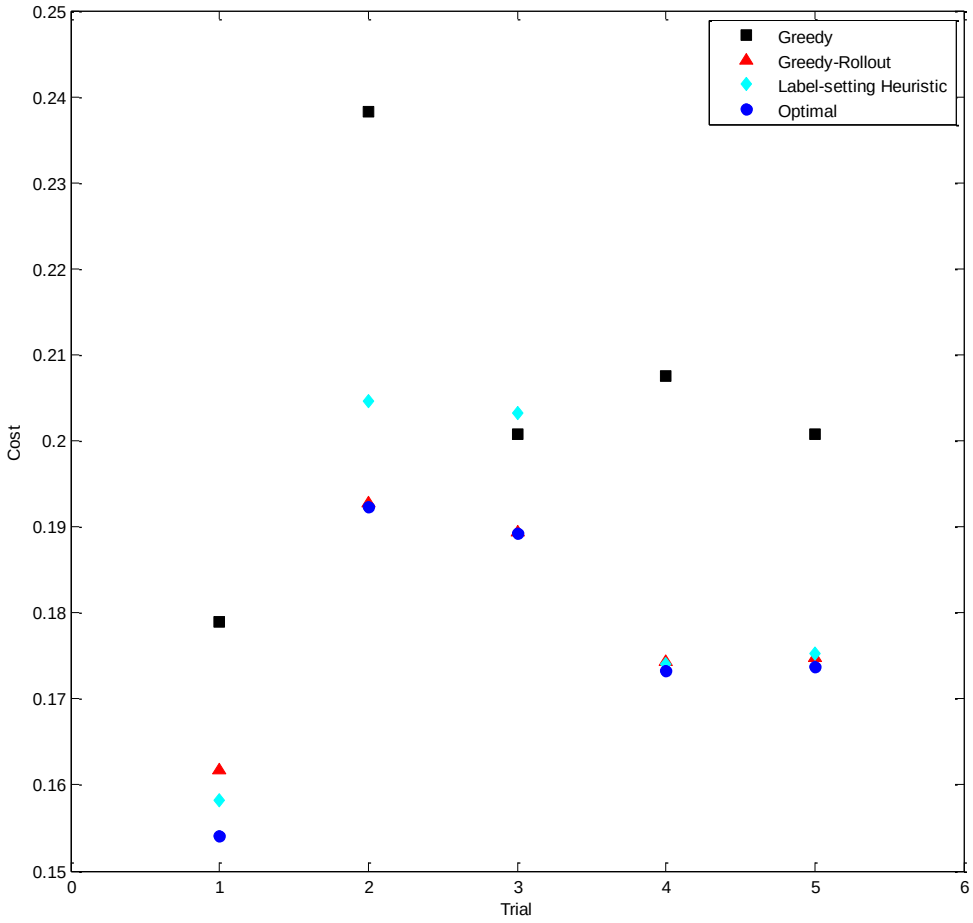

Figure C-2: Performance Results

One can see from the plot that for these trials, the greedy-rollout heuristic performs very close to the optimal solution in most cases. In trial 1, the label-setting heuristic outperforms the greedy-rollout heuristic, but in the other four cases the opposite is true. The greedy heuristic performs the worst in all cases except the third trial, in which the label-setting heuristic returns the highest cost of the four methods.

The running time for each of the heuristics are summarized below in Table C-1. The optimal solution required cost computations for 5057 unique paths on the small network. Included in the table are also the running times for each of the heuristics on the larger Cambridge-based road network described in Chapter 5 using the same assumptions made at the beginning of this appendix (RCT base at node 1, one stage required to clear each arc, and 100% IED clearance rate).

Running Times (seconds)		
Algorithm	Small Network: 13 stages	Cambridge-based Network: 72 stages
Greedy	0.022	0.308
Greedy-Rollout	0.165	37.614
Heuristic	0.28	2.04
Optimal	55.303	---

Table C-1. Running times

C.4 Conclusions

The five trials on the small network reveal that small changes in the problem's parameters do impact the performance of each, and that no immediate general conclusions can be drawn about the label-setting heuristic's performance with respect to either of the other two. It does appear that the label-setting heuristic is more consistent than the greedy heuristic, which performs much worse than any of the other heuristics in some cases, as in the second trial. From the computation times, we find that the exact solution is intractable in the 72-stage problem. The greedy heuristic is the fastest, followed by the label-setting heuristic. The greedy rollout algorithm is fast for small problems but slows down as more stages are added because it runs the greedy algorithm for each state in each stage.

We conclude that the label-setting heuristic is a good choice for generating paths in a route clearance planning algorithm because of its consistency and its computation time.

C.5 Data Inputs

C.5.1 Summary figures

C.5.1.1 Emplacement Rates

The following color-coded figures depict the distribution of IED emplacement rates on the arcs of the network shown in Figure C-1.

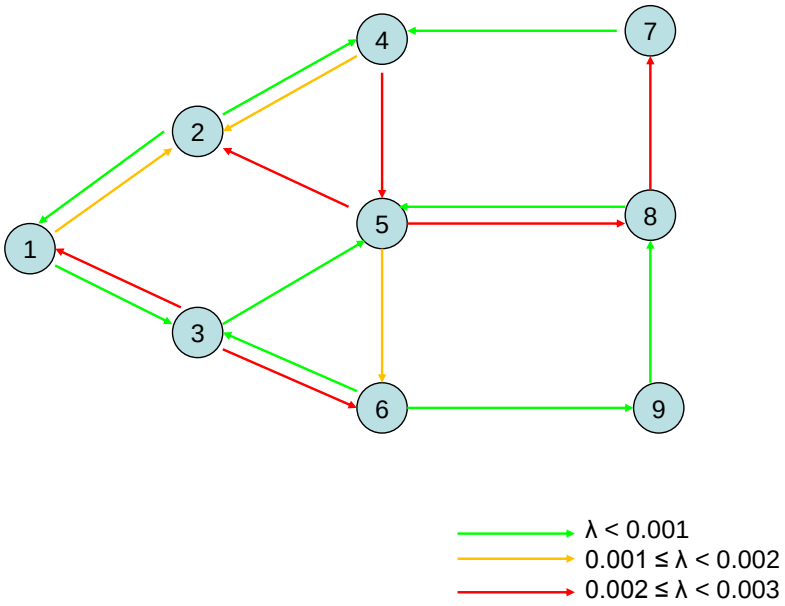

Figure C-3: Emplacement Rates for Trial 1

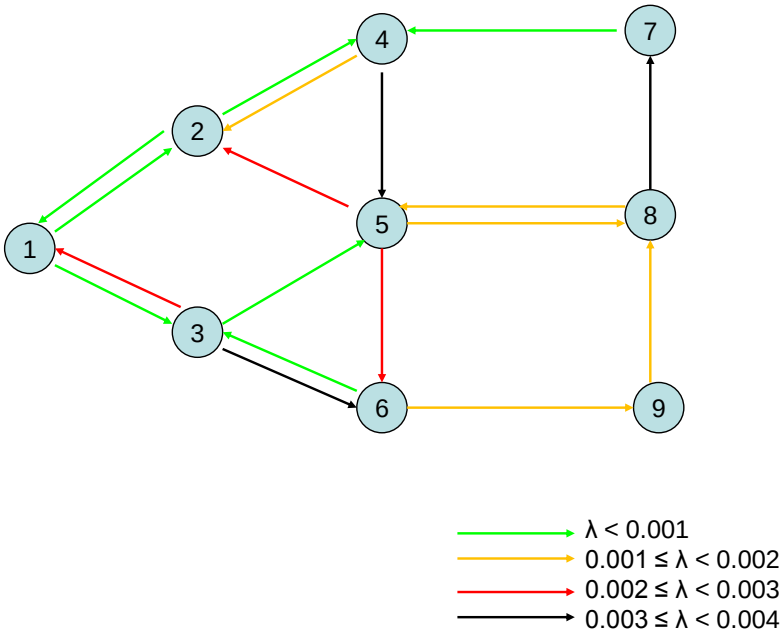

Figure C-4: Emplacement Rates for Trial 2

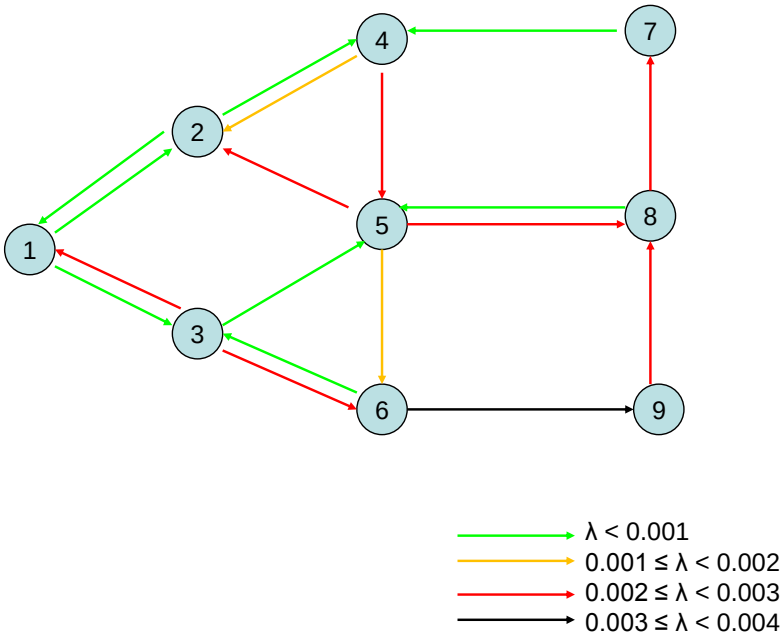

Figure C-5: Emplacement Rates for Trial 3

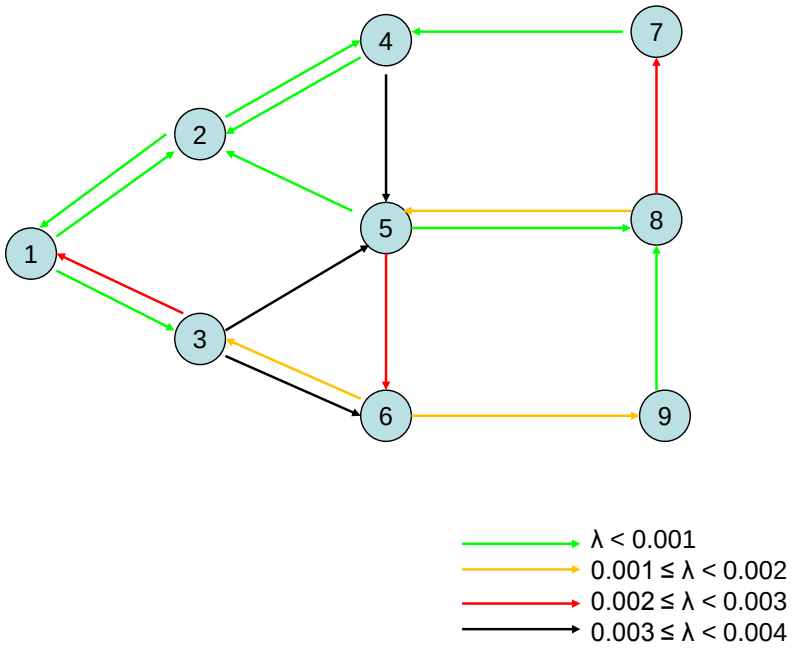

Figure C-6: Emplacement Rates for Trial 4

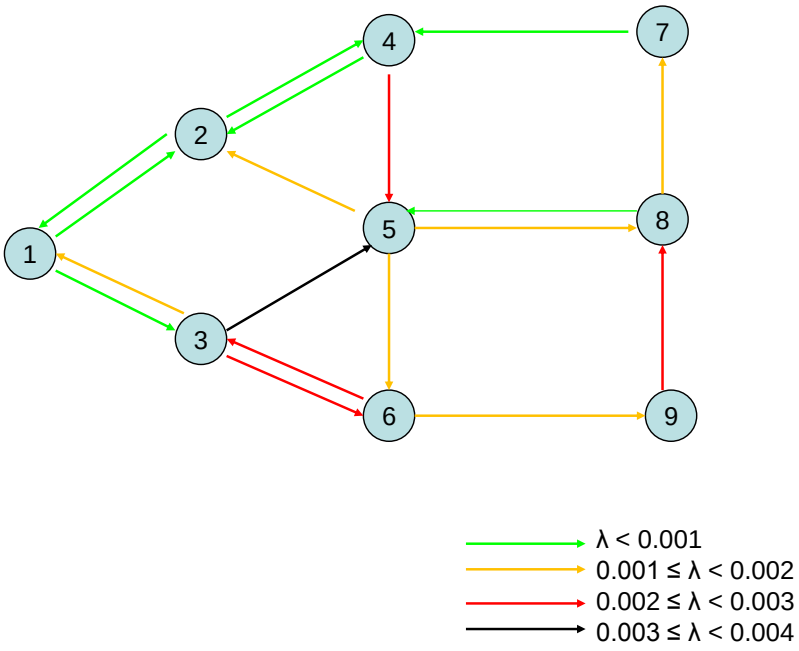

Figure C-7: Emplacement Rates for Trial 5

Use Rates.

