Strategic Location and Dispatch Management of Assets in a Military Medical Evacuation Enterprise

DISSERTATION

Phillip R. Jenkins, Capt, USAF

AFIT-ENS-DS-19-J-037

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

The views expressed in this document are those of the author and do not reflect the official policy or position of the United States Air Force, the United States Department of Defense or the United States Government.

AFIT-ENS-DS-19-J-037

STRATEGIC LOCATION AND DISPATCH MANAGEMENT OF ASSETS IN A MILITARY MEDICAL EVACUATION ENTERPRISE

DISSERTATION

Presented to the Faculty
Graduate School of Engineering and Management
Air Force Institute of Technology
Air University
Air Education and Training Command
in Partial Fulfillment of the Requirements for the
Degree of Doctor of Philosophy

Phillip R. Jenkins, BS, MS
Capt, USAF

June 2019

AFIT-ENS-DS-19-J-037

STRATEGIC LOCATION AND DISPATCH MANAGEMENT OF ASSETS IN A MILITARY MEDICAL EVACUATION ENTERPRISE

DISSERTATION

Phillip R. Jenkins, BS, MS
Capt, USAF

Committee Membership:

Lt Col Matthew J. Robbins, PhD
Co-Chair

Dr. Brian J. Lunday
Co-Chair

Lt Col Richard S. Seymour, PhD
Member

Adedeji B. Badiru, PhD
Dean, Graduate School of Engineering and Management

AFIT-ENS-DS-19-J-037

Abstract

This dissertation considers the importance of optimizing deployed military medical evacuation (MEDEVAC) systems and utilizes operations research techniques to develop models that allow military medical planners to analyze different strategies regarding the management of MEDEVAC assets in a deployed environment. For optimization models relating to selected subproblems of the MEDEVAC enterprise, the work herein leverages integer programming, multi-objective optimization, Markov decision processes (MDPs), approximate dynamic programming (ADP), and machine learning, as appropriate, to identify relevant insights for aerial MEDEVAC operations. This research is conducted in the form of three related research components: the first component develops models for optimal MEDEVAC location within an enterprise of assets, whereas the second and third components seek optimal MEDEVAC dispatching policies for a set of established assets via a sequence of models of increasing operational complexity and corresponding solution methods.

Determining where to locate mobile aeromedical staging facilities (MASFs) as well as identifying how many aeromedical helicopters to allocate to each MASF, commonly referred to as the MEDEVAC location-allocation problem, is vital to the success of a deployed MEDEVAC system. An integer mathematical programming formulation is developed to determine the location and allocation of MEDEVAC assets over the phases of a military deployment to support operations ranging from peacekeeping through combat to post-combat resolution. The model seeks to address the multi-objective problem of maximizing the total expected coverage of demand as a measure of solution effectiveness, minimizing the maximum number of located MASFs in any deployment phase as a measure of solution efficiency, and minimizing the total number of MASF relocations throughout the deployment as a measure of solution robustness. Moreover, the model utilizes the ε-constraint Method to assess and recommend improvements to deployed military MEDEVAC systems designed to provide large-scale emergency medical response for contingency operations that range in casualty-inducing intensity over the phases of a deployment. Comparisons are made between the model's (multi-phase) optimal solution and the phase-specific optimal solutions that disregard concerns of transitions between phases for a realistic, synthetically generated medical planning scenario in southern Azerbaijan. The results highlight the conflicting nature between the objectives and illustrate the tradeoffs between objectives as restrictions applied to the second and third objectives are respectively tightened or relaxed.

Military medical planners must also consider how MEDEVAC assets will be dispatched when preparing for and supporting high-intensity combat operations. The dispatching authority seeks to dispatch MEDEVAC assets to prioritized requests for service such that battlefield casualties are effectively and efficiently transported to nearby medical treatment facilities. A discounted, infinite-horizon MDP model of the MEDEVAC dispatching problem is developed. Unfortunately, the high dimensionality and uncountable state space of the MDP model renders classical dynamic programming solution methods intractable. Instead, ADP solution methods are applied to produce high-quality dispatching policies relative to the currently practiced closest-available dispatching policy. Two distinct ADP solution techniques are developed, tested, and compared, both of which utilize an approximate policy iteration (API) algorithmic framework. The first algorithm uses least-squares temporal differences (LSTD) learning for policy evaluation, whereas the second algorithm uses neural network (NN) learning. A notional, yet representative planning scenario based on high-intensity combat operations in southern Azerbaijan is constructed to demonstrate the applicability of the MDP model and to compare the efficacies of the proposed ADP solution techniques. Thirty problem instances are generated via a designed experiment to examine how selected problem features and algorithmic features affect the quality of solutions attained by the ADP policies. Results show that the respective policies determined by the NN-API and LSTD-API algorithms significantly outperform the closest-available benchmark policies in 27 (90%) and 24 (80%) of the problem instances examined. Moreover, the NN-API policies significantly outperform the LSTD-API policies in each of the problem instances examined. Compared to the closest-available policy for the baseline problem instance, the NN-API policy decreases the average response time of important *urgent* (i.e., life-threatening) requests by 39 minutes.

This dissertation also examines the MEDEVAC dispatching, preemption-rerouting, and redeployment (DPR) problem. A discounted, infinite-horizon MDP model of the MEDEVAC DPR problem is formulated and solved via an ADP strategy that utilizes a support vector regression value function approximation scheme within an API framework. The objective is to maximize the expected total discounted reward attained by the system. The applicability of the MDP model is examined via a notional, representative planning scenario based on high-intensity combat operations in Azerbaijan. Computational experimentation is performed to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated DPR policies and to highlight the efficacy of the proposed solution methodology. The results from the computational experiments indicate the ADP-generated policies significantly outperform the two benchmark policies considered. Moreover, the results reveal that the average service time of high-precedence, time sensitive requests decreases when an ADP policy is adopted during high-intensity conflicts. As the rate in which requests enter the system increases, the performance gap between the ADP policy and the first benchmark policy (i.e., the currently practiced, closest-available dispatching policy) increases substantially. Conversely, as the rate in which requests enter the system decreases, the ADP performance improvement over both benchmark policies decreases, revealing that the ADP policy provides little-to-no benefit over a myopic approach (e.g., as utilized in the benchmark policies) when the intensity of a conflict is low.

In aggregate, these research models, methodologies, and results inform the implementation and modification of current and future MEDEVAC tactics, techniques, and procedures, as well as the design and purchase of future aerial MEDEVAC assets.

To my daughters - dream big, be confident, and never quit.

Acknowledgements

Foremost, I would like to express my sincere gratitude to my advisors, Dr. Matthew J. Robbins and Dr. Brian J. Lunday, for their mentorship, support, and patience throughout the writing of this dissertation. I would not be where I am today without their guidance, and I hope to inspire others like they have inspired me. I am grateful for having the opportunity to work under their advisement.

Thanks to my committee member, Dr. Richard S. Seymour, for his invaluable insights that ultimately improved the presentation and contents of this dissertation.

I owe my appreciation to the Medical Evacuation Propenency Division for sponsoring this line of research. This work is also partially supported by two research fellowships from the Omar Nelson Bradley foundation and a Seth Bonder Scholarship for Applied Operations Research in Military and Security Applications.

Lastly, I would like to thank my family for their endless patience and unwavering support. A special shout-out goes to my wife for always being there for me, especially during the hard times. I couldn't have done this without you.

Phillip R. Jenkins

Table of Contents

List of Figures

List of Tables

STRATEGIC LOCATION AND DISPATCH MANAGEMENT OF ASSETS IN A MILITARY MEDICAL EVACUATION ENTERPRISE

I. Introduction

1.1 Motivation

A major focus of the Military Health System is to evacuate combat casualties in an effective and efficient manner. Casualties are typically transported from predetermined casualty collection points (CCPs) to medical treatment facilities (MTFs) via casualty evacuation (CASEVAC), medical evacuation (MEDEVAC), or aeromedical evacuation (AE). CASEVAC refers to the unregulated transport of casualties to an MTF via non-medical assets *without* en route medical care (Department of the Army, 2016). MEDEVAC refers to the United States (US) Army, Marine Corps, Navy, and Coast Guard transport of casualties to an MTF via standardized medical evacuation platforms equipped and staffed with medical professionals for en route medical care (Department of Defense, 2012). AE refers to the US Air Force (USAF) transport of casualties to an MTF via predesignated tactical platforms equipped and staffed with medical professionals for en route medical care (Department of Defense, 2012). Casualties transported via CASEVAC may not receive the necessary treatment and/or be transported to the appropriate MTF. As such, MEDEVAC and AE are the preferred modes of casualty transport during high intensity combat operations (Department of Defense, 2012).

Patient movement requirements centers (PMRCs) are responsible for managing casualty evacuation throughout the entire duration of joint combat operations. To ensure visibility of the joint assets available for casualty evacuation, PMRCs typically operate at the joint level. However, there are many instances wherein combat operations are conducted independently by each service organization under their respective chains of command (Department of Defense, 2012). This dissertation examines independent US Army combat operations and assumes that other service evacuation platforms (e.g., AE) are unavailable for casualty evacuation. More specifically, this dissertation focuses on the aerial aspect of MEDEVAC operations (i.e., aeromedical helicopter operations).

The US military first employed aeromedical helicopters in combat operations during the Korean Conflict (Bradley et al., 2017). The ability to travel faster, farther, and across terrain in remote areas not accessible to other evacuation platforms (e.g., ground and sea) quickly made aeromedical helicopters a high visibility asset of the MEDEVAC system. The unique capabilities (e.g., speed, coverage, and flexibility) of aeromedical helicopters have greatly contributed to the recent increases in casualty survivability rates. Before aeromedical helicopters were employed, approximately 80% of casualties during World War II survived (Eastridge et al., 2012). This number increased to 84% during the Vietnam War and ultimately 90% during the continuous decade of US conflicts between 2001-2011 (Eastridge et al., 2012). Aeromedical helicopters provide the MEDEVAC system the ability to simultaneously treat and quickly transport combat casualties from CCPs to appropriate MTFs. This leads to decreased response times and improves the survivability rates of combat casualties.

Prior to engaging in major combat operations, military medical planners seek to design an effective, efficient, and flexible MEDEVAC system. An effective and efficient MEDEVAC system reduces the chances of combat casualties acquiring long-term disabilities and increases the probability of survival of combat casualties. Moreover, an effective and efficient MEDEVAC system improves the *esprit de corps* of deployed military personnel, who understand that quality care will be provided to them quickly if they are injured in combat. A flexible MEDEVAC system gives medical planners the ability to rapidly task-organize and relocate MEDEVAC assets (i.e., aeromedical helicopters) to quickly address tactical changes in combat requirements (Department of the Army, 2016). Military medical planners must also consider important decisions such as where to locate aeromedical helicopter staging areas and MTFs; how many aeromedical helicopters to allocate to each staging area; which MEDEVAC dispatching policy to utilize; and when, if necessary and/or possible, to relocate, redeploy, or reroute aeromedical helicopters. These important decisions are vital to the success of MEDEVAC systems and are the primary foci of this dissertation.

This dissertation considers the importance of optimizing deployed military MEDEVAC systems and utilizes operations research techniques to develop models that allow military medical planners to analyze different strategies regarding the management of MEDEVAC assets in a deployed environment. For optimization models relating to selected subproblems of the MEDEVAC enterprise, the work herein leverages integer programming, multi-objective optimization, Markov decision processes (MDPs), approximate dynamic programming (ADP), and machine learning, as appropriate, to identify relevant insights for aerial MEDEVAC operations. Moreover, realistic, but notional, computational examples are utilized to illustrate the impact and relevance of the models developed in this dissertation.

1.2 Dissertation Overview

This dissertation is organized as follows, wherein Chapters II-IV correspond to the three major research components and Chapter V identifies holistic conclusions and recommendations for the research endeavor. Chapter II examines the multi-phase MEDEVAC location-allocation problem wherein military medical planners must decide where to locate mobile aeromedical staging facilities (MASFs) and, implicitly, co-located MTFs as well as identify how many aeromedical helicopters to allocate to each located MASF throughout the phases of a deployment. An integer mathematical programming formulation is constructed to determine the location and allocation of MEDEVAC assets for each phase of a deployment. Whereas the identification of an optimal coverage solution for each phase may require a large number of located MASFs and a significant number of MEDEVAC asset relocations as a force transitions between phases, the model also seeks to minimize both the maximum number of located MASFs in any deployment phase and the total number of MASF relocations throughout the deployment. More specifically, the model seeks to address the multi-objective problem of maximizing the total expected coverage of demand as a measure of solution effectiveness, minimizing the maximum number of located MASFs in any deployment phase as a measure of solution efficiency, and minimizing the total number of MASF relocations throughout the deployment as a measure of solution robustness. An illustration of the model's applicability is demonstrated via a realistic, synthetically generated medical planning scenario in southern Azerbaijan.

Chapter III examines the MEDEVAC dispatching problem wherein a dispatching authority must decide which (if any) MEDEVAC unit to dispatch in response to a submitted 9-line MEDEVAC request. A discounted, infinite-horizon MDP model of the MEDEVAC dispatching problem is formulated to maximize the expected total discounted reward attained by the system. Whereas the MDP model provides an appropriate mathematical framework for solving the MEDEVAC dispatching problem, classical dynamic programming techniques (e.g., policy iteration or value iteration) are computationally intractable due to the high dimensionality and uncountable state space of practical scenarios (i.e., large-scale problem instances). As such, two ADP strategies are designed, tested, and employed to produce high-quality dispatching policies relative to the currently practiced dispatching policy (i.e., closest-available dispatching policy). The first ADP strategy utilizes least-squares temporal differences learning within an approximate policy iteration (API) algorithmic framework, whereas the second strategy leverages neural network learning within an API algorithmic framework. Utilizing features from the MEDEVAC dispatching problem, a set of basis functions is defined to approximate the value function around the post-decision state for both ADP strategies. A notional, representative planning scenario based on high-intensity combat operations in southern Azerbaijan is constructed to demonstrate the applicability of the MDP model and to examine the efficacy of the proposed ADP strategies. Moreover, designed computational experiments are conducted to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated dispatching policies.

Chapter IV examines the MEDEVAC dispatching, preemption-rerouting, and redeployment (DPR) problem wherein a decision maker seeks a policy that determines which MEDEVAC units to assign (i.e., dispatch or preempt-and-reroute) to respond to requests for service and where MEDEVAC units redeploy after finishing a service request (i.e., successfully transferred casualty care to an MTF's staff). A discounted, infinite-horizon MDP model of the MEDEVAC DPR problem is formulated and solved via an ADP strategy that utilizes a support vector regression value function approximation scheme within an API framework. The objective is to generate high-quality policies that dispatch, preempt-and-reroute, and redeploy MEDEVAC units in a way that improves upon the currently practiced closest-available dispatching policy for large-scale, highly kinetic, and intense military conflicts. The applicability of the MDP model and the efficacy of the ADP strategy are illustrated via a notional, representative planning scenario based on high-intensity combat operations in Azerbaijan. Moreover, computational experimentation is performed to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated DPR policies.

Chapter V summarizes the contributions of this dissertation. The assumptions, limitations, and shortcomings of the respective models are identified and discussed, with an emphasis on how they can be mitigated in future extensions to the research examined herein.

II. Robust, Multi-Objective Optimization for the Military Medical Evacuation Location-Allocation Problem

2.1 Introduction

Aerial military medical evacuation (MEDEVAC) is a critical component of the Military Health System and has helped ensure the delivery of healthcare across the continuum of combat operations since its first appearance in the Korean War (Bradley et al., 2017). MEDEVAC involves the rapid transport of combat casualties from a predetermined casualty collection point (CCP) to a medical treatment facility (MTF) via medically equipped platforms staffed with medical personnel for en route medical care. Casualty evacuation (CASEVAC) is an alternative means for transporting combat casualties from a CCP to an MTF; however, CASEVAC does not utilize standardized medical platforms and does not have the capability to provide vital en route medical care to the casualties being transported. As such, MEDEVAC serves as the primary link between levels of medical care in combat operations (Department of the Army, 2016).

According to Kotwal et al. (2016), 21,089 United States (US) military casualties occurred in Afghanistan between September 11, 2001 and March 31, 2014, of which 1,350 were killed in action (KIA). The military casualties that were KIA include those that were killed immediately and those that were injured and died before reaching an MTF. Although significantly fewer casualties were KIA during this time period than during previous wars (e.g., 152,359 in World War II and 38,281 in Vietnam), the US military can still improve its systematic approach for evacuating and treating combat casualties. The time between injury and treatment dictates the effectiveness of a medical evacuation system (Kotwal et al., 2016; Aringhieri et al., 2017; Bélanger et al., 2019). The widely stated "golden-hour" concept for military MEDEVAC refers to the one hour time window during which medical intervention after sustaining critical injury is more likely to be lifesaving. This concept influenced the US Secretary of Defense to institute a policy in 2009 mandating that combat casualties receive medical care within one hour from the initial time of injury (Bradley et al., 2017). This policy, along with the long distances and rugged terrains often encountered in deployed environments (which are unfavorable conditions for ground evacuation platforms) has led to an increase in US military reliance on aerial MEDEVAC for evacuating casualties during combat operations (De Lorenzo, 2003; Clarke and Davis, 2012; Kotwal et al., 2016). For example, over 90% of the 21,089 US military casualties that occurred in Afghanistan between September 11, 2001 and March 31, 2014 were transported via aerial MEDEVAC (Kotwal et al., 2016). Since the majority of military MEDEVAC operations are conducted utilizing aerial MEDEVAC platforms (i.e., aeromedical helicopters), this research focuses solely on the aerial aspect of military MEDEVAC operations.

Military medical planners seek to design effective, efficient, and flexible MEDEVAC systems prior to engaging in combat operations. An effective and efficient MEDEVAC system increases the survivability of combat casualties, reduces the chances of long-term disabilities, and enhances the morale of deployed military personnel by demonstrating that rapid and quality medical care is available upon request. Moreover, a flexible MEDEVAC system allows military medical planners to rapidly task-organize and relocate MEDEVAC assets (i.e., mobile aeromedical staging facilities (MASFs) and aeromedical helicopters) to address tactical changes in battlefield requirements (Department of the Army, 2016). Determining where to locate MASFs and, implicitly, co-located MTFs as well as identifying how many aeromedical helicopters to allocate to each located MASF, commonly referred to as the MEDEVAC location-allocation problem, is vital to the success of a deployed MEDEVAC system and is the primary focus of this research. The MEDEVAC location-allocation problem seeks to maximize the effectiveness, efficiency, and robustness of a MEDEVAC system subject to resource, force projection (i.e., the ability to mobilize, deploy, and redeploy military forces), and logistical constraints. Despite the strategic and long-term nature of combat operations, aeromedical helicopters can be redistributed among the possible MASF locations, particularly as the level of conflict intensity changes, which typically corresponds to changes in the number and location of casualty cluster centers (CCCs). Within this research, we consider the MEDEVAC location-allocation problem over *deployment phases*, wherein each deployment phase corresponds to a different level of conflict intensity that induces a different set of CCCs.

Whereas the identification of an optimal coverage solution for each phase may require a large number of located MASFs and a significant number of MEDEVAC asset relocations as a force transitions between phases, we also seek to minimize both the maximum number of located MASFs in any deployment phase and the total number of MASF relocations throughout the deployment. We develop an integer mathematical programming formulation to determine the location and allocation of MEDEVAC assets for each phase of a deployment.

A key feature of this research is the manner in which uncertainty is taken into consideration. Unlike other MEDEVAC location-allocation models (e.g., see Zeto et al. (2006); Bastian (2010); Grannan et al. (2015), and Lejeune and Margot (2018)), we explicitly account for variations in demand and resources over time via a multiphase model. We determine model robustness by examining the total number of MASF relocations throughout the deployment. More specifically, we consider a location-allocation strategy to be more robust if the total number of MASF relocations throughout the deployment is low, where a maximally robust solution would prevent the relocation of any located MASF throughout the entire deployment.

Herein, our model seeks to address the multi-objective problem of maximizing the total expected coverage of demand as a measure of solution effectiveness, minimizing the maximum number of located MASFs in any deployment phase as a measure of solution efficiency, and minimizing the total number of MASF relocations throughout the deployment as a measure of solution robustness. We assume that each located MASF is co-located with a medical treatment facility that has sufficient resources and capacity to handle all incoming demand. The corresponding location and relocation decisions are subject to resource, force projection, and logistical constraints.

This research makes two contributions. First, it formulates a representative mathematical programming formulation and identifies an accompanying solution methodology to assess and recommend improvements to deployed military MEDEVAC systems designed to provide large-scale emergency medical response for contingency operations that range in casualty-inducing intensity over the phases of a deployment. Second, the research illustrates the application of the model for a realistic, synthetically generated medical planning scenario in southern Azerbaijan. Comparisons are made between the model's (multi-phase) optimal solution and the phase-specific optimal solutions that disregard concerns of solution robustness.