Figure C-8 shows the distribution of arc usage, which remains constant for all five trials.

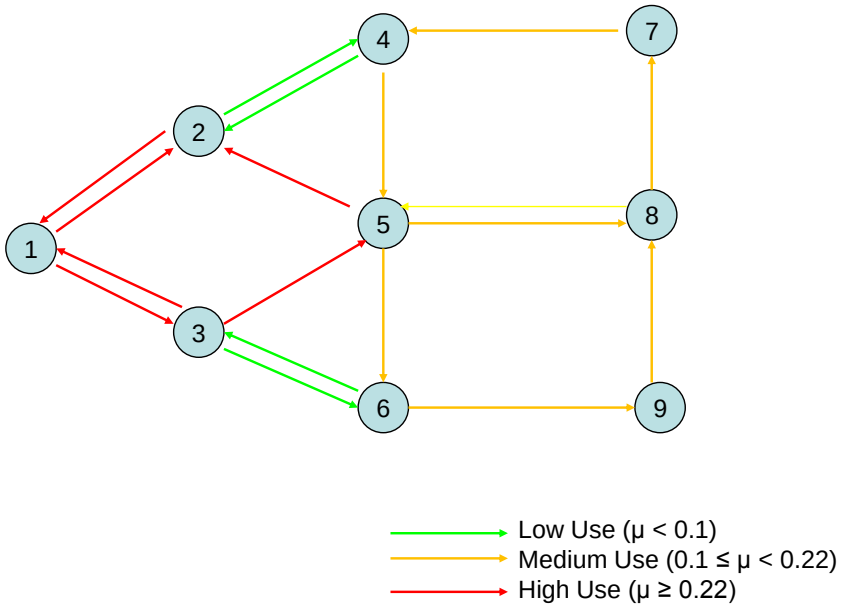

Figure C-8: Use Rates

C.5.2 Tables of Parameter Values

C.5.2.1 Emplacement Rates

Table C-2 shows the values for λ_a used in each trial.

Arc	Trial 1	Trial 2	Trial 3	Trial 4	Trial 5
(2, 1)	0.0005	0.0006	0.0007	0.0008	0.0006
(3, 1)	0.0022	0.0029	0.0021	0.0026	0.0017
(1, 2)	0.0010	0.0001	0.0000	0.0003	0.0002
(4, 2)	0.0018	0.0019	0.0019	0.0006	0.0009
(5, 2)	0.0021	0.0022	0.0021	0.0007	0.0010
(1, 3)	0.0000	0.0000	0.0001	0.0001	0.0002
(6, 3)	0.0001	0.0003	0.0005	0.0010	0.0021
(2, 4)	0.0003	0.0001	0.0004	0.0005	0.0006
(7, 4)	0.0003	0.0007	0.0006	0.0006	0.0002
(3, 5)	0.0000	0.0001	0.0001	0.0035	0.0032
(4, 5)	0.0029	0.0040	0.0027	0.0034	0.0022
(8, 5)	0.0008	0.0011	0.0009	0.0013	0.0009
(3, 6)	0.0023	0.0031	0.0025	0.0036	0.0024
(5, 6)	0.0016	0.0022	0.0016	0.0021	0.0014
(8, 7)	0.0025	0.0034	0.0022	0.0025	0.0017
(5, 8)	0.0021	0.0018	0.0021	0.0007	0.0010
(9, 8)	0.0007	0.0015	0.0030	0.0003	0.0022
(6, 9)	0.0009	0.0018	0.0037	0.0016	0.0040

Table C-2. Emplacement Rates

C.5.2.2 Use Rates

Table C-3 shows the values for μ_a used in each trial.

Arc	All Trials
(2, 1)	0.2212
(3, 1)	0.2212
(1, 2)	0.2212
(4, 2)	0.0408
(5, 2)	0.2212
(1, 3)	0.2212
(6, 3)	0.0408
(2, 4)	0.0408
(7, 4)	0.1535
(3, 5)	0.2212
(4, 5)	0.1535
(8, 5)	0.1535
(3, 6)	0.0408
(5, 6)	0.1535
(8, 7)	0.1535
(5, 8)	0.1535
(9, 8)	0.1535
(6, 9)	0.1535
(2, 1)	0.2212

Table C-3. Use Rates

[This Page Intentionally Left Blank]

Appendix D: Experimental Data Sets

This appendix contains summaries of all data sets used for experimentation in this thesis.

D.1 Experiment 1 Data

D.1.1 Fixed Parameters

The following parameters are fixed for all tests in Experiment 1.

Parameter	Definition	Value
K	Number of IEDs	2
N	Number of nodes	58
A	Number of arcs	162
D	Planning Horizon	1 day
δ	Discrete Time Step	5 minutes
T	Number of Stages	360
$\overline{h}$	Number of RCTs	2
$P\_0_{a,k}$	Initial state probabilities	0 for all a,k

D.1.1.1 Cambridge-Based Road Network

The Cambridge-based road network is used for all tests in Experiment 1. The graph is shown in Figures 5-1 and D-1. Table D-1 contains a list of all arcs with their start and end nodes, lengths, and classifications.

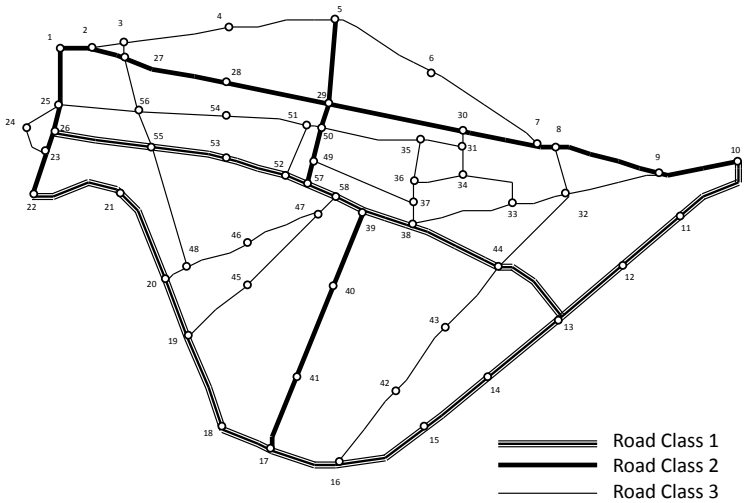

Figure D-1: Cambridge-Based Road Network

D.1.2 Data Set 11

D.1.2.1 IED Effectiveness Parameters

IED Type Index (k)	$P\_effects_k$
1	0.15
2	0.7

D.1.2.2 Counter-Insurgent Modeling Parameters

D.1.2.2.1 RCT Parameters

RCT Index (h)	RCT_type_h	T_min_h (h)	T_max_h (h)	$dwell_h$ (h)	Max_day_h
1	1	5	6	2	2
2	2	4	5	3	2

D.1.2.2.2 RCT Availability

RCT Index (h)	$Avail\_start\_0_{h,1}$	$Avail\_start\_1_{h,1}$
1	0	24
2	0	24

D.1.2.3 Insurgent—Terrain Interaction Parameters

Figures D-2 and D-3 show the emplacement rates for IED types 1 and 2, respectively, as a function of time. The arcs in the Cambridge-based road network graphs are color-coded to match their corresponding emplacement rate plots.

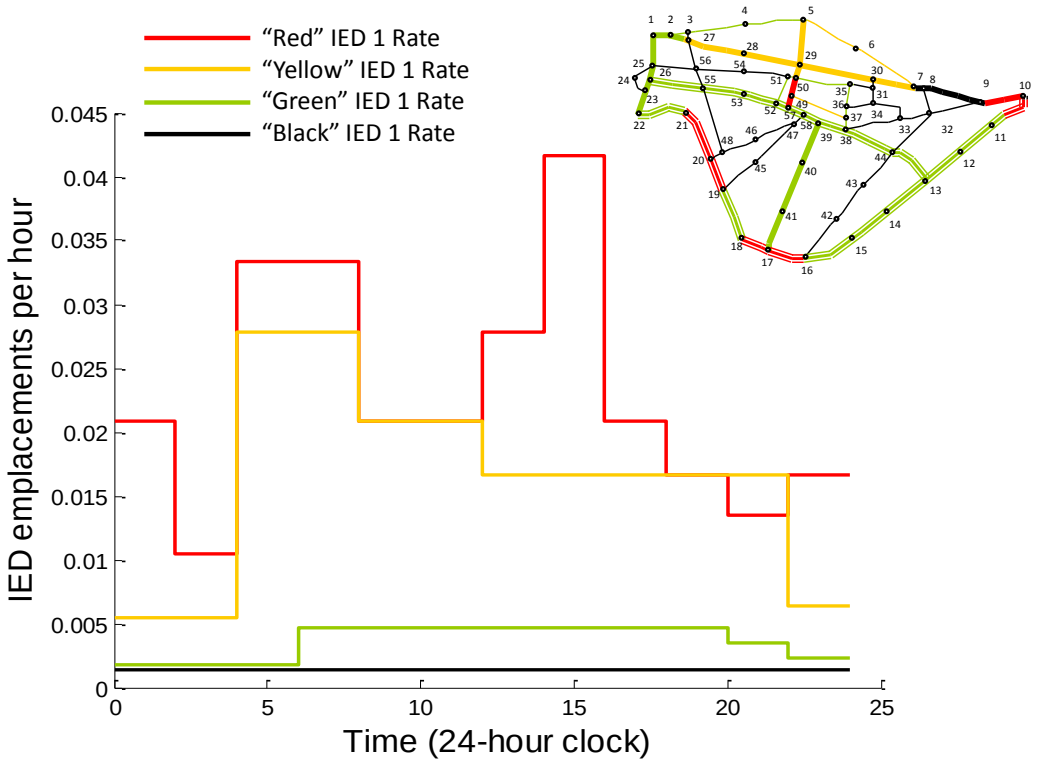

Figure D-2: Experiment 1, Data Set 1, IED 1 Emplacement Rates

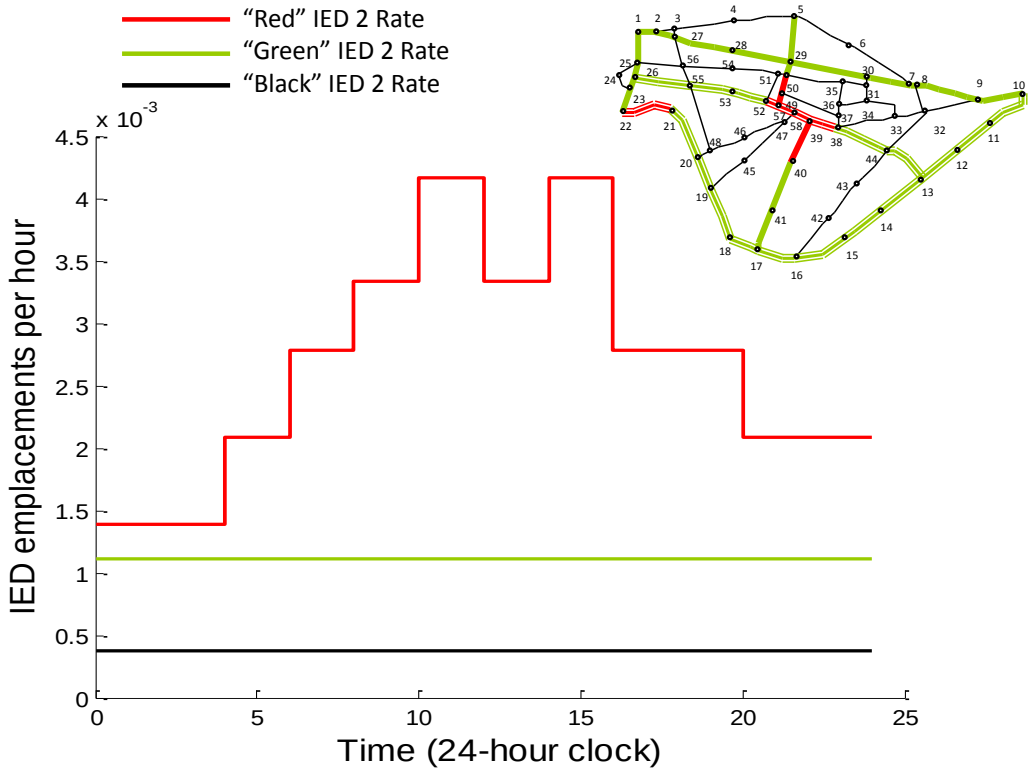

Figure D-3: Experiment 1, Data Set 1, IED 2 Emplacement Rates

D.1.2.4 Counter-Insurgent—Terrain Interaction Parameters

D.1.2.4.1 RCT Locations

RCT Index (h)	$base_h$
1	32
2	48

D.1.2.4.2 RCT Detection Probabilities

D.1.2.4.2.1 RCT Configuration 1

IED type (k)	$P\_detect_{1,k,1}$	$P\_detect_{1,k,2}$	$P\_detect_{1,k,3}$
1	0.9	0.85	0.85
2	0.35	0.25	0.15

D.1.2.4.2.2 RCT Configuration 2

IED type (k)	$P\_detect_{2,k,1}$	$P\_detect_{2,k,2}$	$P\_detect_{2,k,3}$
1	0.9	0.85	0.85
2	0.9	0.85	0.75

D.1.2.4.3 RCT Clearance Speeds

Configuration Type (q)	$V_{RCT\_typeh,1}$	$V_{RCT\_typeh,2}$	$V_{RCT\_typeh,3}$
1	1	0.75	0.5
2	0.9	0.65	0.5

D.1.2.4.4 Arc Use Rates

Figure D-4 displays arc use rates as a function of time. The arcs in the Cambridge-based road network graph are color-coded to match their corresponding use rate plots.

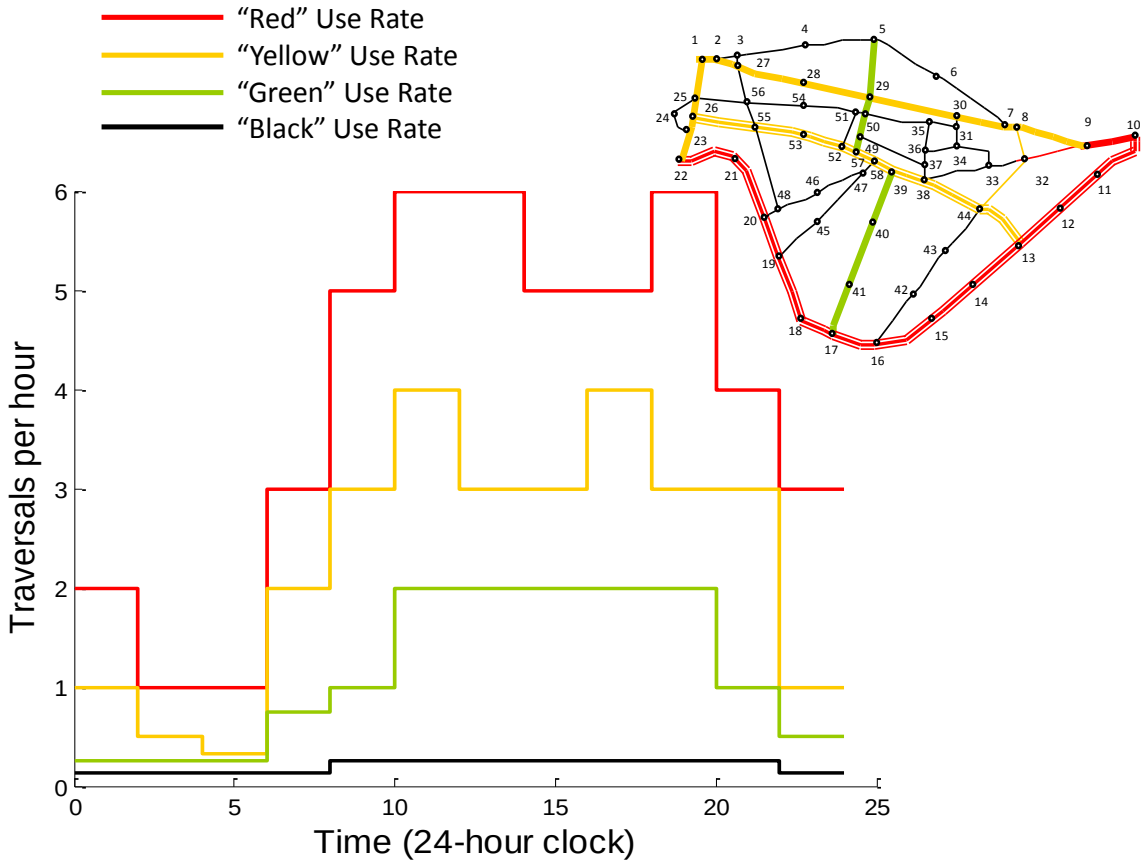

Figure D-4: Experiment 1, Data Set 1 Arc Use Rates

D.1.3 Data Set 12

D.1.3.1 IED Effectiveness Parameters

IED Type Index (k)	$P\_effects_k$
1	0.1
2	0.85

D.1.3.2 Counter-Insurgent Modeling Parameters

D.1.3.2.1 RCT Parameters

RCT Index (h)	RCT_type_h	T_min_h (h)	T_max_h (h)	$dwell_h$ (h)	Max_day_h
1	1	5	6	2	2
2	2	4	5	3	2

D.1.3.2.2 RCT Availability

RCT Index (h)	$Avail\_start\_0_{h,1}$	$Avail\_start\_1_{h,1}$
1	0	24
2	0	24

D.1.3.3 Insurgent—Terrain Interaction Parameters

Figures D-5 and D-6 show the emplacement rates for IED types 1 and 2, respectively, as a function of time. The arcs in the Cambridge-based road network graphs are color-coded to match their corresponding emplacement rate plots.