The remainder of this chapter is organized as follows. Section 2.2 reviews the relevant research literature related to MEDEVAC location-allocation problem and multi-objective optimization, respectively. Section 2.3 presents the modeling assumptions, mathematical programming formulation, and solution method to determine optimal locations and allocations of MEDEVAC assets. Section 2.4 examines an application of the model based on a notional planning scenario, and Section 2.5 concludes the analysis and proposes areas for future research.

2.2 Literature Review

Rigorous, quantitative research concerning both civilian and military emergency medical services (EMS) response systems began in the late 1960s. The primary foci of this research thread include determining where to locate servers (Toregas et al., 1971; Church and ReVelle, 1974; Daskin and Stern, 1981), deciding how many servers to allocate at each location (Hall, 1972; Berlin and Liebman, 1974; Baker et al., 1989), recognizing whether and when server relocation is necessary (Chaiken and Larson, 1972; Kolesar and Walker, 1974; Berman, 1981), developing server dispatch policies (Keneally et al., 2016; Rettke et al., 2016; Jenkins et al., 2018; Robbins et al., 2018; Jenkins et al., 2019), and identifying which performance measure to utilize for casualty survivability rates (Erkut et al., 2008; McLay and Mayorga, 2010; Knight et al., 2012). Another important feature of EMS response systems is the location of hospitals. Most research concerning civilian EMS response systems assume hospital locations are given as fixed. However, this assumption is typically not valid for military EMS response system research since some military planning contexts do not have pre-existing medical treatment facilities (MTFs) in the area of operations. A comprehensive literature review of the current challenges impacting EMS response systems is given by Aringhieri et al. (2017). With regard to a military medical evacuation (MEDEVAC) system planning context, medical planners are responsible for determining where to best place MTF locations and mobile aeromedical staging facilities (MASFs), as well as deciding how many aeromedical helicopters to allocate to each located MASF over the phases of a deployment.

Numerous researchers have studied the MEDEVAC location-allocation problem, which attempts to maximize system performance via the layout of MEDEVAC assets (i.e., MASFs and aeromedical helicopters) subject to resource, force projection, and logistical constraints. Zeto et al. (2006) develop a bi-criteria goal programming model that seeks to maximize system-wide demand coverage and minimize spare capacities of aeromedical helicopters in the Afghanistan theater. The authors utilize a multivariate hierarchical cluster analysis to characterize demand and estimate model parameters via a Monte Carlo simulation. Their model allocates the minimum number of aeromedical helicopters required to maximize the probability of meeting demand. Bastian (2010) proposes a robust, multi-criteria modeling approach to determine the optimal emplacement of MEDEVAC assets. The author's model takes into account the stochastic nature of casualty locations and three separate optimization goals: maximizing the aggregate expected demand coverage, minimizing spare capacities of aeromedical helicopters, and minimizing the maximum MTF site total vulnerability to enemy attack. Grannan et al. (2015) develop a binary linear programming (BLP) model with an objective of maximizing the proportion of high-priority (i.e., urgent) casualties serviced within a predetermined response time threshold (RTT). The BLP model optimally locates two types of military MEDEVAC air assets (i.e., HH-60M MEDEVAC and HH-60G Pave Hawk) and assigns these assets to casualty locations utilizing a dispatch preference list while balancing the workload among each asset type. Coverage thresholds for low-priority (i.e., routine) casualties are incorporated into the BLP model to encourage prompt service for all casualties. Lejeune and Margot (2018) develop a mixed integer nonlinear programming (MINLP) model to determine where to locate MTFs and aeromedical helicopters, how to dispatch aeromedical helicopters to the point-of-injury (POI), and to which MTF to route each MEDEVAC request for patient delivery. The objective of the model is to increase the probability of survival by providing timely evacuation and critical care to urgent battlefield casualties. The model measures system performance via the expected number of casualties transported to an MTF within one hour.

An important aspect of this research is how we account for multiple conflicting objectives. Optimization problems seek to minimize (or maximize) one or more objectives that can be subject to a set of constraints or bounds. A single-objective optimization problem considers one objective function, whereas a multi-objective optimization problem considers two or more objective functions. A single-objective optimization problem often does not adequately represent problems being studied today. More often, there exist several conflicting objectives that must be considered simultaneously. Moreover, multi-objective optimization problems typically do not have a single optimal solution but rather afford a set of alternative solutions that manifest trade-offs, called Pareto optimal solutions (i.e., non-dominated solutions). A non-dominated solution is one that cannot improve any objective function value without degrading another. Multi-objective optimization is, therefore, concerned with two equally important tasks: the generation of Pareto optimal solutions and the selection of a single most preferred Pareto optimal solution (Deb, 2014). The latter task relies on the intuition of the decision maker and their ability to express preferences throughout the optimization cycle.

The methods utilized to solve multi-objective optimization problems belong to one of three major categories: *a priori* articulation of preferences, *a posteriori* articulation of preferences, and no articulation of preferences (Marler and Arora, 2004). Each method considers the expertise of the decision maker in a different manner. Methods with *a priori* articulation of preferences (e.g., weighted sum, ε-constraint, and lexicographic) require the decision maker to define the relative importance of the objectives prior to running an optimization algorithm. However, in some instances, it is difficult for decision makers to express preferences between objectives prior to conducting analysis. Methods with *a posteriori* articulation of preferences (e.g., physical programming, normal boundary intersection, and normal constraint) take this concern into consideration and allow the decision maker to choose a solution from the Pareto optimal set. Methods with no articulation of preferences (e.g., exponential sum, Nash arbitration and objective product, and Rao's Method) do not require the decision maker to distinguish the relative importance of objectives. Most of the methods with no articulation of preferences are simplifications of methods with *a prior* articulation of preferences. We refer the interested reader to Marler and Arora (2004), Deb (2014), and Gutjahr and Nolz (2016) for an extensive summary of multi-objective optimization concepts, methods, and applications.

With regard to multi-objective location-allocation problems, the most widely used methods consider scalarization techniques (e.g., Weighted Sum Method, ε-constraint Method, and Hybrid Method) (Ehrgott, 2013). These methods fall under the *a priori* articulation of preferences category and require the decision maker to define the relative importance of each objective prior to running an optimization algorithm. The Weighted Sum Method is a classical scalarization technique that converts multi-objective problems into scalar problems by constructing a weighted sum of all the objectives (Bérubé et al., 2009). The advantage of the Weighted Sum Method is its simplicity, which has made it popular amongst many decision makers (Marler and Arora, 2010). However, one of the disadvantages of the Weighted Sum Method is its inability to find certain Pareto optimal solutions in the case of a nonconvex objective space, which has caused it to be utilized less frequently (Current et al., 1985; Zhang and Jiang, 2014). The ε-constraint Method overcomes the convexity issues of the Weighted Sum Method by iteratively optimizing one objective after converting the remaining objectives into constraints, for which the right-hand sides are respectively, parametrically changed and the formulation iteratively resolved to identify a set of Pareto optimal solutions. Given proper increments of ε, Chankong and Haimes (2008) show that the ε-constraint Method is guaranteed to find the entire set of Pareto-optimal solutions for a general multi-objective problem. Moreover, Cohon

(2013) shows that an optimal solution identified via the ε-constraint Method is guaranteed to be Pareto optimal if the constraints representing the objectives are binding and the solution is feasible. The ε-constraint Method is more often utilized when compared against other scalarization techniques (e.g., the Weighted Sum Method) within recent research concerning multi-objective location-allocation problems (e.g., see Chanta et al. (2014); Rath and Gutjahr (2014); Carrizosa et al. (2015); Khodaparasti et al. (2016), and Paul et al. (2017)). As such, we embrace the use of the ε-constraint Method herein for its simplicity and ability to address nonconvexity concerns.

2.3 Model Formulation & Solution Methodology

This section sets forth a mathematical program for the military medical evacuation (MEDEVAC) location-allocation problem. After reviewing the general modeling approach, assumptions, and the formal mathematical program, it presents our solution methodology for determining Pareto optimal solutions for medical planners to consider.

2.3.1 Model Formulation

The model focuses on the MEDEVAC location-allocation problem over the phases of a deployment. More specifically, the model herein seeks to locate mobile aeromedical staging facilities (MASFs) and allocate aeromedical helicopters in a manner that simultaneously maximizes the total expected coverage of demand, minimizes the maximum number of located MASFs in any deployment phase, and minimizes the the total number of MASF relocations throughout the deployment.

Prior to presenting the mathematical programming formulation, it is important to discuss the associated assumptions. The number of MASFs, aeromedical helicopters, and deployment phases being modeled are each limited in number. Each located MASF is assumed to be co-located with a medical treatment facility that has sufficient resources and capacity to handle all incoming demand. These assumptions reduce the complexity of the problem and allow the identification of optimal solutions in a timely manner. Moreover, to develop a representative planning scenario, military planning guidelines for combat operations identify different sets of casualty cluster centers (CCCs) for each phase of a deployment that correspond to locations at which enemy attacks are likely to occur.

To formulate the model, consider the definitions for the following sets, parameters, and decision variables:

Sets

- Let $\Psi = \{1, 2, \dots, |\Psi|\}$ denote the set of deployment phases with index $\psi \in \Psi$
- Let I^{ψ} denote the set of casualty cluster centers in deployment phase $\psi \in \Psi$ with index $i \in I^{\psi}$
- Let J denote the set of potential MASF locations with index $j \in J$

Parameters

- Let m denote the maximum number of MASFs that can be located in each deployment phase
- Let h denote the total number of aeromedical helicopters to be allocated in each deployment phase
- Let $\bar{d}$ denote the furthest distance that each aeromedical helicopter can travel per MEDEVAC mission
- Let $a_{i\psi}$ denote the total demand at CCC $i \in I^{\psi}$ in deployment phase $\psi \in \Psi$

- Let d_{ij} denote the distance from CCC i to MASF j
- Let $N_{i\psi} = \{j | j \in J, d_{ij} \leq \bar{d}\}$ denote the set of available MASF locations that are within coverage distance from CCC $i \in I^{\psi}$ in deployment phase $\psi \in \Psi$
- Let $w_{k\psi}$ denote the probability that the kth aeromedical helicopter is available and the 1st through $(k-1)$th helicopters are busy in deployment phase $\psi \in \Psi$

Decision Variables

- Let $l_{j\psi} = 1$ if a MASF is located at site $j \in J$ in deployment phase $\psi \in \Psi$, 0 otherwise
- Let $x_{j\psi}$ denote the total number of aeromedical helicopters allocated to a MASF at location $j \in J$ in deployment phase $\psi \in \Psi$
- Let $y_{ik\psi} = 1$ if CCC $i \in I^{\psi}$ is covered by at least k aeromedical helicopters in deployment phase $\psi \in \Psi$, 0 otherwise
- Let l_{max}, an intermediate decision variable, denote the maximum number of located MASFs at any point throughout the deployment
- Let $r_{j\psi}$, an intermediate decision variable, be equal to 1 if a MASF is located at site $j \in J$ in deployment phase $\psi \in \Psi$ but not in deployment phase $\psi + 1 \in \Psi$, 0 otherwise

Given this framework, we propose the following formulation of the MEDEVAC location-allocation problem, denoted as Problem **P1**:

$$\textbf{P1}\text{: } \max \ \big(f_1(\mathbf{y}), f_2(l_{max}), f_3(\mathbf{r})\big) \tag{1}$$

$$\text{subject to } \ f_1(\mathbf{y}) = \sum_{\psi \in \Psi} \sum_{i \in I^{\psi}} \sum_{k=1}^{h} a_{i\psi} w_{k\psi} y_{ik\psi}, \tag{2}$$

$$f_2(l_{max}) = -l_{max}, \tag{3}$$

$$f_3(\mathbf{r}) = -\sum_{\psi \in \Psi} \sum_{j \in J} r_{j\psi}, \tag{4}$$

$$\sum_{j \in J} l_{j\psi} \leq m, \ \forall \ \psi \in \Psi, \tag{5}$$

$$l_{j\psi} \leq x_{j\psi}, \ \forall \ j \in J, \ \forall \ \psi \in \Psi, \tag{6}$$

$$\sum_{j \in J^{\psi}} x_{j\psi} = h, \ \forall \ \psi \in \Psi, \tag{7}$$

$$x_{j\psi} \leq h l_{j\psi}, \ \forall \ j \in J, \ \forall \ \psi \in \Psi, \tag{8}$$

$$\sum_{k=1}^{h} y_{ik\psi} \leq \sum_{j \in N_{i\psi}} x_{j\psi}, \ \forall \ i \in I^{\psi}, \ \forall \ \psi \in \Psi, \tag{9}$$

$$\sum_{j \in J} l_{j\psi} \leq l_{max}, \ \forall \ \psi \in \Psi, \tag{10}$$

$$l_{j\psi} - l_{j,\psi+1} \leq r_{j\psi}, \ \forall \ j \in J, \ \forall \ \psi \in \Psi \setminus \{|\Psi|\}, \tag{11}$$

$$r_{j\psi} \in \{0,1\}, \ \forall \ \psi \in \Psi \setminus \{|\Psi|\}, \tag{12}$$

$$x_{j\psi} \in \mathbb{Z}_+, \ \forall \ j \in J, \ \forall \ \psi \in \Psi, \tag{13}$$

$$l_{j\psi} \in \{0,1\}, \ \forall \ j \in J, \ \forall \ \psi \in \Psi, \tag{14}$$

$$y_{ik\psi} \in \{0,1\}, \ \forall \ i \in I^{\psi}, \ \text{for } k = 1, 2, \ldots, h, \ \forall \ \psi \in \Psi. \tag{15}$$

The objective function (1) optimizes a combination of three objectives: (2) maximizing the total expected demand covered, (3) minimizing the maximum number of located MASFs in any deployment phase, and (4) minimizing the the total number of MASF relocations throughout the deployment. Constraint (5) ensures that no more than the maximum number of available MASFs m are located in each deployment phase ψ. Constraint (6) ensures that the MASFs located in phase ψ have at least one aeromedical helicopter allocated to them in phase ψ, thereby preventing the emplacement of a MASF without supporting helicopters. Constraint (7) ensures that all available aeromedical helicopters h are allocated across the located MASFs in phase ψ. The logical restriction that an aeromedical helicopter cannot be allocated to a non-located MASF is handled by Constraint (8). Constraint (9) counts the number of aeromedical helicopters that cover each CCC i in each deployment phase ψ. Constraint (10) determines the maximum number of located MASFs in any deployment phase. Constraint (11) determines from where MASFs are relocated between deployment phases ψ and $\psi + 1$. Constraints (12)-(15) represent non-negativity and integrality constraints.

This model belongs to the family of probabilistic models for ambulance location that, in contrast to deterministic models, account for the fact that aeromedical helicopters may not be able to support an incoming request for service because they are busy servicing other requests. The probability that a randomly selected aeromedical helicopter is busy in deployment phase ψ, which we denote as p_ψ, depends on (a) the average number of MEDEVAC requests per hour in phase ψ, λ_ψ; (b) the average service time per MEDEVAC mission, $\frac{1}{\mu}$; and (c) the number of aeromedical helicopters deployed in phase ψ, h. We define p_ψ with the following equation

$$p_\psi = \frac{\lambda_\psi}{h\mu}.$$

This definition of p_ψ assumes that each aeromedical helicopter operates independently. One approximate way to relax this assumption is to utilize correction factors in an embedded hypercube model (Batta et al., 1989; Chanta et al., 2014). The hypercube model was first developed by Larson (1974) to allow for dependencies between servers and has two underlying assumptions: (a) requests for service arrive according to a Poisson process; and (b) if a request for service is submitted to the system and all servers are busy, then the request will enter at the end of a queue and will be served in a first-in-first-out manner. We modify the correction factors given in Batta et al.

(1989) and Chanta et al. (2014) to account for the differences in λ_ψ, μ, and h over the phases of a deployment. We incorporate the modified correction factors into our model to approximate the dependencies among the aeromedical helicopters deployed in each deployment phase. As such, the probability that the kth aeromedical helicopter is available and the 1st through $(k-1)$th helicopters are busy in deployment phase $\psi \in \Psi$ is computed as follows

$$w_{k\psi} = Q(h, p_\psi, k-1)(1-p_\psi)(p_\psi^{k-1}), \text{ for } k = 1, 2, \ldots, h, \tag{16}$$

where

$$Q(h, p_\psi, j) = \frac{\left[\sum_{k=j}^{h-1} \left[\left(\frac{(h-j-1)!(h-k)}{(k-j)!}\right)\left(\frac{h^k}{M!}\right)p_\psi^{k-j}\right]\right]}{\left[\left((1-p_\psi)\sum_{i=0}^{h-1}\left(\frac{h^i}{i!}\right)p_\psi^i\right) + \left(\frac{h^h p_\psi^h}{h!}\right)\right]}, \text{ for } j = 0, 1, \ldots, h-1,$$

is the correction factor.

A virtue of this model is that, in the objective function, the weight of the variable $y_{ik\psi}$ is always less than the weight of the variable $y_{i,k+1,\psi}$ because of the definition of $w_{k\psi}$ given by Equation (16). By definition, $w_{k\psi}$ is the probability that the kth aeromedical helicopter is available and the first through $(k-1)$th helicopters are busy in deployment phase $\psi \in \Psi$ such that $w_{k\psi}$ is greater than $w_{k+1,\psi}$. As such, the variables enter the solution in the correct order, and the formulation avoids the need to add ordering constraints of the type $y_{ik\psi} \geq y_{i,k+1,\psi}$.

2.3.2 Solution Methodology

Rather than solve Problem P1 directly to identify the set of Pareto optimal solutions (e.g., via the Weighted Sum Method), we utilize the ε-constraint Method. We first reformulate Problem P1 to Problem **P2** as follows:

$$\textbf{P2}\text{: max} \quad \sum_{\psi\in\Psi}\sum_{i\in I^{\psi}}\sum_{k=1}^{h} a_{i\psi}w_{k\psi}y_{ik\psi}, \tag{17}$$

$$\text{subject to} \quad \text{Constraints (5)-(15)},$$

$$l_{max} \leq \varepsilon_2, \tag{18}$$

$$\sum_{\psi\in\Psi}\sum_{j\in J} r_{j\psi} \leq \varepsilon_3. \tag{19}$$

In this formulation, objective (17) replaces objective (1) from P1, strictly seeking to maximize the total expected demand covered. Constraint (18) bounds the second objective, the minimization of the maximum number of located MASFs in any deployment phase, to be no more than ε_2, an allowed maximum number of located MASFs in any deployment phase. Moreover, Constraint (19) bounds the third objective, the minimization of the total number of MASF relocations throughout the deployment, to be no more than ε_3, an allowed total number of MASF relocations throughout the deployment.

2.4 Testing, Results, & Analysis

In this section, we develop and utilize a representative military medical evacuation (MEDEVAC) planning scenario to demonstrate the applicability of our integer programming formulation to the military medical planning community. We design and conduct computational experiments to examine how different parameter settings for P2 (i.e., ε_2, ε_3, and h) impact the total expected demand covered. Moreover, we examine a baseline scenario and compare the model's (multi-phase) optimal solution against the phase-specific optimal solutions that disregard robustness concerns. Experiments and analysis are conducted using a dual Intel Xeon E5-2650v2 workstation having 128 GB of RAM and invoking the commercial solver CPLEX (Version

12.8) within the GAMS modeling environment (Version 24.8.5) to identify optimal solutions to instances of Problem P2. All solutions reported are optimal (i.e., both the relative and absolute optimality gaps are set equal to zero), and each instance is solved to optimality in less than 10 seconds of computational time.

2.4.1 Representative Scenario Development for Testing

Military deployments are typically categorized into six distinct phases: *Shape* (i.e., Phase 0), *Deter* (i.e., Phase 1), *Seize the Initiative* (i.e., Phase 2), *Dominate* (i.e., Phase 3), *Stabilize* (i.e., Phase 4), and *Enable Civil Authority* (i.e., Phase 5) (Department of Defense, 2017). Casualty events resulting in MEDEVAC requests are expected to occur in Phases 1, 2, and 3 with the majority of requests occurring in Phases 2 and 3. Deciding where to locate and allocate MEDEVAC assets (i.e., MASFs and aeromedical helicopters) are important decisions and should be considered throughout the phases of a deployment. Since the majority of MEDEVAC requests typically occur in Phases 2 and 3, we subdivide these phases within our scenario to Phases 2A, 2B, 3A, and 3B to provide decision makers with additional opportunities to adjust MEDEVAC assets location-allocation decisions. Our representative scenario considers five deployment phases (i.e., Phases 1, 2A, 2B, 3A, and 3B) where, for each deployment phase, the decision maker must decide where to locate and allocate MEDEVAC assets among the set of potential MASF locations.

We develop a notional, representative military MEDEVAC planning scenario in which the United States (US) military is performing high-intensity combat operations in support of the government of Azerbaijan. We utilize the Georgia, Armenia, Azerbaijan, Turkey (GAAT) scenario from the US Army's Training and Doctrine Command (TRADOC) as part of the basis for our representative military MEDEVAC planning context (Burland, 2008). TRADOC's GAAT scenario is an unclassified exercise that depicts a conflict between Azerbaijan and a fictional nation to the south named Ahuristan, and it requires participants to plan for threats from conventional forces, insurgents, and civil unrest.

Our planning scenario considers five distinct deployment phases and 29 potential mobile aeromedical staging facility (MASF) locations across Azerbaijan. Moreover, we develop five sets of casualty cluster centers (CCCs), one for each phase of the deployment, based on the projected locations of both friendly and enemy forces. Each deployment phase has between 10 and 40 CCCs with a total of 122 CCCs throughout all five phases of the deployment. Both the number of CCCs and the demand at each CCC in each deployment phase depend on the expected conflict intensity. Figure 1 depicts the 29 potential MASF locations and the 122 CCCs across the five deployment phases in the representative military MEDEVAC planning scenario.

We leverage data provided from TRADOC's GAAT scenario to determine the total number of CCCs in each phase, average number of MEDEVAC requests per hour in each phase (i.e., λ_ψ), total demand in each phase, and conflict intensity in each phase, which is listed in Table 1. We assume an average service time per MEDEVAC mission of $\frac{1}{\mu} = 1.1$ hours which is based on real MEDEVAC mission data provided by Kotwal et al. (2016).

Table 1. Representative Military MEDEVAC Planning Scenario Settings

Phase, ψ	Number of CCCs	λ_ψ	Total Demand	Conflict Intensity
1	28	0.75	180	21.4%
2A	10	1.25	300	35.7%
2B	19	2.25	540	65.3%
3A	37	3.50	840	100%
3B	28	2.50	600	71.4%

Kotwal et al. (2016) show that the time between critical injury and definitive care is an important factor for the survival of combat casualties. As such, we take into account the expected times to complete MEDEVAC mission tasks (e.g., mission

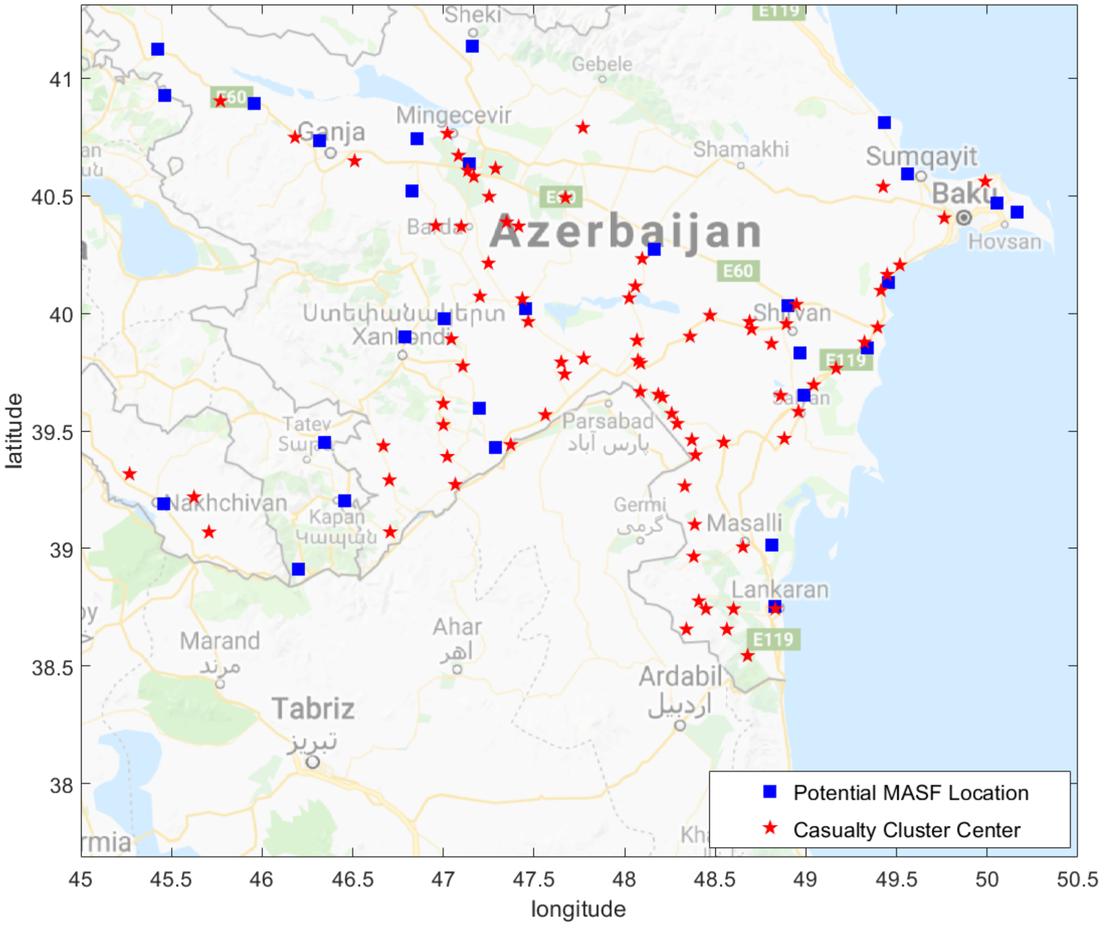


Figure 1. Representative Military MEDEVAC Planning Scenario Disposition

preparation, travel to and from casualty collection point, and load and unload casualties) and assume an aeromedical helicopter distance threshold of $\bar{d} = 437$ nautical miles (Bastian, 2010; Keneally et al., 2016). By limiting aeromedical helicopters to servicing CCCs within our distance threshold, we ensure that casualties receive the necessary, life-saving medical care in an appropriate amount of time.

2.4.2 Testing Procedure, Results, & Analysis

Utilizing the aforementioned parameter settings, we solve P2 with different values for (a) the total number of aeromedical helicopters to allocate in each phase, h; (b) the maximum number of located MASFs allowed in any phase, ε_2; and (c) the maximum number of MASF relocations allowed throughout the deployment, ε_3. As such, it is necessary to find the upper bounds on ε_2 and ε_3. To do this for a given h-value, we solve our three objectives in a lexicographic manner. That is, we solve Problem P2 in the absence of Constraints (18) and (19), denoting the optimal objective function value as ν. We find the upper bound on ε_2 by minimizing $f_2(l_{max})$, subject to Constraints (5)-(15) and $f_1(y) \geq \nu$, setting ϵ_2 equal to the resulting optimal objective function. Likewise, the upper bound on ϵ_3 is found by maximizing $f_3(r)$ subject to Constraints (5)-(15), and (18), and $f_1(y) \geq \nu$, setting ϵ_3 equal to the resulting optimal objective function value. Accordingly, we repeat this procedure to find the upper bounds on ε_2 and ε_3 for $h \in \{4, 6, 8\}$.

Once the upper bounds for ε-values are determined, we iteratively re-solve P2 over a lattice of decreasing values of ε_2 and ε_3 for each level of h to develop three h-value specific sets of operationally feasible solutions for the $(\varepsilon_2, \varepsilon_3)$-combinations (i.e., as shown in Figure 2), among which subsets of non-inferior (i.e., Pareto optimal) solutions can be identified (i.e., as shown in Figure 3).

The results from Figure 2 show that, as the number of aeromedical helicopters to

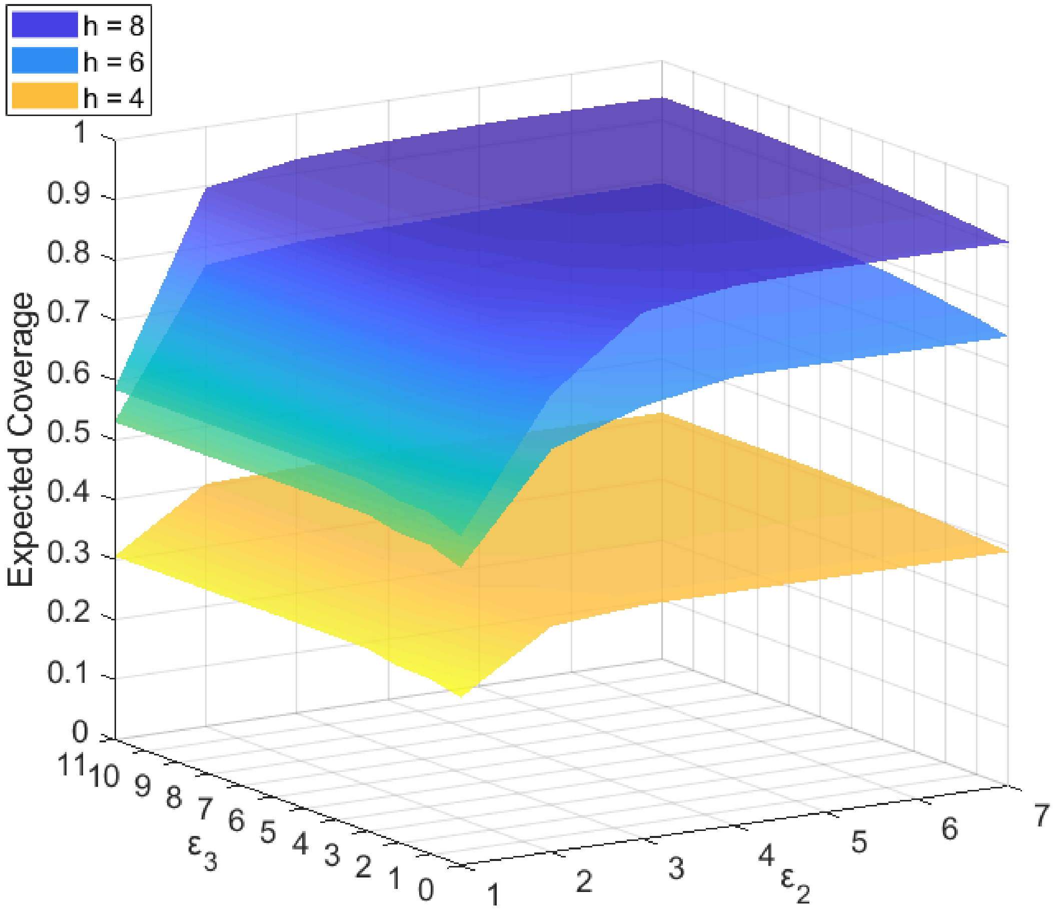


Figure 2. Optimal demand coverage values for different h-levels and $(\varepsilon_2, \varepsilon_3)$-combinations

allocate (i.e., h) increases, the total expected coverage of demand increases for all ε_2 and ε_3 values tested. For example, the total expected coverage of demand for the most restrictive cases (i.e., when $\varepsilon_2 = 1$ and $\varepsilon_3 = 0$) when h equals 4, 6, and 8 are 28.1%, 49.7%, and 55.0%, respectively. Similarly, the total expected coverage of demand for the least restrictive cases (i.e., when ε_2 and ε_3 are unbounded) when h equals 4, 6, and 8 are 41.2%, 79.5%, and 93.9%, respectively. Figure 2 also illustrates the trade-offs between the maximum number of located MASFs allowed in any phase, ε_2, and the maximum number of MASF relocations allowed throughout the deployment, ε_3. Military deployments are typically subject to resource, force projection, and logistical restrictions that impact how many MASFs can be operating (i.e., located) in any given phase and how many relocations can occur throughout the deployment. It is beneficial for military medical planners to assess the trade-offs between increasing and/or decreasing these restrictions. For example, when $(h, \varepsilon_2, \varepsilon_3) = (6, 1, 1)$, the total expected coverage is 51.6%. The results from our model reveal that it is more advantageous to increase ε_2 from 1 to 2, which results in a total expected coverage of 70.5%, rather than increase ε_3 from 1 to 2, which results in a total expected coverage of 51.8%. For this particular example, this result reveals that it is more beneficial to locate an additional MASF (if resources are available) during at least one phase of the deployment rather than allowing for one additional relocation over the course of the entire deployment.

From the solutions represented in Figure 2, Figure 3 depicts the subset of solutions that are non-inferior for each h-level. That is, Figure 3 depicts the Pareto-optimal frontier for $h \in \{4, 6, 8\}$ over the combinations of $(\varepsilon_2, \varepsilon_3)$, indicating the solutions that should be considered for implementation. For reference, the percent of demand covered by the non-inferior solutions depicted in Figure 3 are reported in Table 2. Of the 84 $(\varepsilon_2, \varepsilon_3)$-combinations depicted in Figure 2, only $4.7\%, 7.1\%$, and 8.3% are

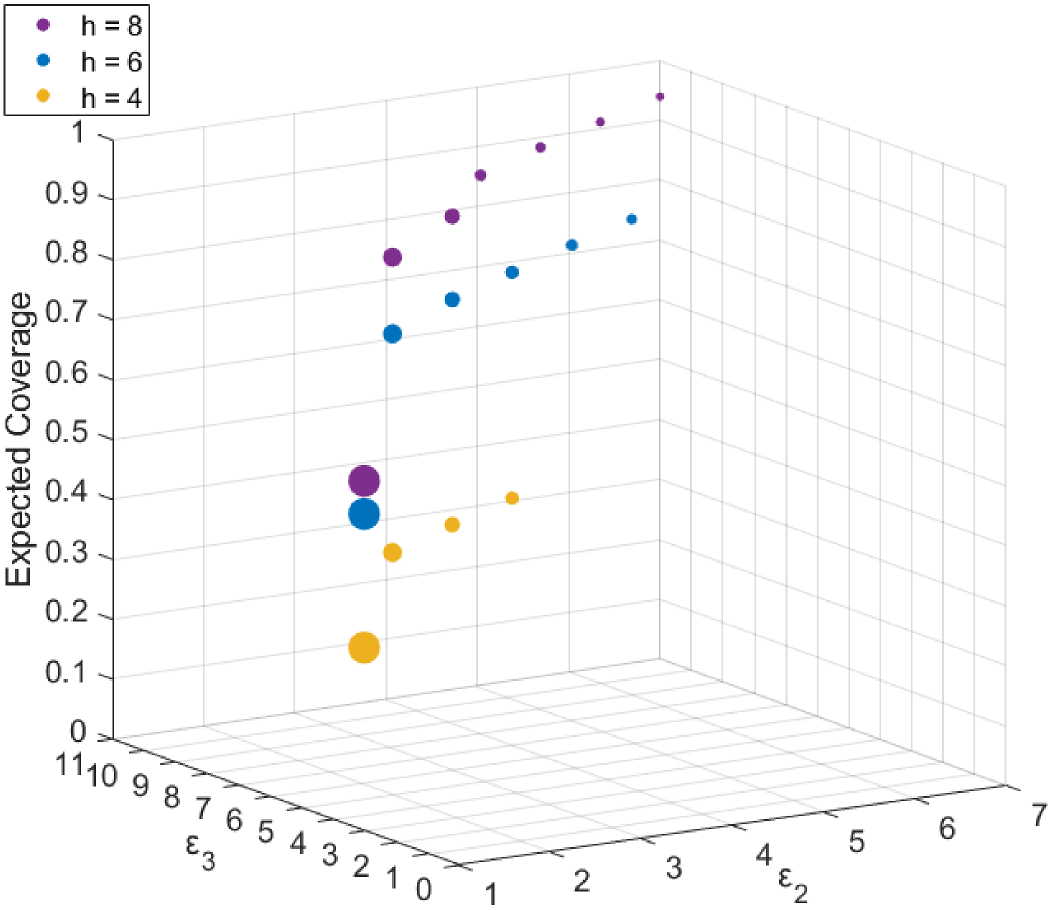


Figure 3. Pareto Optimal Points

Pareto-optimal for $h \in \{4, 6, 8\}$, respectively. These results indicate that, as h increases, the number of Pareto optimal solutions does, as well. However, this result does not hold true for larger h-values. For example, when h increases from 8 to 12 the percentage of Pareto optimal solutions remains 8.3%. By considering the operational constraints on the second and third objectives, the Pareto optimal solutions represent the minimum number of relocations required to maximize demand coverage, given ε_2. Figures 2 and 3 and Table 2 highlight the conflicting nature of the decision maker's objectives. These results can subsequently inform decisions regarding how many bases should be located and how many relocations are necessary throughout the deployment. Moreover, these results help decision makers identify effective, efficient, and robust military MEDEVAC resource allocations (e.g., concerning personnel and materiel) prior to deployment.

Table 2. Pareto Optimal Solutions

h	$(\varepsilon_2, \varepsilon_3)$	Percentage of Demand Covered
4	(1,3)	30.6%
	(2,5)	40.4%
	(3,6)	40.9%
	(4,7)	41.2%
6	(1,3)	52.9%
	(2,5)	76.9%
	(3,6)	78.5%
	(4,7)	78.9%
	(5,8)	79.3%
	(6,9)	79.5%
8	(1,3)	58.4%
	(2,5)	89.7%
	(3,6)	92.4%
	(4,8)	93.2%
	(5,9)	93.7%
	(6,10)	93.8%
	(7,11)	93.9%

2.4.3 Comparison to Phase-Specific Optimal Solutions

The analyses herein assume a baseline of $(h, \varepsilon_2, \varepsilon_3) = (8, 2, 1)$. That is, the baseline instance considers a scenario wherein eight aeromedical helicopters must be allocated in each phase, a maximum of two MASFs can be located in any given phase, and a maximum of one MASF relocation is allowed throughout the entire deployment. Utilizing the baseline case, we compare the model's multi-phase optimal solution against the phase-specific optimal solutions for each deployment phase and report the results in Figures 4-6. The multi-phase optimal solution considers both ε_2 and ε_3, whereas the phase-specific optimal solutions only consider ε_2-restrictions. That is, the phase-specific optimal solutions are not restricted by relocation constraints.

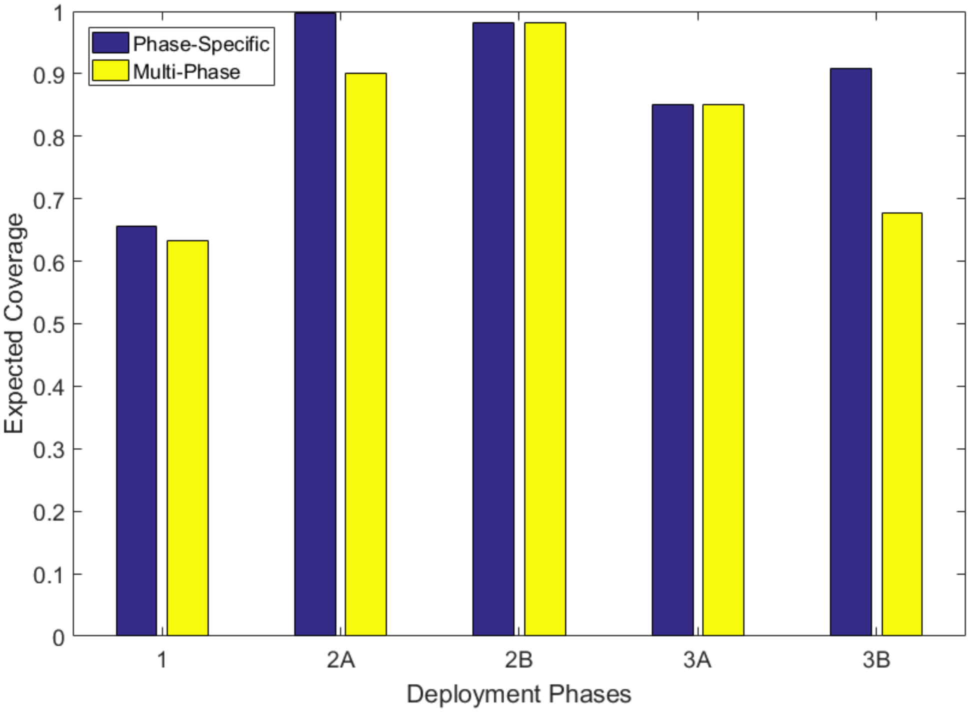
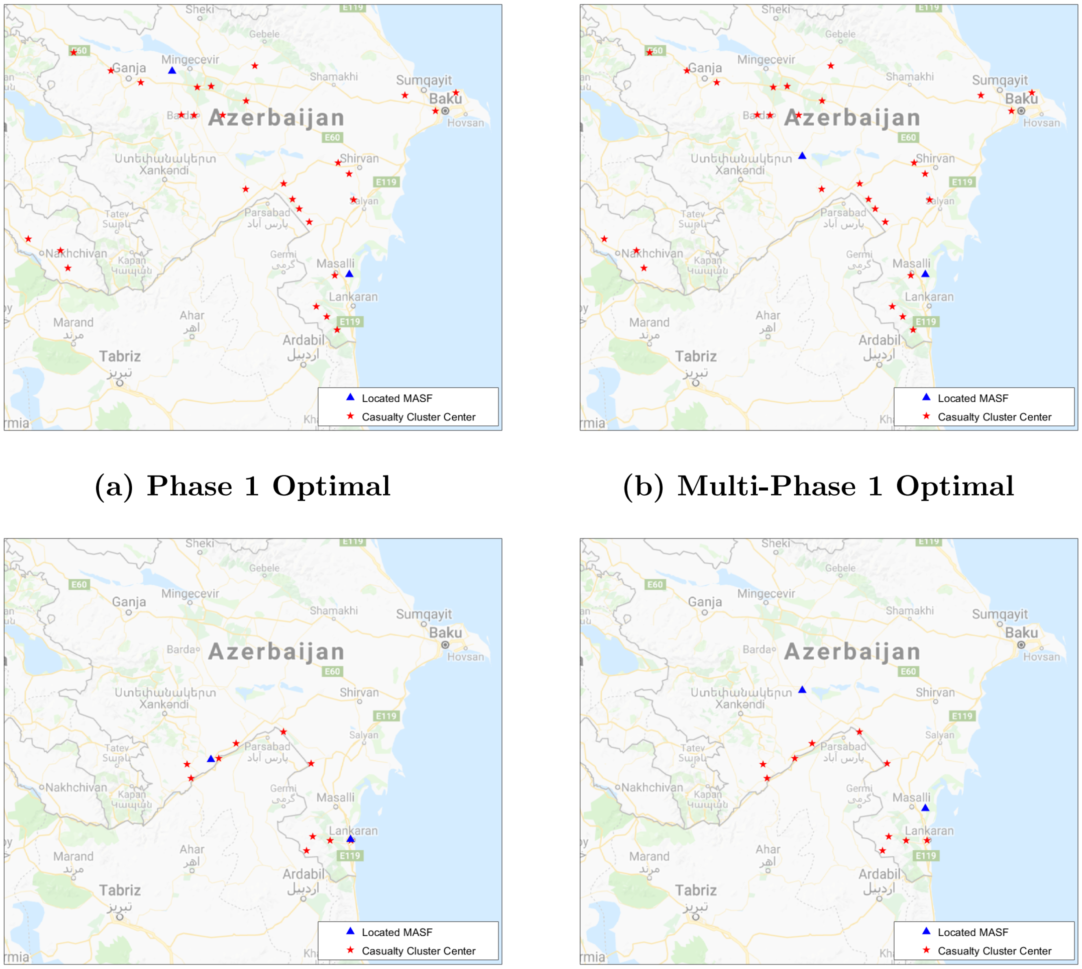


Figure 4. Phase-Specific Versus Multi-Phase Expected Coverage

Figure 4 compares the expected coverage of the phase-specific and multi-phase optimal solutions for each deployment phase. The results show that the expected coverage for the phase-specific optimal solutions is greater than or equal to the expected coverage of the multi-phase optimal solutions for each deployment phase. When only one MASF relocation is allowed (i.e., in the multi-phase solution approach), the overall expected coverage is approximately 80.86%. When this restriction is relaxed (i.e., in the phase-specific solution approach), the overall expected coverage increases to approximately 87.89%.