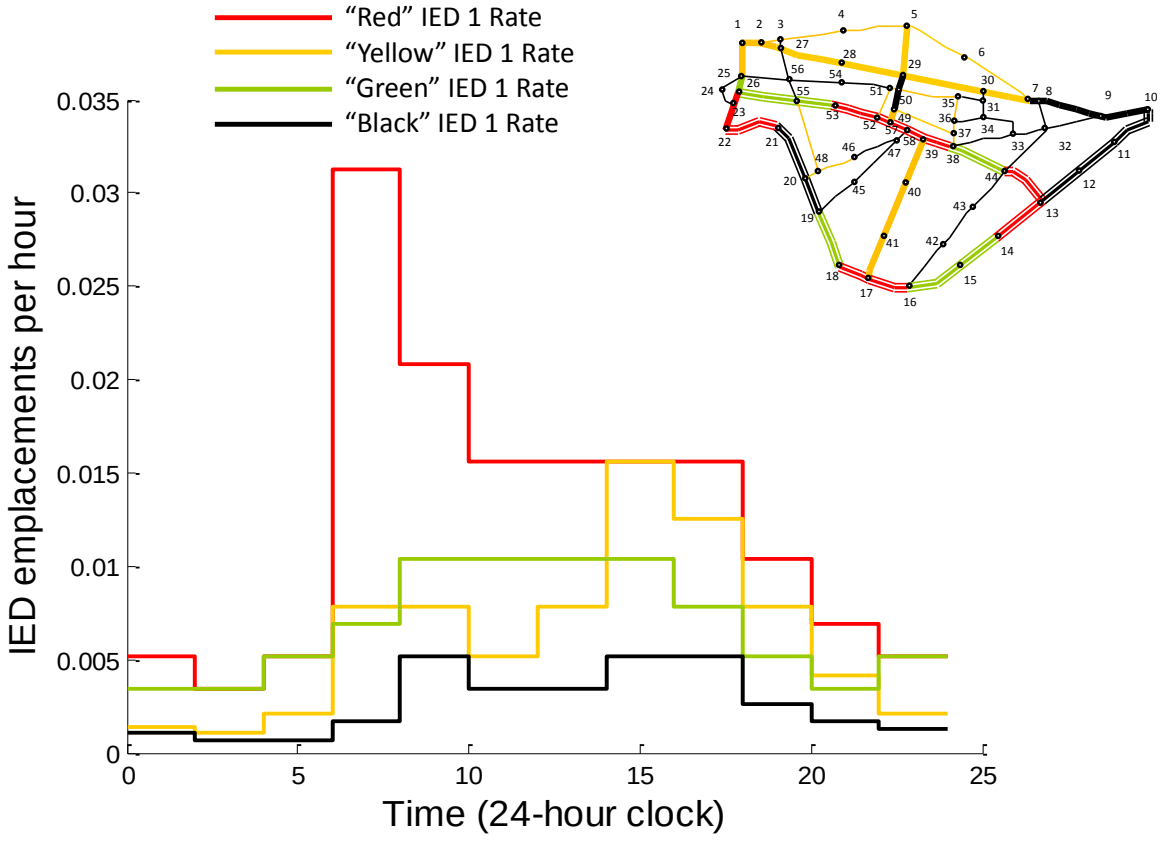

Figure D-5: Experiment 1, Data Set 2, IED 1 Emplacement Rates

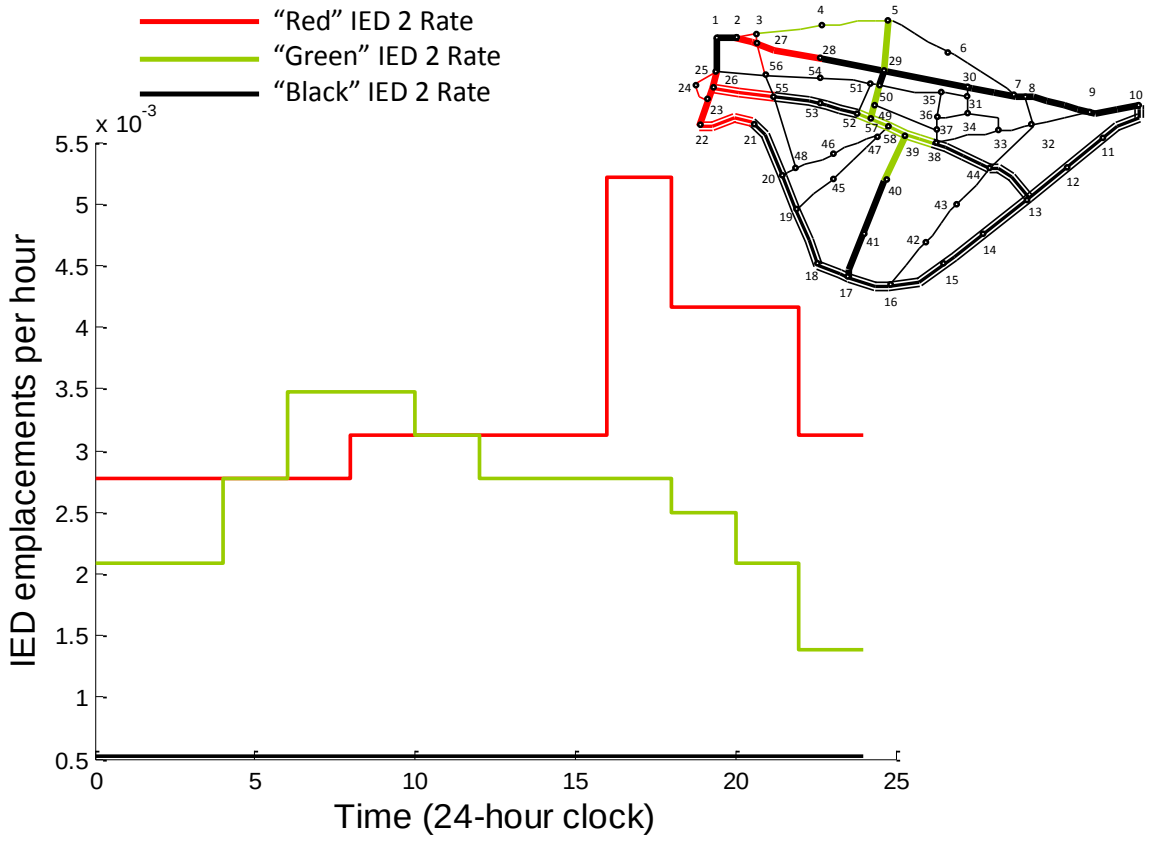

Figure D-6: Experiment 1, Data Set 2, IED 2 Emplacement Rates

D.1.3.4 Counter-Insurgent—Terrain Interaction Parameters

D.1.3.4.1 RCT Locations

RCT Index (h)	$base_h$
1	32
2	48

D.1.3.4.2 RCT Detection Probabilities

D.1.3.4.2.1 RCT Configuration 1

IED type (k)	$P\_detect_{1,k,1}$	$P\_detect_{1,k,2}$	$P\_detect_{1,k,3}$
1	0.9	0.85	0.85
2	0.35	0.25	0.15

D.1.3.4.2.2 RCT Configuration 2

IED type (k)	$P\_detect_{2,k,1}$	$P\_detect_{2,k,2}$	$P\_detect_{2,k,3}$
1	0.9	0.85	0.85
2	0.9	0.85	0.75

D.1.3.4.3 RCT Clearance Speeds

Configuration Type (q)	$V_{RCT\_typeh,1}$	$V_{RCT\_typeh,2}$	$V_{RCT\_typeh,3}$
1	1	0.75	0.5
2	0.9	0.65	0.5

D.1.3.4.4 Arc Use Rates

Figure D-7 displays arc use rates as a function of time. The arcs in the Cambridge-based road network graph are color-coded to match their corresponding use rate plots.

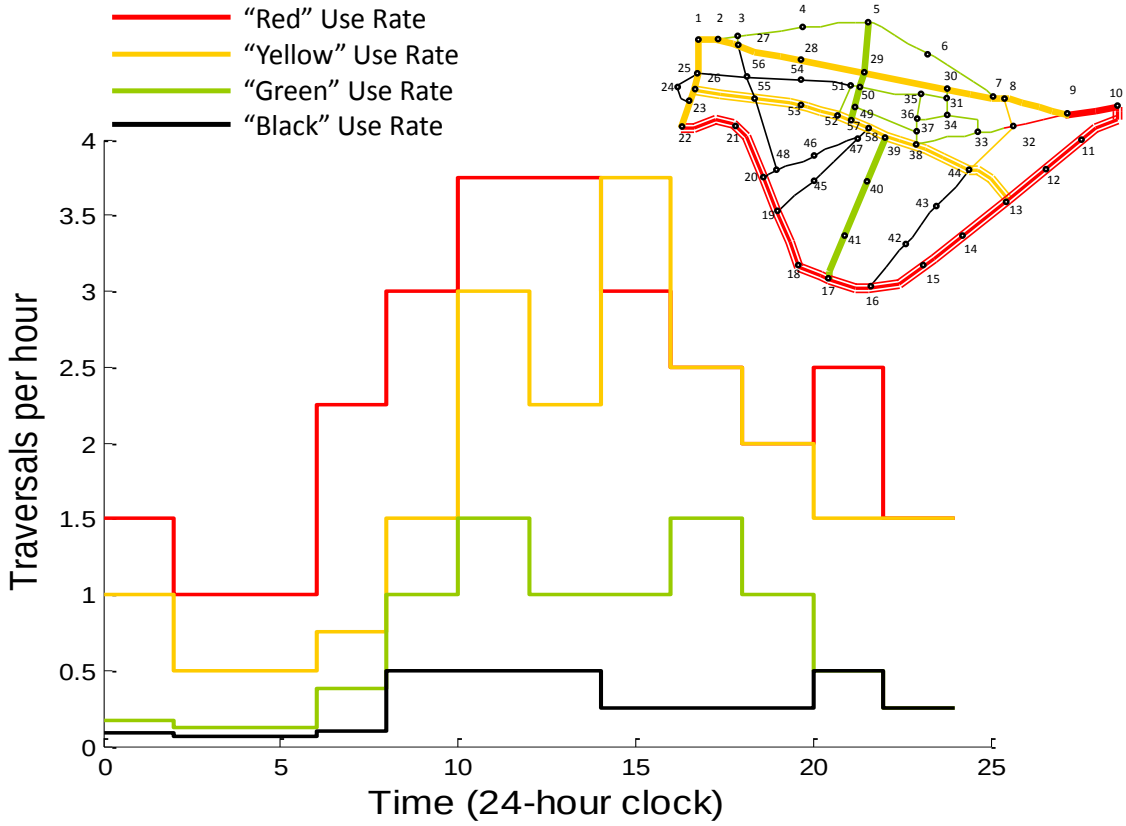

Figure D-7: Experiment 1, Data Set 2 Arc Use Rates

D.1.4 Data Set 13

All inputs in Data Set 13 are the same as Data Set 12, with the exception of IED emplacement rates and arc usage rates, which are given below.

D.1.4.1 IED Emplacement Rates

Figures D-8 and D-9 show the emplacement rates for IED types 1 and 2, respectively, as a function of time. The arcs in the Cambridge-based road network graphs are color-coded to match their corresponding emplacement rate plots.