(a) Phase 1 Optimal

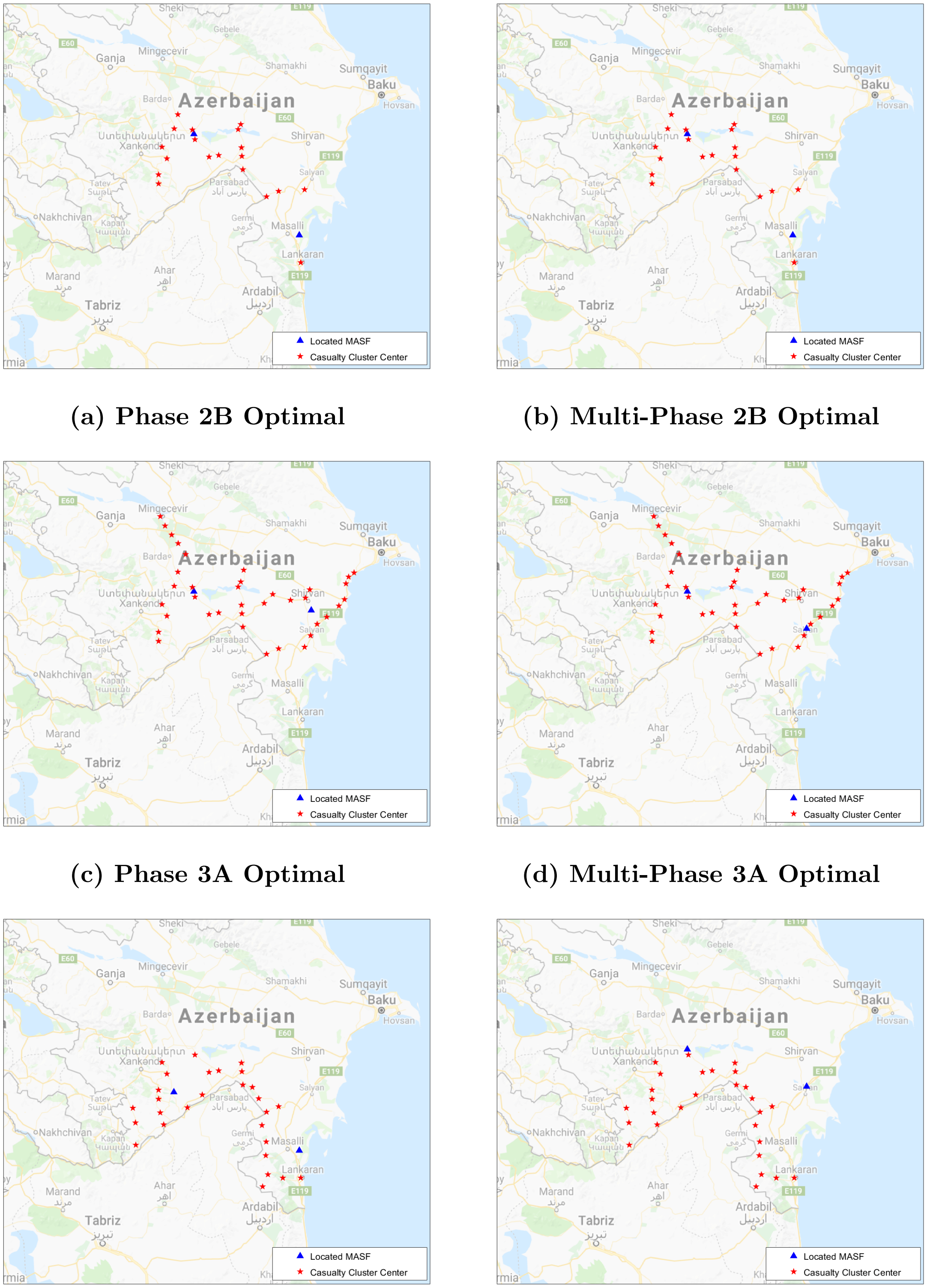
(b) Multi-Phase 1 Optimal

(c) Phase 2A Optimal

(d) Multi-Phase 2A Optimal

Figure 5. Panels (a)-(d) display the respective phase-specific and multi-phase optimal MASF locations for Phases 1 and 2A

Figures 5 and 6 illustrate the MASF locations for both the phase-specific and



(a) Phase 2B Optimal

(b) Multi-Phase 2B Optimal

(c) Phase 3A Optimal

(d) Multi-Phase 3A Optimal

(e) Phase 3B Optimal

(f) Multi-Phase 3B Optimal

Figure 6. Panels (a)-(f) display the respective phase-specific and multi-phase optimal MASF locations for Phases 2B, 3A, and 3B

multi-phase optimal solutions. As expected, the single-phase optimal solutions have more MASF relocations than the multi-phase optimal solutions. More specifically, the optimal layout of MEDEVAC assets in each deployment phase requires a total of seven MASF relocations throughout the deployment. Recall that our baseline scenario limits the multi-phase optimal solutions to one relocation throughout the entire deployment and, therefore, may not attain as much expected coverage as the phase-specific optimal solutions.

2.5 Conclusions

In this chapter, we examined the medical evacuation (MEDEVAC) location-allocation problem wherein military medical planners must decide where to locate mobile aeromedical staging facilities (MASFs) given a set of potential site locations and how many aeromedical helicopters to allocate to each MASF over the phases of a deployment. The intent of this research is to assess different location-allocation strategies that may improve the performance of a deployed Army MEDEVAC system and ultimately increase the survival probability of combat casualties. We developed an integer mathematical programming formulation of the MEDEVAC location-allocation problem which enables examination of a variety of problem features relating to different military medical planning scenarios. The ε-constraint Method was utilized to address the multi-objective problem of maximizing total expected coverage of demand, minimizing the maximum number of located MASFs in any deployment phase, and minimizing the the total number of MASF relocations throughout the deployment. The solution methodology defines trade-offs between competing objectives and substantially reduces the set of alternatives for military medical planners to consider when planning for and executing combat operations. Utilizing a notional, representative planning scenario based on high-intensity combat operations in southern Azerbaijan, we demonstrated the applicability of our model and illustrated the differences between the model's (multi-phase) optimal solution and the phase-specific optimal solutions that disregard concerns of MASF relocation limitations.

The results from the computational experiments reveal the trade-offs between the objectives considered. As the number of aeromedical helicopters available to allocate increases, the total expected coverage of demand increases. Moreover, the results highlight the conflicting nature between the objectives and show the trade-offs as restrictions applied to the second and third objectives are respectively tightened or relaxed. As the restrictions on the second and/or third objectives decrease, the total expected coverage of demand increases in a non-decreasing manner and vice versa.

The comparisons between the model's optimal solution and the phase-specific optimal solutions show the differences in MASF locations and total expected demand coverage. Whereas the phase-specific solutions offer more expected coverage compared to our model's solutions (i.e., 87.89% versus 80.69%, respectively), they do not consider relocation restrictions. More specifically, the phase-specific solutions present a greater operational burden via the relocation of seven MASFs throughout the deployment, which is a 600% increase compared to our model's solution for the baseline instance (i.e., one MASF relocation).

This research is of interest to both military and civilian medical planners. Medical planners can utilize our mathematical formulation and solution approach to compare different location-allocation strategies for a variety of planning scenarios. Whereas the process for implementing this research into active MEDEVAC operations may be difficult, the point of this research is to show that there exist operations research techniques that may improve MEDEVAC system performance. Moreover, this research can be extended by examining different objectives, parameter settings, and scenarios.

III. Approximate Dynamic Programming for Military Medical Evacuation Dispatching Policies

3.1 Introduction

One of the primary objectives of the Army Health System (AHS) is to evacuate combat causalities to medical treatment facilities (MTFs) in an effective, efficient, and responsive manner that saves casualties' lives. The Army casualty evacuation (CASEVAC) system and the Army medical evacuation (MEDEVAC) system are the two main options available to evacuate combat casualties. The Army CASEVAC system rapidly transports combat casualties from predetermined casualty collection points (CCPs) to MTFs via non-medical evacuation platforms *without* en route medical care. The Army MEDEVAC system rapidly transports combat casualties from predetermined CCPs to MTFs via dedicated, standardized medical evacuation platforms having onboard medical professionals who are equipped and prepared to provide necessary en route medical care. Combat casualties transported via the Army CASEVAC system may not be transported to the appropriate MTF and/or receive proper en route medical care, increasing their chances of long-term disabilities and death. As such, the AHS prioritizes the Army MEDEVAC system as the primary link between roles of medical care for casualties throughout combat operations (Department of the Army, 2016).

Whereas the Army MEDEVAC system utilizes both ground and aerial MEDEVAC platforms, this chapter focuses solely on aerial MEDEVAC platforms (i.e., aeromedical helicopters), as they are the predominant platform utilized in contemporary MEDEVAC operations. For example, approximately 91% of the United States (US) military combat casualties that occurred in Afghanistan between September 11, 2001 and March 31, 2014 were evacuated via aerial MEDEVAC (Kotwal et al., 2016). Aeromedical helicopters are better suited for MEDEVAC operations when compared to ground platforms due to their speed, versatility, and ability to travel across terrain in remote areas inaccessible to ground platforms (De Lorenzo, 2003). Aeromedical helicopters were first employed during the Korean Conflict, where they immediately became a highly valued asset within the MEDEVAC system. The US military recognizes the importance of aeromedical helicopters and continues to improve upon the capabilities of aeromedical helicopters. Such improvements contribute to the recent increases in combat casualty survivability rates. For example, the case fatality rate (CFR) (i.e., percentage of fatalities among all combat casualties) decreased from 19.1% in World War II to 15.8% in Vietnam. The CFR further decreased to 8.6% during the 13 years of conflict in Afghanistan ranging from 2001-2014. The decreased CFRs, or increased combat casualty survivability rates, are attributed to the advances in the capabilities of aeromedical helicopters and the resulting decrease in response time for combat casualties to receive proper medical care (Kotwal et al., 2016).

Military medical planners seek to design effective and efficient deployed MEDEVAC systems prior to major combat operations. An effective and efficient MEDEVAC system minimizes combat mortality, serves as a force multiplier, boosts the morale of deployed military personnel, and provides connectivity, as appropriate, between the AHS and the military health system (Department of the Army, 2016). Many important decisions must be considered when designing a MEDEVAC system. These decisions include determining where to locate both MEDEVAC staging areas and MTFs, establishing how many MEDEVAC units to allocate at each staging area, identifying an appropriate MEDEVAC dispatching policy, and recognizing when redeployment of MEDEVAC units is necessary and/or possible.

Identifying a dispatching policy that dictates which MEDEVAC unit to dispatch to a particular 9-line MEDEVAC request (i.e., a request for MEDEVAC support from a combat unit that contains nine standardized elements of information (Department of the Army, 2016)), commonly referred to as the MEDEVAC dispatching problem, is vital to the success of a deployed MEDEVAC system and is the primary focus of this chapter. The US military currently utilizes a *closest-available* dispatching policy wherein the dispatch authority dispatches the closest-available MEDEVAC unit to an incoming 9-line MEDEVAC request, regardless of the incoming request's characteristics (e.g., location and precedence level) or the MEDEVAC system state (e.g., location and availability of all MEDEVAC units). Moreover, the US military typically considers a MEDEVAC unit to be unavailable until it returns to its original staging area due to the challenges redeployment poses (e.g., refueling and resupplying requirements).

This chapter examines the MEDEVAC dispatching problem wherein a dispatching authority must decide which available MEDEVAC unit (if any) to dispatch to prioritized requests for service (i.e., 9-line MEDEVAC requests). We assume that the locations of MEDEVAC staging areas and MTFs are known, as are the allocation of MEDEVAC units to each staging area. Moreover, we assume that each MEDEVAC unit has the capability to satisfy all mission requirements of any submitted 9-line MEDEVAC request. Distinct from previous research efforts (e.g., Keneally et al. (2016); Rettke et al. (2016); Jenkins et al. (2018), and Robbins et al. (2018)), we consider a high-intensity combat scenario and allow redeployment (i.e., dispatching a MEDEVAC unit during an ongoing mission) when a MEDEVAC unit is returning to its respective staging area.

We develop a discounted, infinite-horizon Markov decision process (MDP) model of the MEDEVAC dispatching problem to maximize the expected total discounted reward attained by the system. The MDP model provides an appropriate framework for solving the MEDEVAC dispatching problem; however, the large size of the motivating problem instance yields an uncountable state space, rendering classical dynamic programming methods (e.g., value iteration or policy iteration) inappropriate. As such, we employ approximate dynamic programming (ADP) solution techniques to produce high-quality dispatching policies relative to the currently practiced (i.e., closest-available) dispatching policy. We develop and test two distinct ADP solution techniques that both utilize an approximate policy iteration (API) algorithmic framework. The first API algorithm utilizes least-squares temporal differences (LSTD) learning for policy evaluation, whereas the second API algorithm leverages neural network (NN) learning for policy evaluation. Given the MEDEVAC dispatching problem features, we define a set of basis functions to approximate the value function around the post-decision state for both of our proposed algorithms. We construct a notional, representative planning scenario based on high-intensity combat operations in southern Azerbaijan to demonstrate the applicability of our MDP model and to examine the efficacy of our proposed ADP solution techniques. Moreover, we design and conduct computational experiments to determine how selected problem features and algorithmic features impact the quality of solutions attained by our ADP-generated dispatching policies.

An important difference between this chapter and other papers in this research area is the incorporation of redeployment. This aspect gives the dispatching authority the ability to task a MEDEVAC unit to service incoming or queued requests directly after the MEDEVAC unit completes service at an MTF's co-located MEDEVAC staging area (i.e., completes refuel and re-equip of MEDEVAC supplies). This relaxes the restriction that MEDEVAC units must return to their own staging areas to refuel and re-equip after delivering combat casualties to an MTF prior to being tasked with another service request, which is a recognized limitation of previous work. Since MTFs are co-located with MEDEVAC staging areas, it is reasonable to assume that MEDEVAC units can refuel and re-equip at an MTF's co-located MEDEVAC staging area immediately after the MEDEVAC unit transfers combat casualties to the MTF staff, especially during high-intensity combat operations. In addition to the incorporation of redeployment, this chapter jointly considers the relevant problem features examined in earlier research efforts, including admission control, queueing, and explicit modeling of the number of casualties per casualty event. Lastly, this chapter demonstrates the improved efficacy of an NN-based ADP solution technique for the MEDEVAC dispatching problem, as compared to a solution technique previously offered in the literature (e.g., in Rettke et al. (2016)).

The remainder of this chapter is structured as follows. Section 3.2 provides a literature review of recent germane research concerning the civilian ambulance dispatching problem and the military MEDEVAC dispatching problem. Section 3.3 describes the MEDEVAC dispatching problem. Section 3.4 presents the MDP model formulation of the MEDEVAC dispatching problem. Section 3.5 proposes two ADP solution techniques for the MEDEVAC dispatching problem. Section 3.6 demonstrates the applicability of our MDP model and examines the efficacies of our proposed ADP solution techniques via computational experiments. Section 3.7 concludes the chapter.

3.2 Literature Review

The importance, sensitivity, and vital nature of the decision-making process for both civilian and military emergency medical service (EMS) response systems have been recognized and studied by many operations research scientists since the 1960s. Operations research techniques such as stochastic modeling, discrete optimization, and simulation have commonly been applied by researchers examining EMS response systems due to their ability to provide rigorous, defensible, and quantitative insights (Green and Kolesar, 2004). The primary areas of research in this field include determining where to locate servers (e.g., Toregas et al. (1971); Church and ReVelle (1974); Daskin and Stern (1981); Grannan et al. (2015),and Lejeune and Margot (2018)) and how many servers to allocate per location (e.g., Hall (1972); Berlin and Liebman (1974), and Baker et al. (1989)); establishing how to dispatch servers in response to service requests (e.g., Ignall et al. (1982); Green and Kolesar (1984); Majzoubi et al. (2012); Mayorga et al. (2013), and Bandara et al. (2014)); deciding when and where to relocate servers (if necessary) (e.g., Chaiken and Larson (1972); Kolesar and Walker (1974); Berman (1981); van Barneveld et al. (2016), and Sudtachat et al. (2016)); and identifying which performance measure to use to model casualty survivability rates (e.g., Erkut et al. (2008); McLay and Mayorga (2010), and Knight et al. (2012)). Indeed, the literature examining EMS systems is quite extensive. Herein, we briefly review only highly related previous research concerning the civilian ambulance dispatching problem and the related military medical evacuation (MEDEVAC) dispatching problem. We refer the reader to the survey papers by Swersey (1994), Brotcorne et al. (2003), Ingolfsson (2013), and Aringhieri et al. (2017) and the references therein for an extensive review of the related literature.

Decisions concerning which ambulance to dispatch to a request for service must be made sequentially over time and under uncertainty. Accordingly, many researchers utilize a dynamic programming approach to model the ambulance dispatching problem.

McLay and Mayorga (2013*b*) develop a Markov decision process (MDP) model of the ambulance dispatching problem. A novel problem feature they consider is the presence of patient prioritization classification errors within the EMS system. The authors utilize relative value iteration, a classical exact dynamic programming algorithm, to solve the dispatching problem for relatively small problem instances. Importantly, the authors include a simulation-based analysis as an excursion, showing that relaxing the assumption of exponentially distributed service times in their problem formulation has little impact on the optimal policy and attendant policy insights. McLay and Mayorga (2013*a*) modify the Markov decision problem introduced in McLay and Mayorga (2013*b*) to consider another interesting problem feature - - that of balancing equity and efficiency. For example, it may be optimal from a system-wide perspective to adopt dispatching policies that decrease performance in rural, low-population density districts in favor of increased performance in urban, high-population density districts. However, notions of equity may constrain the magnitude of acceptable performance decreases in some districts despite their positive impact on overall system performance. The authors formulate a linear programming model to solve a constrained ambulance dispatching problem, analyzing dispatch policies based on four different notions of equality. Both McLay and Mayorga (2013*b*) and McLay and Mayorga (2013*a*) do not consider redeployment (i.e., sending an ambulance that just completed service to a new location or call rather than allowing it to return to its predetermined base) nor do they consider relocation (i.e., moving an idle ambulance to a new base in anticipation of improving coverage of expected calls for service). Moreover, they neither allow the queueing of calls nor examine admission control. They assume all calls must be serviced, and if all ambulances are busy, then it is assumed that a nearby EMS system (exogenous to the model) services the call.