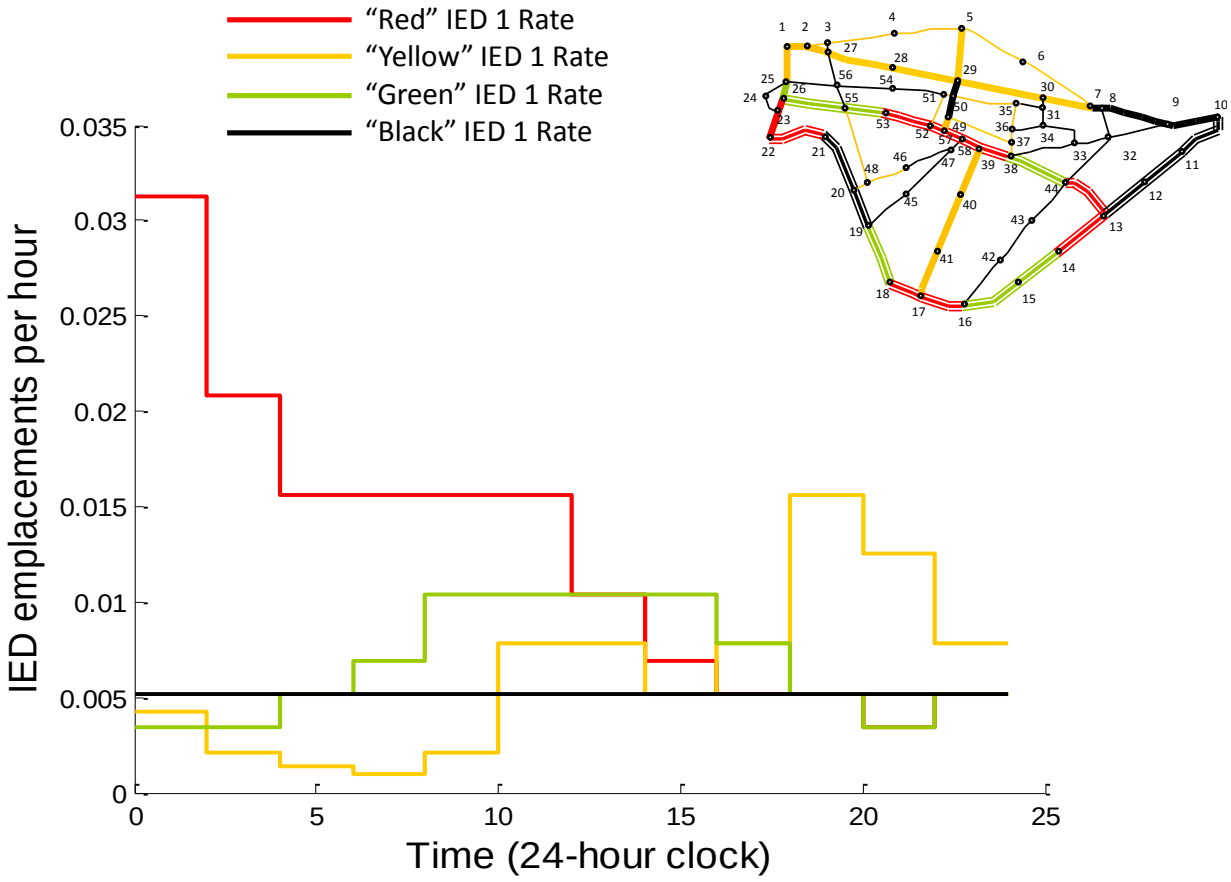

Figure D-8: Experiment 1, Data Set 3, IED 1 Emplacement Rates

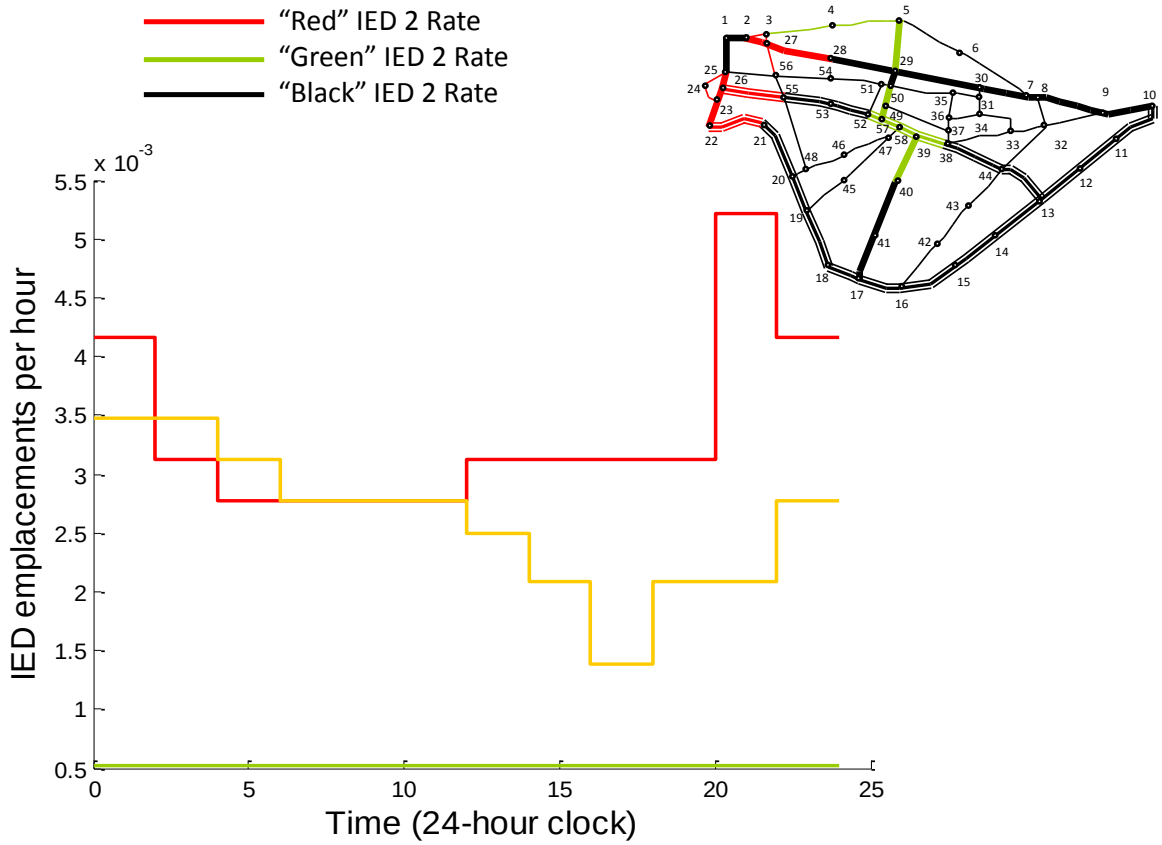

Figure D-9: Experiment 1, Data Set 3, IED 2 Emplacement Rates

D.1.4.2 Arc Use Rates

Figure D-10 displays arc use rates as a function of time. The arcs in the Cambridge-based road network graph are color-coded to match their corresponding use rate plots.

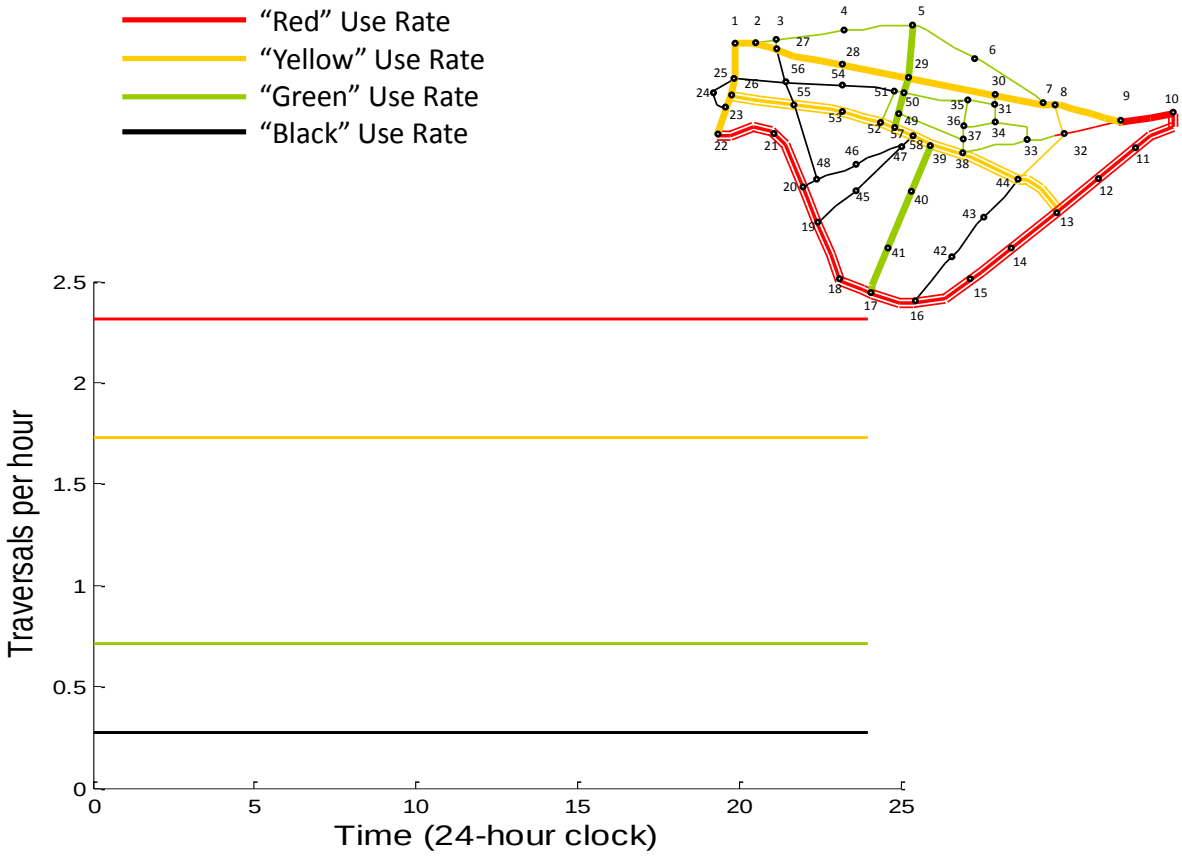

Figure D-10: Experiment 1, Data Sets 3 and 5 Arc Use Rates

D.1.5 Data Set 14

All inputs in Data Set 14 are the same as Data Set 12, with the exception of IED emplacement rates and arc usage rates, which are given below.

D.1.5.1 IED Emplacement Rates

Figures D-11 and D-12 show the emplacement rates for IED types 1 and 2, respectively, as a function of time. The arcs in the Cambridge-based road network graphs are color-coded to match their corresponding emplacement rate plots.

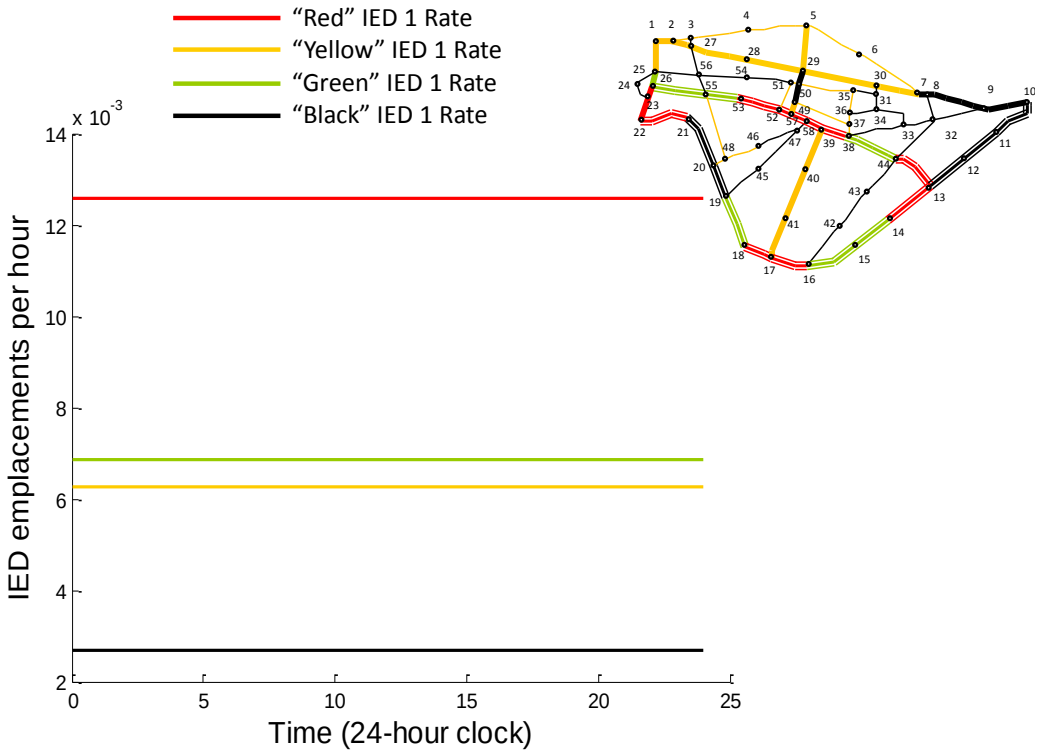

Figure D-11: Experiment 1, Data Sets 4 and 5, IED 1 Emplacement Rates

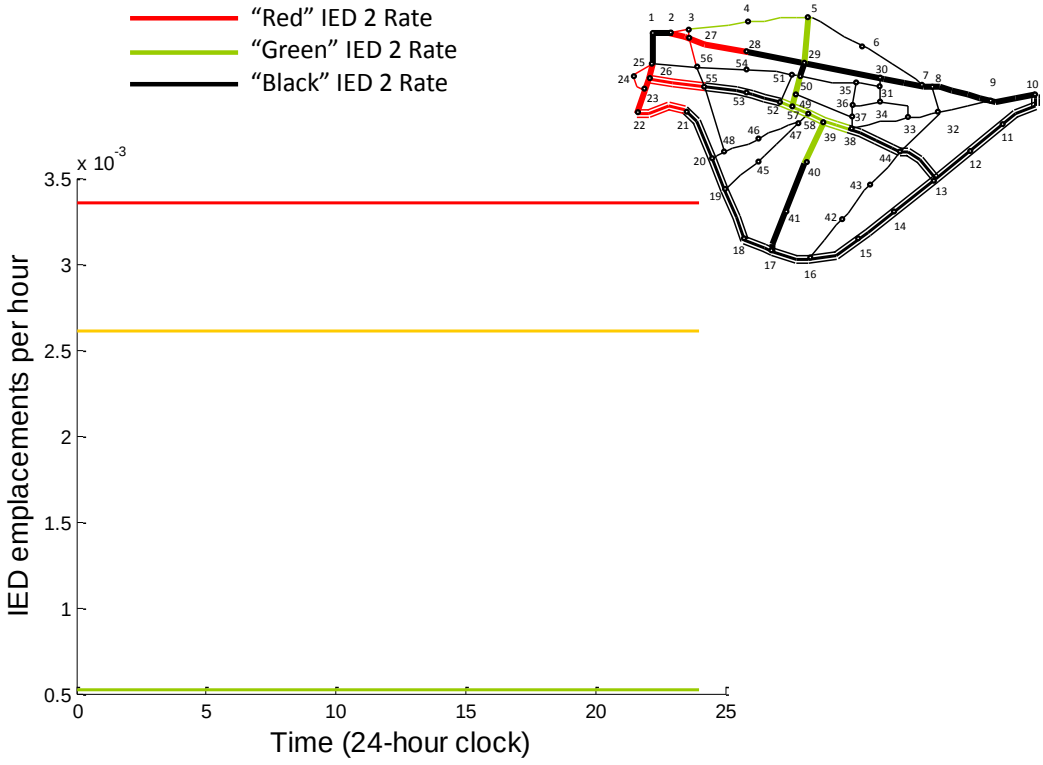

Figure D-12: Experiment 1, Data Sets 4 and 5, IED 2 Emplacement Rates

D.1.5.2 Arc Use Rates

Figure D-13 displays arc use rates as a function of time. The arcs in the Cambridge-based road network graph are color-coded to match their corresponding use rate plots.