Whereas McLay and Mayorga (2013*b*) and McLay and Mayorga (2013*a*) provide meaningful insights concerning the ambulance dispatching problem, their problem formulations and attendant solution approaches only allow for the investigation of small-scale problem instances. The following three papers proffer formulations and solution approaches that allow for the investigation of large-scale problem instances.

Maxwell et al. (2010) model and solve an ambulance dispatching problem. More specifically, the authors seek to determine a high-quality ambulance redeployment policy -- the movement of ambulances that have just completed service at a hospital to another base (rather than simply return to its “home” base) -- to improve system performance. They do not consider dispatching decisions, instead enforcing a closest-available (i.e., myopic) dispatching policy when responding to calls. Moreover, they do not consider relocation -- the repositioning of idle ambulances already located at bases. Queueing of calls is allowed. The authors adopt an approximate dynamic programming (ADP) approach, utilizing an approximate policy iteration (API) algorithmic strategy to solve their ambulance dispatching (redeployment) problem. The value function is approximated via an affine combination of deliberately designed, problem-specific basis functions. They utilize a least-squares policy evaluation (LSPE) technique within their API algorithm to update their basis function coefficients. The authors demonstrate the efficacy of their approach by application to two metropolitan EMS response scenarios, achieving improved performance as compared to benchmark policies in both circumstances.

Schmid (2012) also models and solves an ambulance dispatching problem. The author jointly considers ambulance dispatching and redeployment decisions to improve system performance. Relocation is not considered, but queueing of calls is allowed. The author adopts an ADP solution approach, utilizing an approximate value iteration algorithmic strategy to solve their ambulance dispatching and redeployment problem. The value function is approximated via a spatial and temporal aggregation scheme. The author demonstrates the efficacy of their approach by application to a Vienna, Austria EMS response scenario, achieving improved performance as compared to the currently practiced policy.

Nasrollahzadeh et al. (2018) also model and solve an ambulance dispatching problem. The authors jointly consider ambulance dispatching, redeployment, and relocation decisions. Queueing of calls is allowed. Similar to Maxwell et al. (2010), the authors adopt an ADP approach, also utilizing an API algorithmic strategy to solve their ambulance dispatching, redeployment, and relocation problem. The value function is approximated via an affine combination of deliberately designed, problem specific basis functions. They also utilize an LSPE technique within their API algorithm to update their basis function coefficients. Moreover, the authors construct and exploit a lower bound on expected response time to improve algorithm performance. They demonstrate the efficacy of their approach by application to a Mecklenburg County, North Carolina EMS response scenario, achieving improved performance as compared to multiple benchmark policies.

Compared to the three aforementioned research endeavors, our research also jointly considers dispatching and redeployment decisions but in a military MEDEVAC context. The key differences extend beyond the problem application area and also include the specifics of the solution methodologies applied. Our ADP approach also utilizes an API algorithmic framework, but we develop a distinct set of basis functions, accounting for our slightly different problem structure. Moreover, we develop and compare two different policy evaluation mechanisms within our ADP algorithms, least-squares temporal differences (LSTD) learning and neural network (NN) learning, both of which are distinct from the LSPE and aggregation schemes employed by Maxwell et al. (2010), Schmid (2012), and Nasrollahzadeh et al. (2018). Moreover, we examine 30 problem instances herein to analyze our solution approaches and to gain generalizable policy insights rather than focus the testing on only one or two case studies.

While similarities exist between civilian and military EMS dispatching problems, several substantive differences remain that must be considered when examining the performance of a military EMS system. Military medical evacuation is often a more complex process, wherein the travel, load, and unload times are much greater and exhibit more variance (Jenkins et al., 2018). In a civilian EMS system, an ambulance crew may arrive at the scene of a call and determine that subsequent transport of the patient to a hospital is unwarranted. The ambulance then might either wait for further orders from its dispatching authority or travel to another waiting location. In a military EMS system, such situations do not occur. All calls result in casualties being transported to a medical treatment facility (MTF), and an opportunity for redeployment or repositioning from a field location does not exist because the cause of the casualty evacuation requirement may be an adversary who poses a continuing threat to the MEDEVAC itself, possibly across a geographic subregion. Moreover, routine relocation of idle aeromedical helicopter units is rare due to several resource and availability requirements (e.g., refueling, resupply, and armed escort). Another distinguishing problem feature of civilian and military EMS systems concerns how the system earns rewards, typically a function of response time. Unlike civilian EMS systems wherein the primary cause of death is typically cardiac arrest, the primary cause of death for military casualties is extreme blood loss (Garrett, 2013). As such, it is vital to stabilize and transport battlefield casualties to an appropriate MTF and into surgery rather than simply transporting medical personnel to the casualty collection point as rapidly as possible (Keneally et al., 2016). Accordingly, instead of defining response time as the time it takes an ambulance to reach the patient (as for civilian EMS systems), military EMS systems must define response time as the time it takes an aeromedical helicopter to pick up and then transport the casualties to an appropriate MTF. These key problem features impact the structure of the dispatching problem and attendant decision models, suggesting that successful solution methods (e.g., novel basis functions in an ADP approach) and resulting policy insights for military EMS systems -- the primary novel contributions of such military focused research -- differ enough from those of civilian EMS systems to warrant specific investigation.

With regard to a military MEDEVAC system planning context, researchers have spent most of their time examining the MEDEVAC location-allocation problem wherein medical planners must determine where to locate MEDEVAC staging areas and MTFs as well as decide how many aeromedical helicopters to allocate to each staging area. Solution methodologies for the MEDEVAC location-allocation problem typically attempt to balance maximizing demand coverage and minimizing response time subject to resource, force projection, and logistical constraints. Another important, but less studied problem within the military health care system is the MEDEVAC dispatching problem wherein a dispatching authority must decide which (if any) MEDEVAC unit (i.e., aeromedical helicopter) to dispatch in response to a submitted 9-line MEDEVAC request. The military currently utilizes a closest-available dispatching policy, which tasks the closest-available MEDEVAC unit to respond to service requests regardless of other factors (e.g., precedence, demand distribution, and request arrival rate). Many researchers (e.g., Carter et al. (1972); Nicholl et al. (1999), and Kuisma et al. (2004)) show that closest-available dispatching policies generally are not optimal. Moreover, the incorporation of precedence levels and other system factors (e.g., demand distribution and request arrival rate) into the construction of dispatching polices tends to improve the overall casualty survivability rates.

EMS research that focuses specifically on the military MEDEVAC dispatching problem exists but is relatively new to the field. To the best of our knowledge, Keneally et al. (2016) are responsible for the first paper that focuses solely on the military MEDEVAC dispatching problem. Keneally et al. (2016) utilize an MDP model to examine different MEDEVAC dispatching policies in the Afghanistan theater. The authors assume that each 9-line MEDEVAC request (i.e., service call) arrives sequentially according to a Poisson process and that the locations of MEDEVAC staging areas and MTFs are predetermined and do not change during combat operations. Their MDP model accounts for three different evacuation precedence categories: *urgent*, *priority*, and *routine*. Moreover, their MDP model accounts for other system factors such as the possibility that an armed escort may be required to escort aeromedical helicopters during MEDEVAC missions. The authors utilize a response time threshold (RTT) to model their MDP reward function and conduct computational experiments wherein MEDEVAC units operate in support of counterinsurgency operations in Afghanistan. The results identify that the default dispatching policy in practice, the closest-available policy, is not always optimal.

Several researchers expand upon the work of Keneally et al. (2016) by incorporating more realistic problem parameters (e.g., queueing and admission control) and examining large-scale scenarios that require approximation techniques. Jenkins et al. (2018) improve the fidelity of MEDEVAC dispatching models considered by Keneally et al. (2016) by incorporating admission control and queueing. Similar to Keneally et al. (2016), Jenkins et al. (2018) utilize an MDP model to examine the MEDEVAC dispatching problem. However, Jenkins et al. (2018) utilize a survivability function based on response time instead of an RTT to model their MDP reward function. The incorporation of admission control gives the dispatching authority the ability to reject incoming requests, thereby reserving MEDEVAC units for higher precedence requests instead of satisfying all requests for service. Moreover, the incorporation of queueing allows the dispatching authority to accept incoming requests regardless of the status of the MEDEVAC units and place them in a queue to be serviced at a later time. This differs from Keneally et al. (2016), who do not allow requests to be queued and simply reject requests if all MEDEVAC units are busy, thereby assuming they are then serviced by an exogenous resource (i.e., casualty evacuation (CASEVAC)). Jenkins et al. (2018) conduct a computational experiment based on counterinsurgency operations in Afghanistan similar to that examined by Keneally et al. (2016). The authors compare the optimal dispatching policies produced via their MDP model against three practitioner-friendly baseline policies. Their results align with the results yielded by Keneally et al. (2016), which show that current dispatching polices that are derived from a closest-available approach are suboptimal. Both Keneally et al. (2016) and Jenkins et al. (2018) examine small-scale computational experiments wherein their respective MDP models are able to generate optimal solutions in a tractable amount of time. However, these scenarios are not practical due to their small size, and while they do yield insights related to MEDEVAC dispatching polices, a practical (i.e., large-scale) scenario should be analyzed to give military medical planners more realistic insights. Unfortunately, the "curse of dimensionality" renders dynamic programming techniques intractable for the analysis of such scenarios.

Rettke et al. (2016) expand upon Keneally et al. (2016) by incorporating queueing and by examining large-scale scenarios via approximation techniques. Similar to Keneally et al. (2016), Rettke et al. (2016) utilize an MDP model to examine the MEDEVAC dispatching problem. The MDP model provides an appropriate framework for solving the MEDEVAC dispatching problem, but classical dynamic programming techniques are computationally intractable for large-scale instances due to their high dimensionality and uncountable state space. As such, Rettke et al. (2016) employ an ADP technique to determine high-quality dispatching policies. Their ADP technique involves an API algorithmic strategy that incorporates LSTD learning for policy evaluation. Moreover, Rettke et al. (2016) utilize a survivability function based on response time instead of an RTT to model their reward function. The authors utilize a large-scale computational experiment based on contingency operations in northern Syria to compare the ADP-generated dispatching policies against the default policy typically implemented in practice (i.e., closest-available dispatching policy). Their results indicate that the ADP policy outperforms the closest-available policy by over

30% with regard to a measure related to expected response time. These results support the notion that the current policy in practice is suboptimal, which aligns with the findings in Keneally et al. (2016).

Robbins et al. (2018) expand upon Keneally et al. (2016) by incorporating admission control and by examining large-scale scenarios via approximation techniques. Similar to Keneally et al. (2016), Robbins et al. (2018) utilize an MDP model to examine the MEDEVAC dispatching problem. As discussed above, the MDP gives an appropriate framework for solving the MEDEVAC dispatching problem, but classical dynamic programming techniques are computationally intractable in large-scale scenarios due to their high dimensionality and uncountable state space. As such, Robbins et al. (2018) employ ADP techniques to determine high-quality dispatching policies. Their ADP technique involves an API algorithmic strategy that utilizes a hierarchical aggregation value function approximation scheme. Moreover, Robbins et al. (2018) utilize a survivability function based on response time instead of an RTT to model their reward function. The authors utilize both small-scale and large-scale computational experiments based on contingency operations in Afghanistan to compare optimal dispatching policies, ADP-generated dispatching policies, and closest-available dispatching policies. Their results indicate that the ADP-generated policies are nearly optimal (i.e., within 1% optimal) for the small-scale experiments and outperform the closest-available policy in both the small-scale and large scale experiments by up to nearly 10% with regard to a measure related to expected response time. Similar to the findings of Keneally et al. (2016), Jenkins et al. (2018), and Rettke et al. (2016), these results indicate that the current MEDEVAC dispatching policy in practice is suboptimal.

This chapter seeks to build upon the current MEDEVAC dispatching problem research by utilizing an MDP model that incorporates the problem features previously examined (i.e., admission control and queueing) as well as redeployment, which has yet to be included in any model of the MEDEVAC dispatching problem. One of the limiting assumptions present in each of the aforementioned works is that MEDEVAC units must return to their own staging areas to refuel and re-equip after delivering combat casualties to an MTF prior to servicing another request. Redeployment gives a MEDEVAC unit the flexibility to service an incoming or queued request directly after the MEDEVAC unit completes service at an MTF's co-located MEDEVAC staging area (i.e., completes refuel and re-equip of MEDEVAC-related supplies). This assumption is reasonable to make when the receiving MTF is co-located with a MEDEVAC staging area, which is nearly always the case in practice, since MEDEVAC units can refuel and re-equip at the co-located MEDEVAC staging area immediately after they transfer combat casualties to the MTF. This chapter also builds upon the current MEDEVAC dispatching problem research by proposing and testing an NN-based ADP solution technique, which has not been previously applied in this research area.

3.3 Problem Description

In this section, we describe the military medical evacuation (MEDEVAC) dispatching problem in detail. The Army Health System (AHS) seeks to provide MEDEVAC capabilities across a wide range of military operations. One of the primary components of the AHS is the Army MEDEVAC system. The effectiveness of the Army MEDEVAC system is measured by how quickly combat casualties are transferred from the battlefield to medical treatment facilities (MTFs), which depends on the dispatching policy of MEDEVAC units (i.e., aeromedical helicopters) (Department of the Army, 2016). Identifying a MEDEVAC dispatching policy resulting in rapid evacuation of combat casualties from the battlefield to an MTF, commonly referred to as the MEDEVAC dispatching problem, is an important and vital task that military medical planners must consider prior to execution of combat operations. It is important to note that the primary cause of death on the battlefield is hemorrhage (i.e., severe blood loss), which is why the effectiveness of a MEDEVAC system is measured by the total time it takes to transfer casualties to an MTF rather than the time it takes MEDEVAC units to arrive at the casualty collection points (CCPs) (Malsby III et al., 2013). The effect of blood loss on mortality rates is sufficiently important that, in an effort to increase combat survivability rates, senior Army leaders established policies to equip MEDEVAC units with in-flight blood transfusion capabilities (Malsby III et al., 2013). A recent study on the MEDEVAC blood transfusion capabilities by Elster and Bailey (2017) indicates that there is not enough evidence to support a different effectiveness measure of MEDEVAC systems. As such, this research measures the effectiveness of a MEDEVAC system by utilizing the mission profile of evacuating combat casualties from CCPs to the nearest, suitable MTFs to receive the necessary level of medical treatment.

The dedicated aeromedical helicopters utilized in the Army MEDEVAC system are under the command of the general support aviation battalion (GSAB). The GSAB serves as the primary decision-making authority for the Army MEDEVAC system and is responsible for monitoring and synchronizing the execution of all aerial MEDEVAC operations (Department of the Army, 2016). Moreover, the GSAB operates as the dispatching authority and is responsible for managing all submitted 9-line MEDEVAC requests during combat operations. Dispatching decisions must be made quickly upon receipt of 9-line MEDEVAC requests since any delay in dispatching decisions may significantly decrease the chances of survival for combat casualties. As such, it is vital that the GSAB implements a dispatching policy resulting in high-quality and rapid transport of combat casualties from CCPs to appropriate MTFs to ensure the highest probability of survival.

Most 9-line MEDEVAC requests are transmitted over a dedicated MEDEVAC radio frequency with a prescribed amount of information regarding the casualty event. Such information includes the location of the pick-up site (i.e., point of injury (POI) or CCP), radio frequency and call sign, number of casualties by precedence, special equipment required, status and nationality of each casualty, and the threat level at the pick-up site. Note that in practice aeromedical evacuation assets can be dispatched directly to a POI instead of a CCP if the situation allows (i.e., little to no hostile fire is present). However, in this chapter, we assume that POI sites are located in unfavorable, unsecured locations and, as such, casualties must be transported to nearby, pre-determined CCPs for aeromedical evacuation. We assume CCPs are located in areas that are more secure and viable for helicopter landings, reducing the need for armed escorts and/or rescue hoists as well as reducing the chances of aeromedical assets being damaged by enemy combatants. It is a straightforward modification to consider a direct flight to a POI rather than a CCP, but it should be recognized that such a modification would alter MEDEVAC service and response times. The on-site medic (if available) or the senior military person in charge utilizes a three-category casualty triage scheme to determine the evacuation precedence category. *Urgent* and *priority* 9-line MEDEVAC requests are life-threatening requests that must be serviced within 1 hour and 4 hours, respectively. *Routine* 9-line MEDEVAC requests are not life-threatening requests, but still must be serviced within 24 hours to prevent further deterioration of health (Department of the Army, 2016).

Once a casualty event takes place and a 9-line MEDEVAC request is submitted, the GSAB makes an admission control decision. Admission control gives the GSAB the ability to observe the current state of the MEDEVAC system prior to making the decision to accept or reject submitted requests. If a submitted request is accepted (i.e., admitted) into the MEDEVAC system and there is at least one MEDEVAC unit available, then the GSAB makes a dispatching decision regarding which (if any) available MEDEVAC unit to task to service the accepted request. The GSAB has the ability to place accepted requests in a queue, even when a MEDEVAC unit is available for dispatch. The ability to place an accepted request in a queue regardless of the availability status of each MEDEVAC unit allows the dispatching authority to forgo immediately servicing a lower precedence request to ensure a MEDEVAC unit is available to service a likely, higher precedence request that is anticipated to occur in the near future. If a submitted request is admitted into the MEDEVAC system and there are no available MEDEVAC units, then the request will be placed in a queue. If the submitted request is rejected from the system, then the request is redirected to be handled by an exogenous organization via casualty evacuation (CASEVAC).

When a MEDEVAC unit completes service at an MTF's co-located MEDEVAC staging area (i.e., completes refuel and re-equip of MEDEVAC supplies) and there is at least one queued request, the GSAB must make a dispatching decision regarding which (if any) available MEDEVAC unit to task to service the queued request. Again, the GSAB is not required to task immediately an available MEDEVAC to service queued requests. Queued requests are serviced in order based first on their precedence category (i.e., *urgent*, *priority*, and *routine*) and second on their entry time into the MEDEVAC system. Figure 7 visually depicts the prioritized first-come-first-serve (FCFS) single-queue, multiple service MEDEVAC queueing system with admission control.

Once a MEDEVAC unit is tasked to service a particular 9-line MEDEVAC request, it must respond immediately, flying to the pre-determined CCP and then evacuating the combat casualties to the nearest MTF while onboard medical professionals provide necessary enroute medical care to the casualties. Each MEDEVAC mission can be

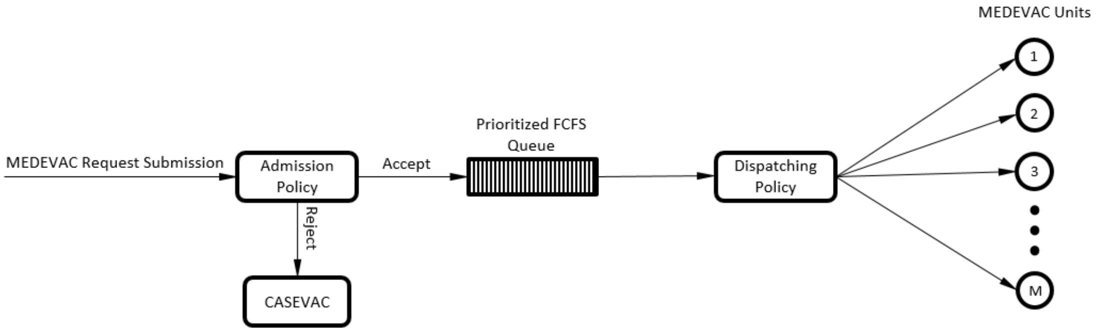


Figure 7. MEDEVAC Queueing System with Admission Control

broken down into eleven distinct and sequential events as reported in Table 3 (Rettke et al., 2016; Jenkins et al., 2018; Robbins et al., 2018). Unlike other papers in this research area, we assume that once a MEDEVAC unit completes Event 9 it becomes available to service incoming or queued requests. If there are no requests in the MEDEVAC system when a MEDEVAC unit completes Event 9, the MEDEVAC unit will travel back to its staging area. However, if a 9-line MEDEVAC request is submitted and admitted into the system, a MEDEVAC unit that is traveling back to its staging area is considered to be available, and the GSAB can task the MEDEVAC unit to service the submitted request.

Table 3. MEDEVAC Mission Distinct and Sequential Events

Event	Description
1	9-line MEDEVAC request submission.
2	MEDEVAC unit tasked to service a request.
3	MEDEVAC unit travels to CCP.
4	MEDEVAC unit arrives at CCP.
5	MEDEVAC unit travels from CCP to nearest MTF.
6	MEDEVAC unit arrives at nearest MTF.
7	MEDEVAC unit completes transfer of the combat casualties to the MTF staff
8	MEDEVAC unit travels from MTF to MTF's co-located MEDEVAC staging area.
9	MEDEVAC unit completes refuel and re-equip.
10	MEDEVAC unit travels from MTF's co-located MEDEVAC staging area to its dedicated staging area.
11	MEDEVAC unit arrives at its dedicated staging area.

The admission control and dispatching decisions concerning how 9-line MEDEVAC requests and MEDEVAC units are managed are complicated by the fact that future requests are not known *a prior*. Instead, the information regarding future requests only becomes known to the MEDEVAC system upon their occurrence. Both a dynamic and stochastic approach are needed when analyzing the MEDEVAC dispatching problem. The stochastic aspects of this problem arise from the uncertainty concerning the demand for service (i.e., casualty event arrivals, severity, and locations) as well as the variability in MEDEVAC unit dispatch, travel, and service times. In this research, we leverage information related to MEDEVAC dispatch, travel, and service times to parameterize the models. Moreover, we utilize stochastic simulation methods to model 9-line MEDEVAC request submissions to assist military medical planners in identifying an appropriate MEDEVAC dispatching policy *prior* to the execution of major combat operations.

3.4 MDP Formulation

This section describes the Markov decision process (MDP) model formulation of the medical evacuation (MEDEVAC) dispatching problem. The objective of the MDP model is to determine which (if any) available MEDEVAC unit (i.e., aeromedical helicopter) to dispatch in response to a 9-line MEDEVAC request to maximize the expected total discounted reward over an infinite horizon. Our model assumes that casualty events (i.e., 9-line MEDEVAC requests) arrive sequentially over time according to a Poisson process with parameter λ. Recall that a Poisson process has independent and stationary increments. The assumption of independent increments is reasonable in the context of MEDEVAC request arrivals because there are a large number of violent interactions that take place between small, widely dispersed groups of forces that result in localized requests for medical evacuation that are unrelated to one another, and therefore the numbers of arrivals that occur in disjoint time intervals are independent. Moreover, the assumption of stationary increments is also reasonable due to the underlying presumption that the implicit sizes, locations, and dispositions of forces generally remain fixed with respect to time. As such, the number of arrivals that occur in any interval of time depends only on the length of the time interval. Furthermore, it is important to note that a fixed, stationery λ-value can be interpreted as a parameter-value representing a particular season and time of day (e.g., summer during daylight hours). As such, military medical planners should utilize an appropriate λ-value to investigate peak activity for anticipated conditions when planning medical evacuation support.

Each arrival is characterized by the location (i.e., the coordinates of latitude and longitude), the precedence category (i.e., *urgent*, *priority*, and *routine*), and the number of casualties of the service request. Our MDP model utilizes a contribution (i.e., reward) function that is monotonically decreasing with respect to response time. That is, we assume that the MEDEVAC system earns greater rewards for servicing requests in lesser time. Moreover, the reward function accounts for the different precedence categories of requests by rewarding the service of higher precedence requests with higher rewards than the service of lower precedence requests. We define the response time for each MEDEVAC mission as $T_7 - T_1$, where T_7 and T_1 are the times at which Events 7 and 1 take place, respectively. We define the service time for each MEDEVAC mission as $T_9 - T_2$, where T_9 and T_2 are the times at which Events 9 and 2 take place, respectively. The MEDEVAC mission response and service times depend on the location of the casualty collection point (CCP) and the servicing MEDEVAC unit.

Having introduced the characteristics of the arrival process and the nature of the service times, we can now proceed with the MDP model formulation. We define and describe the components of our MDP model (i.e., the decision epochs, state space, action space, reward function, transition function, objective function, and optimality equation) in detail below.

The decision epochs in the MEDEVAC dispatching problem are the points in time wherein the dispatch authority is required to make a decision. Let $\mathcal{T} = \{1, 2, \ldots\}$ denote the set of decision epochs. A decision epoch is observed when one of two event types occurs. The first event type is the submission of a 9-line MEDEVAC request. The second event type is the change in a MEDEVAC unit's status from busy to available, which occurs once a MEDEVAC unit completes service at a medical treatment facility's (MTF's) co-located MEDEVAC staging area (i.e., completes refuel and re-equip of MEDEVAC supplies). That is, once a MEDEVAC unit completes the refuel and re-equip of MEDEVAC supplies at an MTF's co-located MEDEVAC staging area, a decision epoch occurs and the MEDEVAC unit becomes available to service requests.

At decision epoch $t \in \mathcal{T}$, the state $S_t \in \mathcal{S}$ gives the minimum description of the MEDEVAC system required to compute the decision function, transition function, and reward function. The MEDEVAC system state is represented by the tuple $S_t = (\tau_t, M_t, Q_t, \hat{R}_t)$ wherein τ_t represents the current system time at epoch t, M_t represents the MEDEVAC unit status tuple at epoch t, Q_t represents the queue status tuple at epoch t, and $\hat{R}_t$ represents the incoming service request status tuple at epoch t.

The MEDEVAC unit status tuple M_t describes the status of every MEDEVAC unit in the system at epoch t. The tuple M_t can be written as

$$M_t = (M_{tm})_{m \in \mathcal{M}} \equiv (M_{t1}, M_{t2}, \ldots, M_{t|\mathcal{M}|}),$$

where $\mathcal{M} = \{1, 2, \ldots, |\mathcal{M}|\}$ denotes the set of MEDEVAC units in the system and the tuple M_{tm} contains information pertaining to MEDEVAC unit $m \in \mathcal{M}$ at epoch t. The tuple M_{tm} can be written as

$$M_{tm} = (a_{tm}, e_{tm}, r_{tm}, \mu_{tm}, p_{tm}, c_{tm}),$$

wherein a_{tm} denotes the availability status of MEDEVAC unit m at epoch t, e_{tm} denotes the entry time into the MEDEVAC system of the request being serviced by MEDEVAC unit m at epoch t, r_{tm} denotes the expected system response time for the current MEDEVAC mission (i.e., combat casualties delivered to nearest MTF) of MEDEVAC unit m at epoch t, μ_{tm} denotes the expected system service time for the current MEDEVAC mission of MEDEVAC unit m at epoch t, p_{tm} denotes the precedence category of the request being serviced by MEDEVAC unit m at epoch t, and c_{tm} represents the number of combat casualties being serviced by MEDEVAC unit m at epoch t. We assume that MEDEVAC units refuel and re-equip at the MTF's co-located MEDEVAC staging area immediately after delivering combat casualties. As such, MEDEVAC units are available to service incoming or queued requests upon completing service at the MTF's co-located MEDEVAC staging area. MEDEVAC units return back to their staging areas after completing service at an MTF's co-located MEDEVAC staging area if they are not tasked to service another request. However, if a 9-line MEDEVAC request is submitted while a MEDEVAC unit is returning back to its staging area, that MEDEVAC unit can be tasked to service the incoming request because we assume that the MEDEVAC unit refueled and re-equipped at the MTF's co-located MEDEVAC staging area. We let $a_{tm} = 0$ if MEDEVAC unit m is idle and available at its staging area at epoch t, 1 if MEDEVAC unit m is unavailable (e.g., enroute to CCP, transferring casualties to MTF staff, or refueling) at epoch t, and 2 if MEDEVAC unit m is available and enroute to its original staging area (i.e., completed service at an MTF's co-located staging area).

The queue status tuple Q_t describes the status of every queued request in the system at epoch t. The tuple Q_t can be written as

$$Q_t = (Q_{tq})_{q \in \mathcal{Q}} \equiv (Q_{t1}, Q_{t2}, \ldots, Q_{t|\mathcal{Q}|}),$$

where $\mathcal{Q} = \{1, 2, \ldots, |\mathcal{Q}|\}$ denotes the set of queued requests in the system and the tuple Q_{tq} contains information pertaining to queued request $q \in \mathcal{Q}$ at epoch t. We let q^{max} denote the maximum number of requests that can be queued at any given time and, therefore, $|\mathcal{Q}| \leq q^{max}$ at any given time. The tuple Q_{tq} can be written as

$$Q_{tq} = (l_{tq}, \zeta_{tq}, \rho_{tq}, \kappa_{tq}),$$

wherein l_{tq} denotes the location of the queued request q at epoch t, ζ_{tq} denotes the entry time into the MEDEVAC system of queued request q at epoch t, ρ_{tq} denotes the precedence category of queued request q at epoch t, and κ_{tq} is the number of combat casualties within queued request q at epoch t. If there are no queued requests in the system at epoch t, then $Q_{t1} = (0, 0, 0, 0)$. The order in which queued requests are serviced is first based on precedence and second by the entry time. That is, higher precedence requests are placed in front of lower precedence request in the queue, and requests having the same precedence level are ordered based on their respective entry times.

The incoming service request status tuple $\hat{R}_t$ describes the status of an incoming request, if one is present, awaiting an admission decision at epoch t. The tuple $\hat{R}_t$ can be written as

$$\hat{R}_t = (\hat{l}_t^{sr}, \hat{p}_t^{sr}, \hat{c}_t^{sr}),$$

wherein the random variable $\hat{l}_t^{sr}$ denotes the location of the incoming service request at epoch t, the random variable $\hat{p}_t^{sr}$ denotes the precedence category of the incoming service request at epoch t, and the random variable $\hat{c}_t^{sr}$ represents the number of combat casualties within the incoming service request at epoch t. At epoch t, the information in $\hat{l}_t^{sr}$, $\hat{p}_t^{sr}$, and $\hat{c}_t^{sr}$ has just been realized and is no longer uncertain. However, $\hat{l}_t^{sr}$, $\hat{p}_t^{sr}$, and $\hat{c}_t^{sr}$ are random variables at epochs $1, 2, \ldots, t-1$ because the information they contain is still uncertain. Let $\hat{R}_t = (0, 0, 0)$ if there is not an incoming service request at epoch t (i.e., when the state transition occurs due to a change in a MEDEVAC unit's status from busy to available).

Events are generated by an arrival of a 9-line MEDEVAC request or a change in a MEDEVAC unit's status from busy to available. When a 9-line MEDEVAC request is submitted to the system, the dispatching authority must consider the current state of the MEDEVAC system as well as the location and precedence category of the submitted service request and quickly determine which (if any) available MEDEVAC unit to dispatch to service the request. The dispatching authority can only dispatch available MEDEVAC units that are allowed to traverse to the CCP. We assume that the MEDEVAC system employs an intra-zone policy regarding airspace access, which allows any MEDEVAC unit to service any request, regardless of the CCP location. Moreover, our model allows the dispatching authority to reject submitted requests regardless of the current state of the MEDEVAC system. If a submitted request is rejected from entering the MEDEVAC system, it is transferred to the regional command authority for a decision regarding how the request will be serviced (e.g., via ground MEDEVAC or non-medical air evacuation platforms) (Robbins et al., 2018). If a submitted request is accepted and there is at least one available MEDEVAC unit, then the general support aviation battalion (GSAB) must make a dispatching decision regarding which (if any) available MEDEVAC unit to task to service the accepted request. As we discussed in the previous section, the GSAB is not required to task available MEDEVAC units to service a request. Instead, the GSAB may forgo servicing a lower precedence request in anticipation of servicing a likely, higher precedence request in the near future. If a submitted request is accepted and there are no available MEDEVAC units, then the request is placed in a queue. Recall that the order at which queued requests are serviced is first based on precedence and then by the entry time.

The admission control decision is represented by the decision variable $x_t^{reject} \in \{\Delta, 0, 1\}$ at epoch t. When $\hat{R}_t = (0,0,0)$ (i.e., there is no incoming service request present in the system) then $x_t^{reject} = \Delta$. When $x_t^{reject} = 0$, the incoming service request is admitted to the MEDEVAC system whereas, when $x_t^{reject} = 1$, the incoming service request is rejected from entering the MEDEVAC system.

The dispatching decision is represented by the tuple $x_t^d = (x_t^{sr}, x_t^{qr})$, wherein x_t^{sr} represents the incoming service request dispatch decision tuple and x_t^{qr} represents the queued request dispatch decision tuple at epoch t. A dispatching decision may be necessary when either a 9-line MEDEVAC request is submitted to the system or the status of a MEDEVAC unit changes from busy to available. Let $\mathcal{A}(S_t) = \{m : m \in \mathcal{M}, a_{tm} \neq 1\}$ denote the set of available MEDEVAC units when the state of the system is S_t at epoch t.

The incoming service request dispatch decision tuple x_t^{sr} describes the dispatching authority's decision regarding which MEDEVAC unit (if any) to dispatch to the incoming service request at epoch t. The tuple x_t^{sr} can be written as

$$x_t^{sr} = (x_{tm}^{sr})_{m \in \mathcal{A}(S_t)}.$$

The decision variable $x_{tm}^{sr} = 1$ if MEDEVAC unit $m \in \mathcal{A}(S_t)$ is dispatched to service the incoming service request at epoch t, and 0 otherwise.

The queued request dispatch decision tuple x_t^{qr} describes the dispatching authority's decision with regard to which MEDEVAC unit (if any) to dispatch to the first queued request (i.e., Q_{t1}) at epoch t. The tuple x_t^{qr} can be written as

$$x_t^{qr} = (x_{tm}^{qr})_{m \in \mathcal{A}(S_t)}.$$

The decision variable $x_{tm}^{qr} = 1$ if MEDEVAC unit $m \in \mathcal{A}(S_t)$ is dispatched to service Q_{t1} at epoch t, and 0 otherwise.

Let $x_t = (x_t^{reject}, x_t^d)$ denote a compact representation of the admission control and dispatching decision variables at epoch t. At each epoch t, the dispatching authority is bounded by two constraints. Given the definition of an indicator variable as $I_R = 1$ if $\hat{R}_t \neq (0,0,0)$, 0 otherwise, the first constraint,

$$\sum_{m \in \mathcal{A}(S_t)} (I_R x_{tm}^{sr} + x_{tm}^{qr}) \leq 1, \tag{20}$$

limits the dispatching authority to dispatch at most one MEDEVAC unit at epoch t. The next constraint,

$$x_t^{reject} + \sum_{m \in \mathcal{A}(S_t)} x_{tm}^{sr} \leq 1, \tag{21}$$

forces the dispatching authority to accept incoming service requests (i.e., $x_t^{reject} = 0$) if a MEDEVAC unit is tasked to service the incoming service request (i.e., $x_{tm}^{sr} = 1$ for some $m \in \mathcal{A}(S_t)$).

By letting $I_A = 1$ if $\mathcal{A}(S_t) \neq \emptyset$, 0 otherwise and letting $I_Q = 1$ if $|\mathcal{Q}| \neq 0$, 0 otherwise, the set of feasible actions when a decision is required can be written as

$$\mathcal{X}(S_t) = \begin{cases} \big(\Delta, (\{0\}^{|\mathcal{A}(S_t)|}, \{0,1\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 0, I_A = 1, I_Q = 1 \\ \big(\Delta, (\{0\}^{|\mathcal{A}(S_t)|}, \{0\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 0, I_A = 1, I_Q = 0 \\ \big(1, (\{0\}^{|\mathcal{A}(S_t)|}, \{0,1\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 1, I_A = 1, |\mathcal{Q}| = q^{max} \\ \big(1, (\{0\}^{|\mathcal{A}(S_t)|}, \{0\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 1, I_A = 0, |\mathcal{Q}| = q^{max} \\ \big(\{0,1\}, (\{0,1\}^{|\mathcal{A}(S_t)|}, \{0,1\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 1, I_A = 1, I_Q = 1, |\mathcal{Q}| < q^{max} \\ \big(\{0,1\}, (\{0\}^{|\mathcal{A}(S_t)|}, \{0\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 1, I_A = 0, |\mathcal{Q}| < q^{max} \\ \big(\{0,1\}, (\{0,1\}^{|\mathcal{A}(S_t)|}, \{0\}^{|\mathcal{A}(S_t)| \times |\mathcal{Q}|})\big), & \text{if } I_R = 1, I_A = 1, I_Q = 0 \end{cases} \tag{22}$$

where Constraints (20) and (21) must be satisfied. The first two cases in Equation (22) represent all feasible actions when an event is triggered due to a change in a MEDEVAC unit's status from busy to available, whereas the last five cases represent all feasible actions when an event is triggered due to a 9-line MEDEVAC request submission.

The next component of the MEDEVAC MDP model is the *transition function.* This function describes how the MEDEVAC system evolves from one state to another as new information arrives and decisions are made. The state transition function $S_{t+1} = S^M(S_t, x_t, W_{t+1})$ represents the dynamics of the MEDEVAC system, wherein the state of the system at the beginning of epoch $t+1$ (i.e., S_{t+1}) is determined by the state of the system at epoch t (i.e., S_t), the decision that is made at epoch t (i.e., x_t), and the information that arrives at epoch $t+1$ (i.e., W_{t+1}).

The MEDEVAC system earns rewards when the dispatching authority tasks MEDEVAC units to service incoming or queued 9-line MEDEVAC requests. There are several factors that impact the amount of reward attained by the system, e.g., the location and precedence category of the request being serviced as well as the location of the servicing MEDEVAC unit. Let $C(S_t, x_t)$ denote the reward (i.e., contribution) attained by the MEDEVAC system if MEDEVAC unit $m \in \mathcal{M}$ is tasked to service a precedence k (i.e., *urgent*, *priority*, or *routine*) 9-line MEDEVAC request, 0 otherwise. That is, the MEDEVAC system only earns rewards when MEDEVAC units are tasked to service requests. When a MEDEVAC unit is tasked to service a request, the reward attained by the MEDEVAC system is computed as follows

$$C(S_t, x_t) = w_k c_{tm} \delta(r_{tm}, e_{tm}),$$

wherein w_k is a tradeoff parameter that varies the reward attained based on $k \in \mathcal{K} = \{1, 2, 3\}$ (i.e., the precedence category of the request being serviced), c_{tm} is the number of combat casualties within the request being serviced, m is the MEDEVAC unit tasked to service the request, and $\delta(r_{tm}, e_{tm})$ is a utility function that is monotonically decreasing with respect to the system response time and entry time of the request being serviced. The elements contained in $\mathcal{K} = \{1, 2, 3\}$ correspond to the casualty event precedence categories *urgent*, *priority*, and *routine*, respectively.

We let $X^{\pi}(S_t)$ represent the decision function that returns a decision, x_t, for each state $S_t \in \mathcal{S}$ based on a given policy, π. Our MDP model seeks to find the optimal policy, π^*, from the class of policies $(X^{\pi}(S_t))_{\pi\in\Pi}$ to maximize the expected total discounted reward earned by the MEDEVAC system. The objective of the MDP model can be written as

$$\max_{\pi\in\Pi} \mathbb{E}^{\pi}\left[\sum_{t=1}^{\infty} \gamma^{\tau_t} C\left(S_t, X^{\pi}(S_t)\right)\right],$$

where $\gamma \in [0, 1)$ is a fixed discount factor and τ_t is the time at which the system visits state S_t. The optimal policy π^* is found using the Bellman Equation

$$V(S_t) = \max_{x_t\in\mathcal{X}(S_t)} \left(C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1})-\tau_t)}\mathbb{E}\left[V(S_{t+1})|S_t, x_t\right]\right), \tag{23}$$

wherein $\hat{\tau}(S_{t+1})$ denotes the time at which the system visits state S_{t+1}.

Unfortunately, the high dimensionality and uncountable state space of our MDP model makes computing an optimal policy using Equation (23) intractable. Instead, in the following section we propose two approximate dynamic programming (ADP) techniques that utilize value function approximation schemes to attain high-quality dispatching policies relative to the currently practiced closest-available dispatching policy. The ADP-generated policies are compared with the closest-available dispatching policy, applied to a realistic scenario in Section 3.6.

3.5 ADP Formulation

This section presents two approximate dynamic programming (ADP) solution techniques for the military medical evacuation (MEDEVAC) dispatching problem. Both of our ADP techniques approximate the value function, Equation (23), via a post-decision state convention due to its computational advantages (Powell, 2011; Ruszczynski, 2010). The computational advantages of utilizing a post-decision state convention are two-fold: first, it allows us to avoid computing expectations explicitly; and second, it substantially reduces the dimensionality of the state space. The post-decision state S_t^x refers to the state of the MEDEVAC system immediately after the system is in pre-decision state S_t and action x_t is taken. With this information, we proceed by modifying the optimality equation to incorporate the post-decision state convention. Let

$$V^x(S_t^x) = \mathbb{E}\left[V(S_{t+1})|S_t^x\right] \tag{24}$$

denote the value of being in post-decision state S_t^x. By substituting Equation (24) into Equation (23), the optimality equation is given as follows

$$V(S_t) = \max_{x_t \in \mathcal{X}(S_t)} \big(C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1})-\tau_t)} V^x(S_t^x)\big). \tag{25}$$

By recognizing that the value of being in post decision state S_{t-1}^x is given by

$$V^x(S_{t-1}^x) = \mathbb{E}\left[V(S_t)|S_{t-1}^x\right], \tag{26}$$

and substituting Equation (25) into Equation (26), the optimality equation around the post-decision state is given as follows

$$V^x(S_{t-1}^x) = \mathbb{E}\left[\max_{x_t \in \mathcal{X}_{S_t}} \big(C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1})-\tau_t)} V^x(S_t^x)\big)\big|S_{t-1}^x\right]. \tag{27}$$

Despite the computational advantages of the post-decision state convention, Equation (27) remains computationally intractable due to the size of our motivating problem instance. As such, both of our ADP techniques utilize a value function approximation approach via a fixed set of basis functions to determine approximate solutions to Equation (27). The key challenge in our approximation scheme is the identification of basis functions and features that are important to the MEDEVAC dispatching problem.