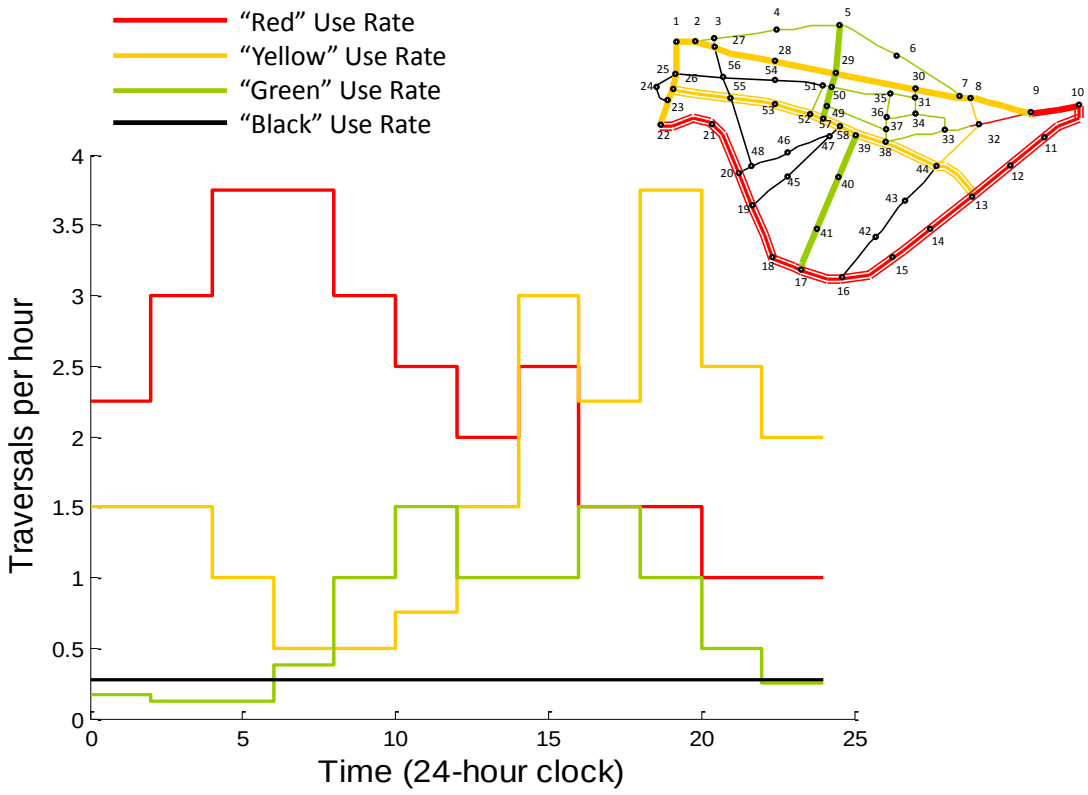

Figure D-13: Experiment 1, Data Set 4 Arc Use Rates

D.1.6 Data Set 15

All inputs in Data Set 15 are the same as Data Set 12, with the exception of IED emplacement rates. The IED emplacement rates for Data Set 15 are the same as those for Data Set 14, given in Figures D-11 and D-12. The arc usage rates for Data Set 15 are the same as those for Data Set 13, given in Figure D-10.

D.1.7 Data Set 16

All inputs in Data Set 16 are the same as Data Set 12, with the exception of IED emplacement rates and arc usage rates, which are given below.

D.1.7.1 IED Emplacement Rates

Figures D-14 and D-15 show the emplacement rates for IED types 1 and 2, respectively, as a function of time. The arcs in the Cambridge-based road network graphs are color-coded to match their corresponding emplacement rate plots.

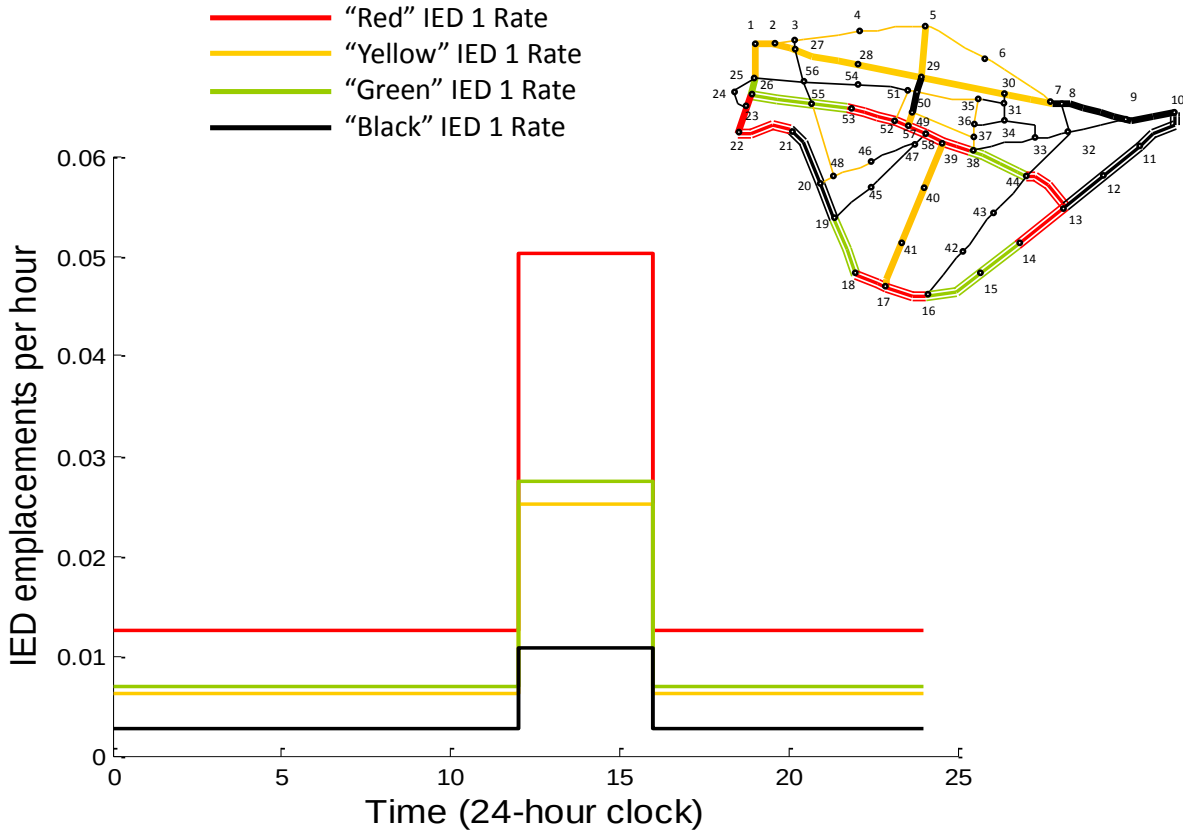

Figure D-14: Experiment 1, Data Set 6, IED 1 Emplacement Rates

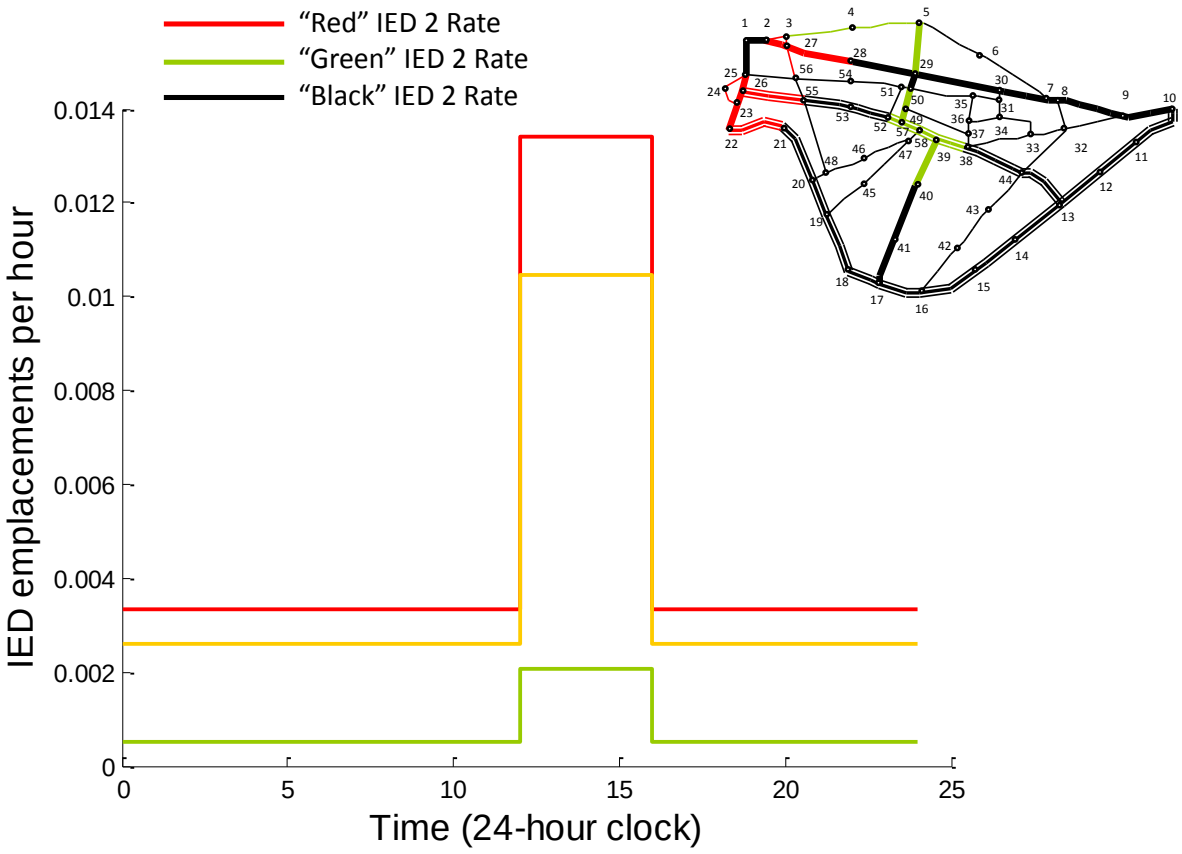

Figure D-15: Experiment 1, Data Set 6, IED 2 Emplacement Rates

D.1.7.2 Arc Use Rates

Figure D-16 displays arc use rates as a function of time. The arcs in the Cambridge-based road network graph are color-coded to match their corresponding use rate plots.

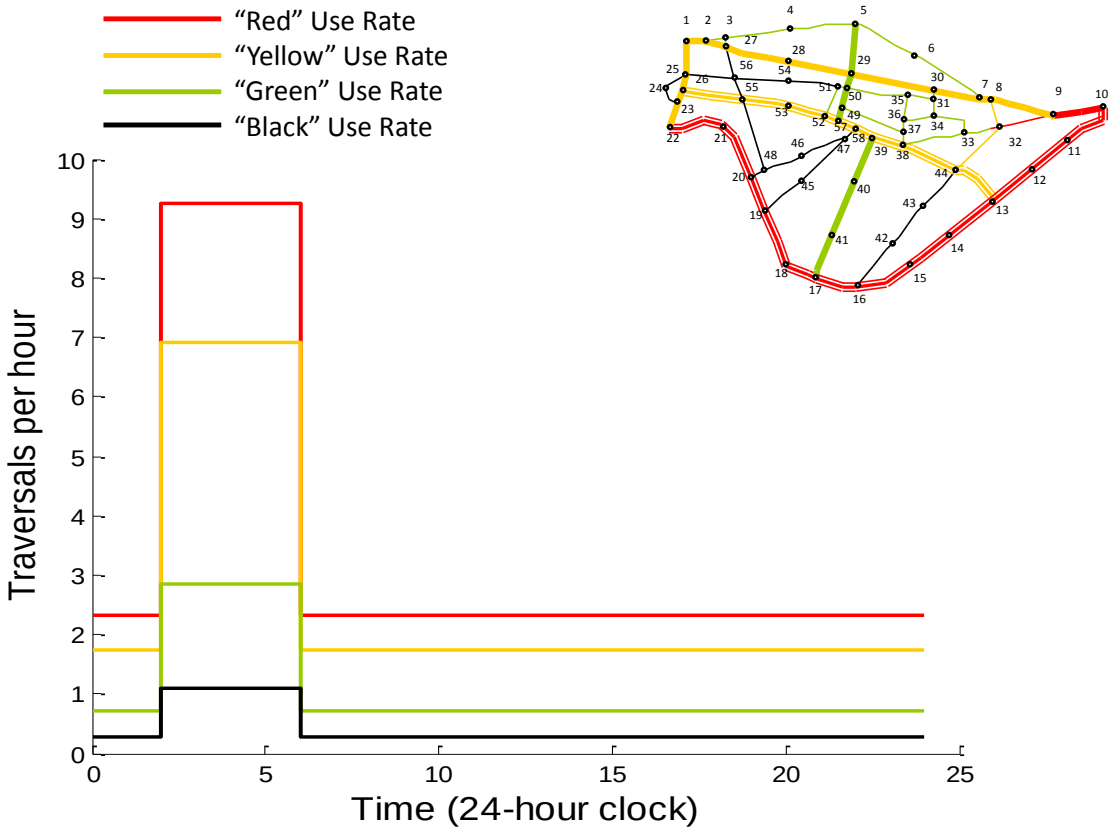

Figure D-16: Experiment 1, Data Set 6 Arc Use Rates

Arc Index	Start Node	End Node	Length	Class
1	2	1	7	2
2	25	1	9	2
3	1	2	7	2
4	3	2	7	3
5	27	2	7	2
6	2	3	7	3
7	4	3	11	3
8	27	3	5	3
9	3	4	11	3
10	5	4	11	3
11	4	5	11	3
12	6	5	11	3
13	29	5	11	2
14	5	6	11	3
15	7	6	13	3
16	6	7	13	3
17	8	7	5	2
18	30	7	9	2
19	7	8	5	2
20	9	8	11	2
21	32	8	7	3
22	8	9	11	2
23	10	9	9	2
25	9	10	9	2
26	11	10	11	1
27	10	11	11	1
28	12	11	9	1
29	11	12	9	1
30	13	12	9	1
31	12	13	9	1
32	14	13	11	1
33	44	13	11	1
34	13	14	11	1
35	15	14	9	1
36	14	15	9	1
37	16	15	11	1
38	15	16	11	1
39	17	16	9	1
40	42	16	11	3
41	16	17	9	1
42	18	17	7	1
43	41	17	9	2
44	17	18	7	1

Table D-1: Cambridge-Based Road Network Arc Data

Arc Index	Start Node	End Node	Length	Class
45	19	18	11	1
46	18	19	11	1
47	20	19	9	1
48	45	19	9	3
49	19	20	9	1
50	21	20	11	1
51	48	20	5	3
52	20	21	11	1
53	22	21	11	1
54	21	22	11	1
55	23	22	7	2
56	22	23	7	2
57	24	23	7	3
58	26	23	5	2
59	23	24	7	3
60	25	24	7	3
61	1	25	9	2
62	24	25	7	3
63	26	25	5	2
64	56	25	9	3
65	23	26	5	2
66	25	26	5	2
67	55	26	11	1
68	2	27	7	2
69	3	27	5	3
70	28	27	11	2
71	56	27	9	3
72	27	28	11	2
73	29	28	11	2
74	5	29	11	2
75	28	29	11	2
76	30	29	13	2
77	50	29	7	2
78	7	30	9	2
79	29	30	13	2
80	31	30	5	3
81	30	31	5	3
82	34	31	7	3
83	35	31	7	3
84	8	32	7	3
85	9	32	9	3
86	33	32	9	3
87	44	32	11	3

Table D-1: Cambridge-Based Road Network Arc Data (Continued)

Arc Index	Start Node	End Node	Length	Class
88	32	33	9	3
89	34	33	9	3
90	38	33	11	3
91	31	34	7	3
92	33	34	9	3
93	36	34	7	3
94	31	35	7	3
95	36	35	7	3
96	50	35	11	3
97	34	36	7	3
98	35	36	7	3
99	37	36	5	3
100	36	37	5	3
101	38	37	5	3
102	49	37	11	3
103	33	38	11	3
104	37	38	5	3
105	39	38	7	1
106	44	38	9	1
107	38	39	7	1
108	40	39	9	2
109	58	39	5	1
110	39	40	9	2
111	41	40	11	2
112	17	41	9	2
113	40	41	11	2
114	16	42	11	3
115	43	42	9	3
116	42	43	9	3
117	44	43	9	3
118	13	44	11	1
119	32	44	11	3
120	38	44	9	1
121	43	44	9	3
122	19	45	9	3
123	47	45	11	3
124	47	46	9	3
125	48	46	9	3
126	45	47	11	3
127	46	47	9	3
128	58	47	5	3
129	20	48	5	3
130	46	48	9	3

Table D-1: Cambridge-Based Road Network Arc Data (Continued)

Arc Index	Start Node	End Node	Length	Class
131	55	48	13	3
132	37	49	11	3
133	50	49	7	2
134	57	49	5	2
135	29	50	7	2
136	35	50	11	3
137	49	50	7	2
138	51	50	5	3
139	50	51	5	3
140	52	51	9	3
141	54	51	11	3
142	51	52	9	3
143	53	52	9	1
144	57	52	5	1
145	52	53	9	1
146	55	53	9	1
147	51	54	11	3
148	56	54	9	3
149	26	55	11	1
150	48	55	13	3
151	53	55	9	1
152	56	55	7	3
153	25	56	9	3
154	27	56	9	3
155	54	56	9	3
156	55	56	7	3
157	49	57	5	2
158	52	57	5	1
159	58	57	5	1
160	39	58	5	1
161	47	58	5	3
162	57	58	5	1

Table D-1: Cambridge-Based Road Network Arc Data (Continued)

D.2 Experiment 2 Data

All inputs in Data Sets 21 and 22 are the same as Data Sets 11 and 12, respectively, with the exception of the RCT inputs and the two-day planning horizon. The RCT inputs remain constant in Data Sets 21 and 22, and are summarized below. The arc use rates and the IED emplacement from Data Sets 11 and 12 were applied to each day in Data Sets 21 and 22.

D.2.1 RCT Parameters (Data Sets 21 and 22)

RCT Index (h)	RCT_type_h	T_min_h (h)	T_max_h (h)	$dwell_h$ (h)	Max_day_h
1	2	5	6	2	2
2	2	5	6	2	2

D.2.2 RCT Availability (Data Sets 21 and 22)

RCT Index (h)	$Avail\_start\_0_{h,1}$	$Avail\_start\_1_{h,1}$	$Avail\_start\_0_{h,2}$	$Avail\_start\_1_{h,2}$
1	0	24	0	24
2	0	24	0	24

D.2.3 RCT Locations (Data Sets 21 and 22)

RCT Index (h)	$base_h$
1	32
2	32

D.2.4 RCT Configuration 2 Detection Probabilities (Data Sets 21 and 22)

IED type (k)	$P\_detect_{2,k,1}$	$P\_detect_{2,k,2}$	$P\_detect_{2,k,3}$
1	0.9	0.85	0.85
2	0.9	0.85	0.75

D.2.5 RCT Clearance Speeds (Data Sets 21 and 22)

Configuration Type (q)	$V_{RCT\_typeh,1}$	$V_{RCT\_typeh,2}$	$V_{RCT\_typeh,3}$
2	1	0.75	0.5

D.3 Experiment 3 Data

D.3.1 Data Set 31

Because of the size of the Utah road network, we do not include all of the Data Set 31 inputs in this appendix. The road network was partitioned into 12 disjoint areas of operations for

Experiment 3. Additionally, two RCTs were assigned to clear interstate highways only and were not limited to a specific area of operations. Table D-2 contains information on this data set, by RCT. Figure D-17 shows the Utah road network color-coded by area of operations, with RCT bases indicated. It is clear from the values in Table D-2 that there is a large variance in arc lengths in the Utah road network.

RCT Index	Area of Operations	Number of Arcs	Total Length
1	1	76	910 miles
2	2	60	739 miles
3	3	60	756 miles
4	4	86	974 miles
5	4	86	974 miles
6	5	58	1105 miles
7	6	50	410 miles
8	7	18	539 miles
9	8	50	184 miles
10	9	98	621 miles
11	10	284	592 miles
12	10	284	592 miles
13	11	30	403 miles
14	12	60	370 miles
15	(Interstate)	332	1,874 miles
16	(Interstate)	340	1,891 miles

Table D-2: Utah Road Network and RCT Summary

Some non-interstate arcs were included in RCT 16's area of operations in order to allow it to travel from its base to the nearest interstates. Each arc in the Utah road network has associated use and IED emplacement rates as a function of time, much like the data sets for the Cambridge-based road network. Arcs are classified into nine distinct categories in this data set.

D.3.2 Data Set 32

With the exception of the seven day planning horizon, Data Set 32 is the same as Data Set 22. The IED emplacement rates and arc use rates from Data Set 12 (and Data Set 22) were applied to each day in the seven day planning horizon for Data Set 32. RCT availability for days 3 and 5 in the planning horizon are summarized below. Each RCT is available for all 24 hours for every other day.

RCT Index (h)	$Avail\_start\_0_{h,3}$	$Avail\_start\_1_{h,3}$	$Avail\_start\_0_{h,5}$	$Avail\_start\_1_{h,5}$
1	0	0	0	24
2	0	24	0	0

As indicated, RCT 1 is not available for missions on day 3 and RCT 2 is not available for missions on day 5.

Experiment 3 also required some planned convoy information for the partial and perfect information cases. The methods for producing this additional data are explained in Appendix E.

[This Page Intentionally Left Blank]

Appendix E: Convoy and Patrol Data Generation

This appendix explains the method used to create the convoy and patrol plans used in Experiment 3.

E.1 Inputs

This method takes a set of convoy and patrol routes, expected numbers of convoys per day, convoy speeds, and the probabilistic arc use data as its inputs. We now discuss each of these input requirements.

E.1.1 Convoy and Patrol Routes

The first inputs we create for Convoy and Patrol Plan Generation are the convoy routes to be used. To generate data for Experiment 3, we categorized these routes into four types. These routes were specified as a sequence of (connected) nodes that the convoy would follow from its origin to its destination.

E.1.1.1 Theater Logistical Routes

We first consider theater-level logistical convoys. These convoys bring supplies and personnel into the operational region, or move supplies and personnel between theater-level supply areas within the region. In the Utah network, two nodes (the base node for RCTs 4 and 15, and the base node for RCTs 11, 12, and 16) were designated as theater supply areas. We then identified four routes in the Utah road network that could be used to either move supplies from neighboring states to each of these bases, or to move supplies between these two locations. Each of these four routes used interstates or other major highways which had relatively high arc usage rates in Data Set 31. In the Cambridge-based Network, we identified three theater logistical routes that simply passed through the area of operations on paths comprised of high-use arcs, presumably on their way to or from a theater supply area.

E.1.1.2 Local Supply Routes

Smaller operating bases naturally require logistical support as well. We therefore create local supply routes to each base in each road network. For the Utah data, we planned one or two routes to each RCT base, other than the ones designated as theater supply areas, from the nearest

theater supply area or state border. Again, these routes consisted of interstates and major highways with high arc use rates in Data Set 31. In all, 18 local supply routes were created on the Utah road network. For the Cambridge-based road network, three paths were created to supply each base, resulting in a total of six local supply routes. These routes entered the Cambridge-based road network on high-use arcs and followed high-use paths (representing the main supply routes) to each base.