The selection of appropriate basis functions and features for the MEDEVAC dispatching problem is difficult but is necessary to attain high-quality dispatching policies. We leverage Rettke et al. (2016) and Maxwell et al. (2010) to design and develop eight conceptually motivated basis functions. Let $\phi_f(S_t^x)$ be a basis function, where $f \in \mathcal{F}$ is a feature and $\mathcal{F}$ is the set of features. The first basis function describes the availability status of each MEDEVAC unit in the system and is written as

$$\phi_{1m}(S_t^x) = a_{tm}, \quad \forall\, m \in \mathcal{M}.$$

The next four basis functions capture information pertaining to the 9-line MEDEVAC requests currently being serviced. The second basis function captures the expected time from the current system time τ_t until MEDEVAC unit m transfers care of onboard casualties to the nearest MTF staff and is written as

$$\phi_{2m}(S_t^x) = \begin{cases} r_{tm} - \tau_t, & \text{if } a_{tm} = 1, \tau_t < r_{tm} \\ 0, & \text{otherwise.} \end{cases}, \quad \forall\, m \in \mathcal{M}.$$

The third basis function captures the expected time from the current system time

τ_t until MEDEVAC unit m completes service and is written as

$$\phi_{3m}(S_t^x) = \begin{cases} \mu_{tm} - \tau_t, & \text{if } a_{tm} = 1 \\ 0, & \text{otherwise.} \end{cases}, \quad \forall\, m \in \mathcal{M}.$$

The fourth basis function captures the precedence category of the 9-line MEDEVAC request being serviced by MEDEVAC unit m and is written as

$$\phi_{4m}(S_t^x) = \begin{cases} p_{tm}, & \text{if } a_{tm} = 1, \tau_t < r_{tm} \\ 0, & \text{otherwise.} \end{cases}, \quad \forall\, m \in \mathcal{M}.$$

The fifth basis function captures the number of casualties being serviced by MEDEVAC unit m and is written as

$$\phi_{5m}(S_t^x) = \begin{cases} c_{tm}, & \text{if } a_{tm} = 1, \tau_t < r_{tm} \\ 0, & \text{otherwise.} \end{cases}, \quad \forall\, m \in \mathcal{M}.$$

The last three basis functions capture information about the 9-line MEDEVAC requests in the queue. The sixth basis function captures the expected total time, including wait time (i.e., time in queue), travel times, and service times (i.e, unload times and load times), that request $q \in \mathcal{Q}$ will incur in the MEDEVAC system if it is serviced by MEDEVAC unit m and is written as

$$\phi_{6qm}(S_t^x) = \psi_{tqm} - e_{tq}, \quad \forall\, q \in \mathcal{Q},\, \forall\, m\ \in \mathcal{M},$$

wherein ψ_{tqm} represents the expected system response time if MEDEVAC unit m is tasked to service queued request q at epoch t.

The seventh basis function captures the precedence category of each queued request and is written as

$$\phi_{7q}(S_t^x) = \rho_{tq}, \quad \forall\, q \in \mathcal{Q}.$$

The last basis function captures the number of casualties contained in each queued request and is written as

$$\phi_{8q}(S_t^x) = \kappa_{tq}, \quad \forall\, q \in \mathcal{Q}.$$

3.5.1 Least-Squares Temporal Differences

The first ADP solution technique we propose utilizes an approximate policy iteration (API) algorithmic strategy that incorporates least-squares temporal differences (LSTD) learning for policy evaluation. API is an algorithmic approach derived from exact policy iteration, wherein a sequence of policies and associated approximate value functions are produced in two repeated, alternating phases: policy evaluation and policy improvement. Within a policy evaluation phase (i.e., inner loop), our LSTD-API algorithm approximates the value function for a fixed policy π via simulation and LSTD learning. Within a policy improvement phase (i.e., outer loop), the algorithm generates a new policy based on the data collected in the immediately preceding policy evaluation phase. Before presenting our LSTD-API algorithm, we define our post-decision state value function approximation scheme, which leverages the basis functions we developed.

For our LSTD-API algorithm, let

$$\bar{V}^x(S_t^x|\theta) = \sum_{f\in\mathcal{F}} \theta_f \phi_f(S_t^x) \equiv \theta^T \phi(S_t^x) \tag{28}$$

denote the linear approximation architecture, wherein $\theta = (\theta_f)_{f\in\mathcal{F}}$ is a column vector of basis function weights and $\phi(S_t^x)$ is a column vector of basis function evaluations. For a given vector θ, which represents a fixed policy π, decisions are made utilizing the policy (i.e., decision function)

$$X^{\pi}(S_t|\theta) = \operatorname*{argmax}_{x_t \in \mathcal{X}_{S_t}} \big\{C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1})-\tau_t)} \bar{V}^x(S_t^x|\theta)\big\}. \tag{29}$$

Substituting Equations (28) and (29) into Equation (27), the approximate post-decision state value function is given as follows

$$\theta^T \phi(S_{t-1}^x) = \mathbb{E}\Big[C\big(S_t, X^{\pi}(S_t|\theta)\big) + \gamma^{(\hat{\tau}(S_{t+1})-\tau_t)} \theta^T \phi(S_t^x) \big| S_{t-1}^x\Big]. \tag{30}$$

Having defined the approximate post-decision state value function, we proceed with the presentation of our LSTD-API algorithm we employ herein, which is adapted in part from research by Rettke et al. (2016) and is displayed in Algorithm 1.

Algorithm 1 Least-Squares Temporal Differences Approximate Policy Iteration (LSTD-API) Algorithm

```
Initialize $\theta$.
for $n = 1$ to $N$ do
    for $j = 1$ to $J$ do
        Generate a random post-decision state, $S_{t-1,j}^x$.
        Record basis function evaluation $\phi(S_{t-1,j}^x)$.
        Simulate transition to next pre-decision state, $S_{t,j}$.
        Determine decision $x_t$ utilizing Equation (29).
        Record contribution $C(S_{t,j}, x_t)$.
        Record discount factor $\gamma^{(\hat{\tau}(S_{t+1,j})-\tau_t)}$.
        Record basis function evaluation $\phi(S_{t,j}^x)$.
    end for
    Update $\theta$ utilizing Equations (31) and (33).
end for
Return the approximate value function $\bar{V}^x(\cdot|\theta)$.
```

The LSTD-API algorithm starts by initializing the basis function weight vector θ, which represents an initial fixed policy. We then begin a policy evaluation phase wherein, for each iteration $j = 1, 2, \ldots, J$, the following steps occur. We randomly select a post-decision state $S_{t-1,j}^x$ and record the associated basis function evaluation

$\phi(S^x_{t-1,j})$. Next, we simulate the system evolving from post-decision state $S^x_{t-1,j}$ to a pre-decision state $S_{t,j}$ and determine the best decision x_t via Equation (29). Once x_t is determined, we record the contribution $C(S_{t,j}, x_t)$, discount factor $\gamma^{(\hat{\tau}(S_{t+1,j})-\tau_{t,j})}$, and basis function evaluation $\phi(S^x_{t,j})$. We collect a total of J temporal difference sample realizations in a single policy evaluation phase, where $C\big(S_{t,j}, X^\pi(S_{t,j}|\theta)\big) + \gamma^{(\hat{\tau}(S_{t+1,j})-\tau_{t,j})}\theta^T\phi(S^x_{t,j}) - \theta^T\phi(S^x_{t-1,j})$ is the jth temporal difference given the basis function weight vector θ.

We proceed into a policy improvement phase wherein, for each iteration $n = 1, 2, \ldots, N$, the following steps occur. The vector $\hat{\theta}$, a sample estimate of θ, is computed via least-squares regression. We seek a basis function weight vector $\hat{\theta}$ that makes the sum of the J temporal differences equal to zero. To provide a more compact representation of our basis function evaluations, discounts, and contributions, we define basis function matrices Φ_{t-1} and Φ_t, a discount matrix Γ_t, and a contribution vector C_t. More precisely, let

$$\Phi_{t-1} = \begin{bmatrix} \phi(S^x_{t-1,1})^\top \\ \vdots \\ \phi(S^x_{t-1,J})^\top \end{bmatrix}, \quad \Phi_t = \begin{bmatrix} \phi(S^x_{t,1})^\top \\ \vdots \\ \phi(S^x_{t,J})^\top \end{bmatrix}, \quad \Gamma_t = \begin{bmatrix} \gamma^{\left(\hat{\tau}(S_{t+1,1})-\tau_{t,1}\right)}\mathbf{1}_{1\times|\mathcal{F}|} \\ \vdots \\ \gamma^{\left(\hat{\tau}(S_{t+1,J})-\tau_{t,J}\right)}\mathbf{1}_{1\times|\mathcal{F}|} \end{bmatrix}, \quad C_t = \begin{bmatrix} C(S_{t,1}) \\ \vdots \\ C(S_{t,J}) \end{bmatrix},$$

wherein the rows and columns in the basis function matrices respectively correspond to sampled post-decision states and basis function evaluations, the rows of the discount matrix are the recorded discounts for the sampled post-decision states, and the elements of the contribution vector are the recorded contribution values. We let $\mathbf{1}_{1\times|\mathcal{F}|}$ denote a row vector of $|\mathcal{F}|$ ones. Utilizing this notation, the sample estimate of θ is computed via the normal equation as follows

$$\hat{\theta} = \left[(\Phi_{t-1} - \Gamma_t \odot \Phi_t)^\top(\Phi_{t-1} - \Gamma_t \odot \Phi_t) + \eta\mathbf{I}\right]^{-1}(\Phi_{t-1} - \Gamma_t \odot \Phi_t)^\top C_t, \tag{31}$$

wherein $\odot$ is the Hadamard product operator and $\eta\mathbf{I}$ is an $|\mathcal{F}| \times |\mathcal{F}|$ diagonal matrix having diagonal penalty entries given by the regularization parameter $\eta \geq 0$. We utilize regularization (i.e., ridge regression here) when computing $\hat{\theta}$ to avoid matrix inversion difficulties (by ensuring $(\Phi_{t-1} - \Gamma_t \odot \Phi_t)^\top(\Phi_{t-1} - \Gamma_t \odot \Phi_t)$ is nonsingular) and to reduce generalization error (by ensuring we do not overfit the data collected in any single policy evaluation iteration). Moreover, when there are many correlated variables in a linear regression model (a common occurrence in a simulation-based solution approach such as ours), its coefficients can become poorly determined and exhibit high variance (Hastie et al., 2009). For example, a very large positive coefficient on one feature can be offset by a similarly large negative coefficient on a correlated feature. Ridge regression shrinks the regression coefficients by imposing a penalty on their size, which helps prevent such unwanted variance in the θ-coefficients over multiple policy improvement iterations.

Once $\hat{\theta}$ is computed, we utilize a polynomial stepsize rule to smooth in $\hat{\theta}$ with the previous estimate θ. The stepsize rule is given by

$$\alpha_n = \frac{1}{n^\beta}, \tag{32}$$

wherein $\beta \in (\frac{1}{2}, 1]$. The polynomial stepsize rule α_n is an extension of the basic harmonic sequence and greatly impacts our algorithm's rate of convergence and attendant solutions. The rate at which α_n declines as the policy improvement iteration counter n increases depends on the value of β. Smaller values of β slow the rate at which α_n declines; however, the best value of β depends on the problem at hand and, as such, is a parameter that must be tuned (Powell, 2011).

Next, we update θ with the following equation

$$\theta \leftarrow \alpha_n\hat{\theta} + (1-\alpha_n)\theta, \tag{33}$$

wherein the θ on the right hand side is the previous estimate based on previous policy improvement iterations, and $\hat{\theta}$ is our new estimate from the current iteration. As the number of iterations n increases, we place less emphasis on sample estimates (i.e., $\hat{\theta}$) and more emphasis on the estimate based on the first $n-1$ iterations (i.e., θ).

Once θ is updated via Equation (33), we have completed one policy improvement iteration of the LSTD-API algorithm. If $n < N$ then the algorithm continues by starting another policy evaluation phase. The algorithmic parameters N, J, η, and β are tunable, where N is the number of iterations of the policy improvement phase, J is the number of iterations of the policy evaluation phase, η is the regularization term within the θ estimate computation, and β is the polynomial stepsize parameter.

3.5.2 Neural Network

The second ADP solution technique we propose also utilizes an API algorithmic strategy, but instead incorporates neural network (NN) learning for policy evaluation. Recall that API is an algorithmic approach derived from exact policy iteration, wherein a sequence of policies and associated approximate value functions are produced in two repeated, alternating phases: policy evaluation and policy improvement. Within a policy evaluation phase (i.e., inner loop), our NN-API algorithm approximates the value function for a fixed policy π via simulation and NN learning. Within a policy improvement phase (i.e., outer loop), the algorithm generates a new policy based on the data collected in the immediately preceding policy evaluation phase. Before presenting our NN-API algorithm, we define our NN-based, post-decision state value function approximation scheme, which also leverages the basis functions presented at the beginning of Section 3.5.

To approximate the value of being in post-decision state S_t^x, we utilize a feed-forward NN comprised of three layers: an input layer, a hidden layer, and an output layer. The information provided to the input layer is a set of $|\mathcal{F}|$ basis function evaluations associated with a post-decision state S_t^x. The hidden layer consists of a set of activation units $\mathcal{H} = \{1, 2, \ldots, |\mathcal{H}|\}$ (i.e., nonlinear perceptron nodes). The size of the hidden layer, $|\mathcal{H}|$, is a tunable characteristic of our NN-API algorithm. Because the hidden layer requires $|\mathcal{H}|$ inputs, the input layer produces $|\mathcal{H}|$ outputs, which are given by

$$Y_h^{(2)}(S_t^x) = \sum_{f \in \mathcal{F}} \Theta_{f,h}^{(1)} \phi_f^s(S_t^x), \quad \forall\, h \in \mathcal{H}, \tag{34}$$

wherein $\Theta^{(1)} \equiv \left[\Theta_{f,h}^{(1)}\right]_{f \in \mathcal{F}, h \in \mathcal{H}}$ is an $|\mathcal{F}| \times |\mathcal{H}|$ matrix of weights controlling the function mapping from the input layer to the hidden layer. A nonlinear logistic sigmoid activation function

$$\sigma(y) = \frac{1}{1 - e^{-y}}$$

is applied to each $Y_h^{(2)}(S_t^x)$ to produce the inputs for the hidden layer, which are given by

$$Z_h^{(2)}(S_t^x) = \sigma\big(Y_h^{(2)}(S_t^x)\big), \quad \forall\, h \in \mathcal{H}. \tag{35}$$

The hidden layer produces a single, scalar output, which is given by

$$Y^{(3)}(S_t^x) = \sum_{h \in \mathcal{H}} \Theta_h^{(2)} Z_h^{(2)}(S_t^x), \tag{36}$$

wherein $\Theta^{(2)} \equiv \left[\Theta_h^{(2)}\right]_{h \in \mathcal{H}}$ is an $|\mathcal{H}| \times 1$ matrix of weights controlling the function mapping from the hidden layer to the output layer. The output layer produces a single, scalar output by applying the sigmoid activation function σ to $Y^{(3)}(S_t^x)$. This computation results in the post-decision state value function approximation given by

$$\bar{V}^x(S_t^x|\Theta) = \sigma\big(Y^{(3)}(S_t^x)\big), \tag{37}$$

wherein $\Theta = \big(\Theta^{(1)}, \Theta^{(2)}\big)$ is the parameter tuple that compactly represents the NN weights.

Note that, for application within the NN model, we scale the $|\mathcal{F}|$ basis function evaluations via a mean normalization procedure. That is, for each feature, we transform each of its values by first subtracting its mean and then dividing by its range. Scaling the inputs provides benefits (Hastie et al., 2009); it ensures all input dimensions are treated equally in our regularization process, allows selection of meaningful initial weights, and enables more effective optimization when we update the NN weights. We denote the scaled basis functions by a superscript s (e.g., in Equation (34)).

For a given tuple Θ, our NN-API algorithm makes decisions utilizing the policy (i.e., decision function)

$$X^{\pi}(S_t|\Theta) = \underset{x_t \in \mathcal{X}_{S_t}}{\operatorname{argmax}} \big\{ C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1}) - \tau_t)} \bar{V}^x(S_t^x|\Theta) \big\}. \tag{38}$$

Substituting Equations (37) and (38) into Equation (27), the approximate post-decision state value function is given as follows

$$\bar{V}^x(S_{t-1}^x|\Theta) = \mathbb{E}\left[C\big(S_t, X^{\pi}(S_t|\Theta)\big) + \gamma^{(\hat{\tau}(S_{t+1}) - \tau_t)} \bar{V}^x(S_t^x|\Theta) \big| S_{t-1}^x \right]. \tag{39}$$

Having defined the NN-based, approximate post-decision state value function, we proceed with the presentation of our NN-API algorithm we employ herein, which is displayed in Algorithm 2.

The NN-API algorithm starts by initializing Θ, the function mapping weight matrices $\Theta^{(1)}$ and $\Theta^{(2)}$, an initial fixed policy. We select small, random values near zero for the initial weights, as recommended by Hastie et al. (2009). This weight initialization policy enables better NN model performance when the weights are up-

Algorithm 2 Neural Network Approximate Policy Iteration (NN-API) Algorithm

```
Initialize $\Theta$ with small, random values near zero.
for $n = 1$ to $N$ do
    for $j = 1$ to $J$ do
        Generate a random post-decision state, $S^x_{t-1,j}$.
        Compute basis function evaluation $\phi^s(S^x_{t-1,j})$.
        Compute $\bar{V}^x(S^x_{t-1,j}|\Theta)$ utilizing Equations (34)-(37). Record observed value.
        Simulate transition to next pre-decision state, $S_{t,j}$.
        Compute $\hat{v}_j$ utilizing Equation (40). Record observed value.
    end for
    Normalize the recorded value function realizations, $(\hat{v}_j)_{j=1}^{J}$.
    Update $\Theta^{(1)}$ and $\Theta^{(2)}$ utilizing Equations (41)-(45).
end for
Return the approximate value function $\bar{V}^x(\cdot|\Theta)$.
```

dated later (in a policy improvement phase) via a quasi-Newton optimization solution procedure; individual units localize to directions and introduce nonlinearities where needed. Moreover, this weight initialization policy forces symmetry breaking; initialization with exact zero weights leads to zero derivatives and perfect symmetry, which results in the weights never being updated.

We then begin a policy evaluation phase wherein, for each iteration $j = 1, 2, \ldots, J$, the following steps occur. We randomly select a post-decision state $S^x_{t-1,j}$ and compute the associated scaled basis function evaluation $\phi^s(S^x_{t-1,j})$. Utilizing Equations (34)-(37), we compute the value function approximation $\bar{V}^x(S^x_{t-1,j}|\Theta)$ and record the value. Next, we simulate the system evolving from post-decision state $S^x_{t-1,j}$ to a pre-decision state $S_{t,j}$, and we compute (and record) a sample realization of the value attained from the current policy by solving

$$\hat{v}_j = \max_{x_t \in \mathcal{X}_{S_{t,j}}} \left(C(S_{t,j}, x_t) + \gamma^{(\hat{\tau}(S_{t+1,j}) - \tau_t)} \bar{V}^x(S^x_{t,j}|\Theta) \right). \tag{40}$$

In a single policy evaluation phase, we collect a total of J sample realizations of the value attained by following the current policy.

We proceed into a policy improvement phase wherein, for each iteration $n = 1, 2, \ldots, N$, the following steps occur. We normalize the J value function sample realizations collected in the just completed policy evaluation phase, so the NN model may be properly fit to the collected approximate value function data.

We are now ready to update our NN weights and obtain an updated policy. We seek Θ-values that make the NN model fit the observed value function data well. We utilize a regularized, mean-squared error measure of fit (i.e., cost, error, or loss function), expressed as follows

$$L(\Theta) = \frac{1}{2J}\sum_{j=1}^{J}\left(\hat{v}_j - \bar{V}^x(S^x_{t-1,j}|\Theta)\right)^2 + \frac{\eta}{2J}\left(\sum_{f\in\mathcal{F}}\sum_{h\in\mathcal{H}}\left(\Theta^{(1)}_{f,h}\right)^2 + \sum_{h\in\mathcal{H}}\left(\Theta^{(2)}_{h}\right)^2\right). \quad (41)$$

The penalty term in the loss function prevents overfitting the data and reduces the generalization error. The regularization (i.e., weight decay) parameter $\eta \geq 0$ is a tunable parameter; larger η-values will tend to shrink the Θ-weights toward zero.