E.1.1.3 Adjacent Base Routes

In addition to routine logistical requirements, there might exist an occasional need to move supplies or personnel from one base to another, for which neither base is a theater supply area. To allow for this possibility, seven paths between nearby bases (not theater supply areas) were created in the Utah road network. In the Cambridge-based road network, three paths were created linking the two bases used in Experiment 1. Again, these routes used interstates and other paths consisting of high-use arcs as much as possible.

E.1.1.4 Patrols

The final routes created for Convoy and Patrol Plan Generation were for local patrols. We expect several patrols to originate and terminate at each base every day as an inherent part of conducting military operations. For each sub-network (partition) in the Utah road network, two to seven local patrol routes were created. These patrol paths consisted of arcs of varying road types and use rates, and were selected in an effort to obtain a representative sample of the sub-network. For the Cambridge-based road network, four patrol routes were created for each base, covering almost every arc connection in the graph.

E.1.2 Expected Numbers of Convoys and Patrols

The second set of inputs to Convoy and Patrols Plan Generation are the expected numbers of convoys and patrols per day, by type. For Experiment 3, we used the values in Tables E-1.

	Utah Road Network (Data Set 3.1)		Cambridge-Based Road Network (Data Set 3.2)	
Information Case	Partial	Perfect	Partial	Perfect
Theater Supply Convoys	10	20	4	8
Local Supply Convoys (per base)	3	6	3	6
Adjacent Base Convoys	12	24	2	4
Patrols (per base)	6	12	6	12

Table E-1: Expected Convoy and Patrol Values for Experiment 3

E.1.3 Convoy Speeds

The speeds at which the convoys move along different types of arcs comprise the third set of inputs to the Convoy and Patrol Plan Generation function. Table E-2 gives the values used in Experiment 3.

Road Classification	1	2	3	4	5	6	7	8	9
Utah Convoy Speed (mph)	45	35	30	20	10	40	25	20	15
Cambridge Convoy Speed (units/min)	6	4	3	--	--	--	--	--	--

Table E-2: Convoy and Patrol Speeds for Experiment 3

E.1.4 Arc Use Rates and Initialization Parameters

The final inputs for the Convoy and Patrol Plan Generator are the statistically determined arc use rates, $\mu_{a,t}$, the discrete time step increment, δ, and a day d in the planning horizon.

E.2 Process

E.2.1 Determine Use Probabilities

The Convoy and Patrol Plan Generation function first determines arc use probabilities for each stage in a one-day horizon from the input arc use rates:

$$D := \left\{(d-1) \times \left(\frac{24\times 60}{\delta}\right) + 1 .. d \times \left(\frac{24\times 60}{\delta}\right)\right\},$$

$$P\_use_{a,t} := 1 - e^{-\mu_{a,t}\delta}, \forall a \in A, \forall t \in D.$$

E.2.2 First Weighting Function

Using the arc use probabilities, the function assigns a weighting value to each input path. For any input route $P \in S^c$, for all input route types *c*, we define

$$A^P := \{\text{The set of arcs on route } P, \text{ in each direction}\},$$

$$S^c := \{\text{The set of input paths of type } c\},$$

$$w^P := \frac{\sum_{\bar{a}\in A^P} \sum_{t\in D} -\ln(1-P\_use_{a,t})}{\sum_{P\prime\in S^c} \sum_{\bar{a}\in A^{P\prime}} \sum_{t\in D} -\ln(1-P\_use_{a,t})}.$$

In these expressions, *c* designates a specific type of input path. For patrols and local supply routes, the parameter *c* includes a supporting base. For the Convoy and Patrol Plan Generation carried out for Experiment 3, *c* could be a theater logistic route, an adjacent base route, a patrol from base *i*, or local supply route to base *j*, for all base nodes *i* and *j*. These weights are normalized by route type so that

$$\sum_{P\in S^c} w^P = 1.$$

E.2.3 Second Weighting Function

The function creates a second normalized weighting function for each path, for each stage in the day. We define for all input paths *P*:

$$e_P := \{\text{The time required for a convoy to traverse route } P\},$$

$$v_t^P := \frac{\sum_{\bar{a}\in A^P} \sum_{\bar{t}\in\{t..t+e_P\}} -\ln(1-P\_use_{a,\bar{t}})}{\sum_{t\prime\in D} \sum_{\bar{a}\in A^{P\prime}} \sum_{\bar{t}\in\{t\prime..t\prime+e_P\}} -\ln(1-P\_use_{a,\bar{t}})}, \forall t \in D.$$

These weights are normalized for each route, so that

$$\sum_{t\in D} v_t^P = 1, \forall P.$$

E.2.4 Stochastic Process

The function uses a Poisson process to generate a convoy and patrol plan. We define probabilities

$$\pi_t^P := 1 - e^{-(n^c \times w^P \times v_t^P)}, \forall P \in S^c, \forall t \in D, \forall c \in \{\text{Input route types}\}, \text{ for}$$

$n^c :=$ {The expected number of type *c* convoys per day (see Table E-2)}

These probabilities are used to determine whether a convoy or patrol begins executing route *P* at stage *t*. The normalized weights put more probability mass on the routes with higher use rates, while maintaining the input expectation on numbers of convoys of each type.

Using a random number generator and the probabilities π_t^P, the Convoy and Patrol Plan Generation function independently determines whether each route (*P*) is executed beginning at each stage (*t*). If a route is to be executed, another random number determines which direction the convoy will travel, with 0.5 probability assigned to each direction.

E.3 Outputs

The Convoy and Patrol Plan Generation function returns a list of all routes to be executed, each indicated by a sequence of nodes and corresponding stages. The stage corresponding to each node is computed using the convoy's start time, arc travel speeds, and arc characteristics along the route. Using this information, the function outputs a deterministic use matrix, where for some small positive ε,

$$\Phi_{a,t} = \begin{cases} 1, \text{if a convoy is traversing arc } a \text{ at stage } t, \\ 0 + \varepsilon, \text{otherwise.} \end{cases}$$

The value ε is required to be positive to prevent division by zero in the route clearance planning algorithm (see Paragraph 5.4.3.4).

The Convoy and Patrol Plan Generation function was used to create deterministic data for each day in the seven day planning horizon for both the Cambridge-based data set (32) and the Utah data set (31) in Experiment 3. Using the values in table E-2, we determine the expected total number of convoys and patrols per day in the Utah road network with perfect information is 248. The average number of convoys per day generated on the Utah road network for test 3.3 was 235.3.

[This Page Intentionally Left Blank]

References

[JIE06] *About JIEDDO*. (n.d.). Retrieved May 04, 2009, from JIEDDO: https://www.jieddo.dod.mil/ABOUTJIEDDO/AJHOME.ASPX

[Bar06] Barnhart, C. (2006, August 13). *Position Statement: Cynthia Barnhart*. Retrieved May 04, 2009, from OR/MS Today: http://lionhrtpub.com/orms/news/informs/frinf806barnhart.html

[Ber05] Bertsekas, D. P. (2005). *Dynamic Programming and Optimal Control.* Nashua, NH: Athena Scientific.

[BT097] Bertsimas, D., & Tsitsiklis, J. N. (1997). Introduction to Linear Optimization. Belmont, MA: Athena Scientific.

[Br109] Brook, T. V. (2009, March 09). Coalition deaths from IED attacks soar in Afghanistan. USA Today.

[Br209] Brook, T. V. (2009, May 01). More Roadside Bomb Attacks Drive Demand for Best Shields. USA Today.

[Buf09] *Buffalo Specifications.* (2009). Retrieved March 20, 2009, from Force Protection: http://www.forceprotection.net/models/buffalo/specs/buffalo_spec.pdf

[CAL02] Center for Army Lessons Learned. (2002). *Operation Enduring Freedom Tactics, Techniques and Procedures Handbook No. 02-8.* Retrieved 03 19, 2009, from Strategy Page: http://www.strategypage.com/articles/operationenduringfreedom/chap1.asp

[DeG07] DeGregory, K. W. (2007). *Optimization-Based Allocation of Force Protection Resources in an Asymmetric Environment.* Cambridge, MA: Massachusetts Institute of Technology.

[FM196] Field Manual 100-15. (October 1996). *Corps Operations.* Washington, D.C.: Headquarters, Department of the Army.

[FM101] Field Manual 3-0. (June 2001). *Operations.* Washington, D.C.: Headquarters, Department of the Army.

[FM106] Field Manual 3-24. (December 2006). *Counterinsurgency.* Washington, D.C.: Headquarters, Department of the Army.

[FM104] Field Manual 3-34. (January 2004). *Engineer Operations.* Washington, D.C.: Headquarters, Department of the Army.

[FM109] Field Manual 3-34.22. (Feburary 11, 2009). *Engineer Operations--Brigade Combat Team and Below.* Washington, D.C.: Headquarters, Department of the Army.

[FM105] Field Manual 5-0. (January 2005). *Army Planning and Orders Production.* Washington, D.C.: Headquarters, Department of the Army.

[FM296] Field Manual 6-20-10. (May 1996). *Tactics, Techniques, and Procedures for the Targeting Process.* Washington, D.C.: Headquarters, Department of the Army.

[FM396] Field Manual 71-100. (August 1996). *Division Operations.* Washington, D.C.: Headquarters, Department of the Army.

[FM206] Interim Field Manual 5-0.1. (March 2006). *The Operations Process.* Washington, D.C.: Headquarters, Department of the Army.

[icas09] *Iraq Coalition Casualy Count; IED Fatalities*. (2009). Retrieved May 04, 2009, from icasualties.com: http://icasualties.org/Iraq/IED.aspx

[JIE07] Joint Improvised Explosive Device Defeat Organization (JIEDDO). (2007). *Annual Report: Fiscal Year 2007.* Retrieved May 04, 2009, from JIEDDO: https://www.jieddo.dod.mil/ANNUALREPORTS/20080130_FULL_Annual%20Report%20UNCLS%208.5%20x%2011_v6.pdf

[JP101] Joint Publication 1-02. (April 2001). *Department of Defense Dictionary of Military and Associated Terms.* Washington, D.C.: Headquarters, Department of Defense.

[Mac06] Macy, R. A. (2006, October 23). *Counter-IED Response in the GWOT.* Retrieved March 19, 2009, from DTIC Online: http://www.dtic.mil/ndia/2006expwarfare/archer.pdf

[Mag05] Magness, L. T. (2005, January-March). IED Defeat: Obeservations from the National Training Center. *U.S. Army Engineer Magazine* , pp. 28-31.

[Mit04] Mitchell, C. K., & Roscoe, M. H. (2004, July-September). Assured Mobility in the Army's First Stryker Brigade. *U.S. Army Engineer Magazine* , pp. 4-8.

[Glo09] *Package-Type Improvised Explosive Devices (IEDs)*. (2000-2009). Retrieved March 19, 2009, from GlobalSecurity.org: http://www.globalsecurity.org/military/intro/ied-packaged.htm

[Put08] Putnam, M. D. (2008, May 19). *Interoperability and Combatability Techniques for Counter Radio Controlled IED Electronic Warfare (CREW) and Other Radio Frequency Communication.* Retrieved March 19, 2009, from Navy Small Buisiness Innovation Research/Small Business Technology Transfer: http://www.navysbir.com/n08_2/N082-174.htm

[Tho07] Thomer, J. L. (2007). *Trust-Based Design of Human-Guided Algorithms.* Cambridge, MA: Massachusetts Institute of Technology.

[Was06] Washburn, A. (2006). *Continuous Network Interdiction.* Monterey, CA: Naval Postgraduate School.

[Wes08] West, B. (2008). *The Strongest Tribe: War, Politics, and the Endgame in Iraq.* New York: Random House.