In our LSTD-API algorithm, we minimize a regularized, residual sum-of-squared error (i.e., least-squares) measure of fit, and we are able to directly attain an updated θ via the normal equation, which is an analytical solution to the least-squares regression problem. However, our NN-API algorithm employs a more complicated NN model to approximate the value function, and to attain an updated Θ we must minimize the loss function $L(\Theta)$ via computational methods. The new sample estimate of Θ,

$$\hat{\Theta} = \arg\min_{\Theta} \; L(\Theta), \quad (42)$$

is determined using MATLAB's *fminunc* optimization routine, which employs a Newton's method solution procedure with a trust-region method modification. The solution procedure requires gradient information, and we utilize the conventional NN *back-propagation* approach to compute the gradient. The applicable derivatives can be calculated via the chain rule for differentiation and are given by

$$\frac{\partial L_j}{\partial \Theta_h^{(2)}} = -\frac{1}{J}\left(\hat{v}_j - \bar{V}^x(S_{t-1,j}^x|\Theta)\right)\sigma'\left(Y^{(3)}(S_{t-1,j}^x)\right)Z_h^{(2)}(S_{t-1,j}^x) + \frac{\eta}{J}\Theta_h^{(2)}, \tag{43}$$

$$\frac{\partial L_j}{\partial \Theta_{f,h}^{(1)}} = -\frac{1}{J}\left(\hat{v}_j - \bar{V}^x(S_{t-1,j}^x|\Theta)\right)\sigma'\left(Y^{(3)}(S_{t-1,j}^x)\right)\Theta_h^{(2)}\sigma'\left(Y_h^{(2)}(S_{t-1,j}^x)\right)\phi_f^s(S_{t-1,j}^x) + \frac{\eta}{J}\Theta_{f,h}^{(1)}, \tag{44}$$

wherein $\sigma'(y)$ denotes the first-order derivative of $\sigma(y)$ with respect to y. Once $\hat{\Theta}$ is computed, we utilize the polynomial stepsize rule, Equation (32), to smooth in the input layer $\hat{\Theta}^{(1)}$-weights and hidden layer $\hat{\Theta}^{(2)}$-weights with the previous estimates $\Theta^{(1)}$ and $\Theta^{(2)}$, respectively. We then update Θ as follows

$$\Theta^{(i)} \leftarrow \alpha_n\hat{\Theta}^{(i)} + (1-\alpha_n)\Theta^{(i)}, \quad \text{for } i = 1, 2, \tag{45}$$

wherein the $\Theta^{(i)}$ on the right hand side is the prior estimate based on the previous $n-1$ policy improvement iterations and $\hat{\Theta}^{(i)}$ is our new estimate from the current policy improvement iteration for each $i = 1, 2$. As the number of policy improvement iterations n increases, we place less emphasis on recent sample estimates and more emphasis on the estimate based on the first $n-1$ iterations.

Once Θ is updated via Equation (45), we have completed one iteration of the policy improvement phase of the NN-API algorithm. If $n < N$ then the algorithm continues by starting another policy evaluation phase. The parameters N, J, $|\mathcal{H}|$, η, and β are tunable, where N is the number of iterations of the policy improvement phase, J is the number of iterations of the policy evaluation phase, $|\mathcal{H}|$ is the number of hidden layer units considered, η is the regularization term within the sample estimate computation, and β is the polynomial stepsize parameter.

3.6 Testing, Analysis, & Results

In this section, we develop and utilize a representative military medical evacuation (MEDEVAC) planning scenario to demonstrate the applicability of our Markov decision process (MDP) model to the military medical planning community and to examine the efficacy of our proposed approximate dynamic programming (ADP) solution techniques. We design and conduct computational experiments to examine how different algorithmic parameter settings and different features of the MEDEVAC dispatching problem impact the performance of our proposed ADP solution techniques. Moreover, we perform a series a sensitivity analyses on specific problem and algorithmic features to obtain insight regarding current MEDEVAC initiatives. We utilize a dual Intel Xeon E5-2650v2 workstation having 128 GB of RAM and MATLAB's Parallel Computing Toolbox to conduct the computational experiments and analyses herein.

We develop a notional, representative military MEDEVAC planning scenario in which the US military is performing high-intensity combat operations in support of the government of Azerbaijan. Our planning scenario considers two main operating bases (i.e., bases that host both a medical treatment facility (MTF) and a MEDEVAC staging area) and two forward operating bases (bases that only host a MEDEVAC staging area) in southern Azerbaijan. Moreover, we develop and consider a set of casualty cluster centers based on the projected locations of both friendly and enemy forces. Figure 8 depicts the two main operating bases, the two forward operating bases, and the 55 casualty cluster centers that we consider for our representative military MEDEVAC planning scenario.

The baseline problem instance considers a 9-line MEDEVAC request arrival rate of $\lambda = \frac{1}{30}$. That is, we assume an average 9-line MEDEVAC request arrival rate of one request per 30 minutes. Moreover, we assume that the MEDEVAC system em-

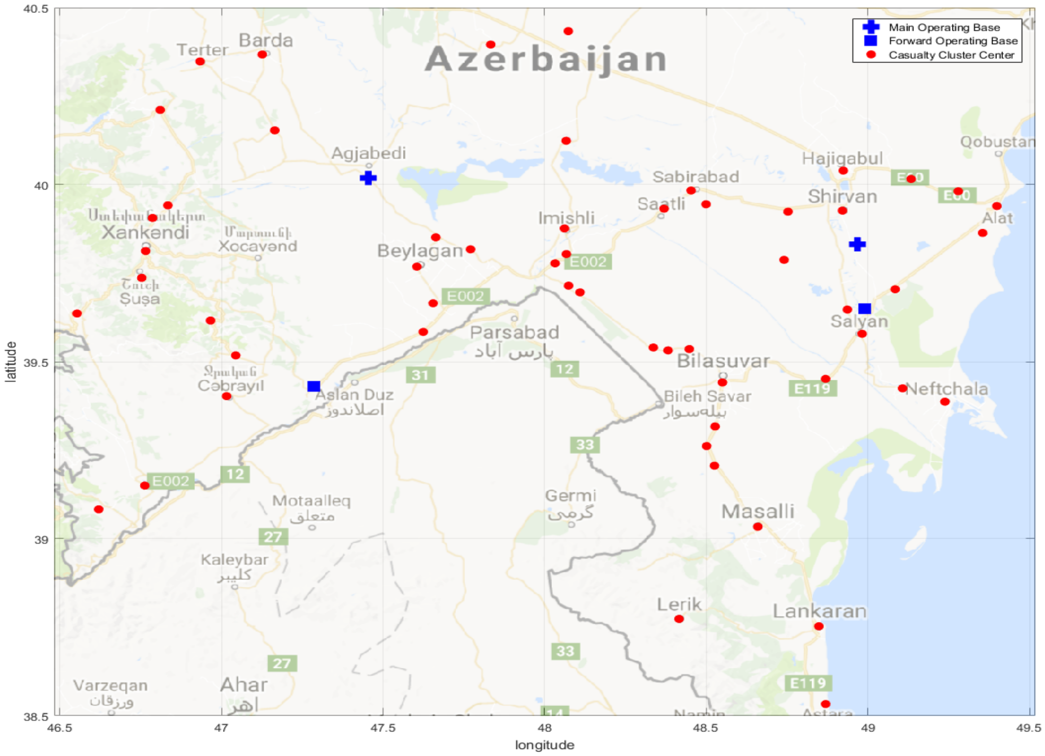


Figure 8. Representative Military MEDEVAC Planning Scenario Disposition

ploys an inter-zone policy regarding airspace access, allowing any MEDEVAC unit to service any request regardless of the casualty collection point (CCP) location. We set the baseline proportions of *urgent*, *priority*, and *routine* requests to 0.7, 0.2, and 0.1, respectively, (i.e., $\mathbb{P}(\textit{urgent, priority, routine}) = (0.7, 0.2, 0.1)$) to portray a highly kinetic conflict that results in more life-threatening engagements between hostile forces. We utilize data provided by Fulton et al. (2010) to determine the number of casualties contained in each 9-line MEDEVAC request. As such, the proportion of 9-line MEDEVAC requests that have one, two, three, and four casualties are 0.574, 0.36, 0.05, and 0.016, respectively. The proportion of 9-line MEDEVAC requests originating from a particular casualty cluster center depends on the casualty cluster center's location. For the baseline problem instance, we consider casualty events more likely to occur in the south (i.e., $L =$ south). We let w_1, w_2, and w_3 equal 1, 0.1, and 0.01, respectively, to prioritize *urgent* requests more than *priority* requests and *priority* requests more than *routine* requests. We utilize a relatively high discount factor of $\gamma = 0.999$ (i.e., exhibiting relatively low discounting of future epochs' operational values) in our computational experiments to incentivize the system to position itself to efficiently respond to future service requests. Lastly, we set $q^{max} = 5$, which allows the system to hold at most five 9-line MEDEVAC requests in the queue at any given time.

Our computational experiments measure ADP performance as the percent increase of total discounted reward obtained over an infinite horizon as compared to that obtained via the default dispatching policy in practice (i.e., closest-available policy). The closest-available policy tasks the closest-available MEDEVAC unit to respond to service requests regardless of other factors (e.g., precedence, demand distribution, and request arrival rate). Note that, since we cannot simulate an infinite trajectory, we instead simulate a 10,000-minute trajectory to produce a reasonable approximation. Moreover, because we are discounting, the γ^{τ_t}-term in our objective function becomes small enough that a longer simulation does not impact the measure of performance.

The performance and computational efficiency of each of our proposed ADP solution techniques vary based on different problem and algorithmic features. The problem features of interest include the average 9-line MEDEVAC request arrival rate, λ; the location in which requests are most likely to occur, L; and the proportions of *urgent*, *priority*, and *routine* requests, $\mathbb{P}(\textit{urgent, priority, routine})$. The algorithmic features of interest for our least-squares temporal differences approximate policy iteration (LSTD-API) algorithm include the number of policy improvement phase iterations, N; the number of policy evaluation phase iterations, J; the regularization term within the sample estimate computation, η; and the polynomial stepsize parameter, β. For our LSTD-API experimental design, we utilize a third-order polynomial of the basis functions to approximate the post-decision state value function. The algorithmic features of interest for our neural network approximate policy iteration (NN-API) algorithm include the number of policy improvement phase iterations, N; the number of policy evaluation phase iterations, J; the number of hidden layer units considered, $|\mathcal{H}|$; the regularization term within the sample estimate computation, η; and the polynomial stepsize parameter, β.

We generate 30 distinct problem instances utilizing a full factorial experimental design comprised of the λ, L, and $\mathbb{P}(\textit{urgent, priority, routine})$ problem factors (i.e., features). For each problem instance, we design and implement both a 3^4 full factorial computational experiment to examine the solution quality of policies determined by our LSTD-API algorithm for different levels of N, J, η, and β and a 3^5 full factorial computational experiment to examine the solution quality of policies determined by our NN-API algorithm. Table 4 shows the problem factor levels utilized in the experimental design to generate the 30 problem instances as well as the factor levels utilized in the computational experimental designs to determine solution quality for the two ADP algorithms. The selected parameter settings (i.e., factor levels) for the algorithmic features are based on our initial experiences and investigations with each ADP solution technique for this problem class. For each experimental design point (i.e., combination of factor levels), we perform 100 simulation runs to achieve our desired level of confidence. These experiments inform the selection of appropriate algorithm parameter values for subsequent sensitivity analyses. Moreover, these experiments provide a general insight regarding the performance of our ADP algorithms for high-intensity MEDEVAC dispatching problem instances.

Table 5 summarizes the results from our computational experiments by reporting the best parameter settings with regards to each ADP solution technique's performance for each problem instance. Starting from the left, Columns 1-3 indicate the problem factor levels associated with each problem instance. Columns 4-7 and

Table 4. Experimental Design Factor Levels

Category	Feature	Parameter Settings
Problem Instance	$\frac{1}{\lambda}$	{10, 20, 30, 40, 50}
	L	{south, east, west}
	$\mathbb{P}$(*urgent, priority, routine*)	{(0.7, 0.2, 0.1), (0.5, 0.4, 0.1)}
LSTD-API	N	{5, 10, 15}
	J	{1000, 5000, 10000}
	η	{10, 100, 1000}
	β	{0.5, 0.7, 0.9}
NN-API	N	{5, 10, 15}
	J	{1000, 5000, 10000}
	$\vert\mathcal{H}\vert$	{3, 5, 7}
	η	{0.001, 0.01, 0.03}
	β	{0.5, 0.7, 0.9}

Columns 8-12 tabulate the superlative parameter settings for the LSTD-API and NN-API algorithms, respectively, for each problem instance. Columns 13 and 14 respectively report the attendant solution qualities of the ADP policies determined by the LSTD-API and NN-API algorithms for each problem instance. The solution qualities of the ADP policies provide a measure of algorithm efficacy and are expressed in terms of the 95% confidence interval of percent improvement over the closest-available policy with respect to total discounted reward over an approximated infinite horizon. The last two columns report the computational efficiency of the LSTD-API and NN-API algorithms, respectively, as measured by the time required to generate a policy using the superlative parameter settings.

The results from Table 5 indicate that the ADP policies determined by the LSTD-API and NN-API algorithms significantly outperform the closest-available benchmark policies in 24 and 27 of the 30 problem instances examined, respectively. The ADP policies perform best in high-intensity conflicts wherein the average request arrival rate is relatively high (e.g., $\frac{1}{\lambda} \leq 30$ for this scenario). For example, the policies determined by our NN-API algorithm attain performances up to $346.56\%, 144.03\%$, and 28.6% for the $\frac{1}{\lambda} = 10$, $\frac{1}{\lambda} = 20$, and baseline $\frac{1}{\lambda} = 30$ problem instances, respectively.

Table 5. Experimental Design Results

Problem Instance Parameter Settings			LSTD-API Parameter Settings				NN-API Parameter Settings					Improvement Over Closest-Available Policy (%)		Computational Efficiency (Minutes)	
$\frac{1}{\lambda}$	L	$\mathbb{P}(\textit{urgent, priority, routine})$	N	J	η	β	N	J	$\lvert\mathcal{H}\rvert$	η	β	LSTD-API	NN-API	LSTD-API	NN-API
10	south	(0.7, 0.2, 0.1)	15	1000	1000	0.7	15	10000	3	0.03	0.9	138.90 ± 0.40	329.80 ± 0.40	7.79	100.59
10	south	(0.5, 0.4, 0.1)	5	1000	1000	0.5	10	10000	3	0.01	0.5	142.84 ± 0.37	346.56 ± 0.45	2.49	76.44
10	east	(0.7, 0.2, 0.1)	10	1000	1000	0.9	15	10000	3	0.001	0.7	123.35 ± 1.17	275.02 ± 0.58	4.26	112.21
10	east	(0.5, 0.4, 0.1)	10	1000	1000	0.7	10	10000	3	0.03	0.5	149.70 ± 0.49	283.67 ± 0.62	4.05	68.62
10	west	(0.7, 0.2, 0.1)	15	1000	1000	0.9	5	5000	3	0.001	0.7	169.75 ± 0.72	325.64 ± 0.86	6.39	26.38
10	west	(0.5, 0.4, 0.1)	5	1000	1000	0.5	10	10000	3	0.01	0.5	66.34 ± 0.27	167.28 ± 0.32	1.99	61.78
20	south	(0.7, 0.2, 0.1)	10	1000	1000	0.9	15	5000	3	0.03	0.7	81.70 ± 0.60	123.90 ± 0.80	5.86	56.51
20	south	(0.5, 0.4, 0.1)	10	1000	1000	0.9	15	10000	3	0.03	0.5	45.14 ± 0.16	85.82 ± 0.11	4.40	86.33
20	east	(0.7, 0.2, 0.1)	10	1000	1000	0.5	15	10000	3	0.01	0.7	66.34 ± 0.93	87.84 ± 0.84	4.64	89.55
20	east	(0.5, 0.4, 0.1)	10	1000	1000	0.5	15	10000	3	0.03	0.5	113.26 ± 0.18	144.03 ± 0.12	3.84	95.11
20	west	(0.7, 0.2, 0.1)	10	1000	1000	0.7	10	10000	3	0.03	0.5	86.63 ± 0.84	121.29 ± 1.06	4.45	64.55
20	west	(0.5, 0.4, 0.1)	10	1000	1000	0.7	15	10000	3	0.03	0.7	88.20 ± 0.28	117.45 ± 0.30	4.48	92.81
30	south	(0.7, 0.2, 0.1)	15	1000	1000	0.9	10	10000	5	0.001	0.5	11.50 ± 0.70	24.20 ± 0.70	8.74	110.59
30	south	(0.5, 0.4, 0.1)	5	1000	1000	0.9	15	10000	3	0.01	0.5	6.17 ± 0.31	16.17 ± 0.35	1.85	75.97
30	east	(0.7, 0.2, 0.1)	5	1000	1000	0.7	10	10000	3	0.001	0.7	6.75 ± 0.63	13.37 ± 0.70	1.92	69.25
30	east	(0.5, 0.4, 0.1)	10	1000	1000	0.5	15	10000	5	0.001	0.7	2.66 ± 0.05	9.97 ± 0.10	3.75	188.54
30	west	(0.7, 0.2, 0.1)	10	1000	1000	0.9	5	10000	3	0.03	0.9	21.45 ± 0.49	28.60 ± 0.54	3.79	31.16
30	west	(0.5, 0.4, 0.1)	10	1000	1000	0.7	5	10000	3	0.03	0.5	12.90 ± 0.05	20.85 ± 0.03	3.78	34.39
40	south	(0.7, 0.2, 0.1)	5	1000	1000	0.9	10	5000	3	0.03	0.5	0.52 ± 0.33	6.14 ± 0.60	2.07	26.68
40	south	(0.5, 0.4, 0.1)	10	10000	1000	0.7	15	10000	3	0.001	0.5	3.36 ± 0.09	12.57 ± 0.04	43.51	110.45
40	east	(0.7, 0.2, 0.1)	10	1000	1000	0.5	15	10000	3	0.01	0.7	1.04 ± 0.73	3.43 ± 0.71	4.66	81.78
40	east	(0.5, 0.4, 0.1)	15	1000	1000	0.7	15	10000	3	0.03	0.7	54.67 ± 0.53	58.84 ± 0.56	6.80	86.55
40	west	(0.7, 0.2, 0.1)	10	1000	1000	0.7	5	10000	3	0.03	0.7	2.24 ± 0.45	6.53 ± 0.48	4.70	30.50
40	west	(0.5, 0.4, 0.1)	10	1000	1000	0.9	15	10000	3	0.01	0.5	7.43 ± 0.31	10.72 ± 0.30	4.38	81.38
50	south	(0.7, 0.2, 0.1)	10	1000	1000	0.5	15	10000	7	0.001	0.7	-2.20 ± 0.56	1.36 ± 0.64	4.01	263.76
50	south	(0.5, 0.4, 0.1)	10	1000	1000	0.5	15	5000	7	0.01	0.7	-29.55 ± 0.23	-25.61 ± 0.22	3.95	168.95
50	east	(0.7, 0.2, 0.1)	10	1000	1000	0.7	10	10000	3	0.001	0.7	-0.37 ± 0.76	1.65 ± 0.75	4.45	71.76
50	east	(0.5, 0.4, 0.1)	5	1000	1000	0.7	15	10000	5	0.03	0.5	-13.98 ± 0.05	-11.94 ± 0.04	2.17	117.23
50	west	(0.7, 0.2, 0.1)	5	5000	10	0.9	10	5000	5	0.001	0.7	-0.73 ± 0.61	1.77 ± 0.63	11.21	66.04
50	west	(0.5, 0.4, 0.1)	15	5000	100	0.9	10	10000	3	0.03	0.9	-25.78 ± 0.54	-24.07 ± 0.54	34.42	60.19

In general, as the average request arrival rate decreases (i.e., as $\frac{1}{\lambda}$ increases), the performance of the ADP policies relative to the closest-available policy decreases. This result is intuitive and comports with the findings from previous related work (e.g., Rettke et al. (2016)). When the request arrival rate is relatively low, the dispatching authority accepts less risk by sending the closest-available MEDEVAC because the just-dispatched MEDEVAC unit is very likely to return to its own staging area prior to the arrival of another request. Accordingly, implementing an ADP policy in low-intensity conflicts wherein the average request arrival rate is relatively low (e.g., $\frac{1}{\lambda} > 30$ for this scenario) will likely yield little-to-no performance gain over the closest-available policy. Indeed, for low-intensity conflicts, the closest-available policy is likely optimal.

Although the relative performance improvements attained by the ADP policies diminish as $\frac{1}{\lambda}$ increases, we still expect both of our ADP algorithms to generate policies that perform at least as well as the closest-available policy, even with low MEDEVAC request arrival rates. However, upon examination of the last six problem instances in

Table 3 (i.e., when $\frac{1}{\lambda} = 50$), we observe that the closest-available policy significantly outperforms the policies generated from both of our ADP algorithms when $\mathbb{P}($*urgent, priority, routine*$) = (0.5, 0.4, 0.1)$. This result suggests that the algorithmic parameter settings should be tuned via a focused experimental design specifically to address problem instances having a low request arrival rate and a lesser number of *urgent* requests entering the system. Theoretically, after determining proper algorithmic parameter settings for this subclass of problem instances, the ADP algorithms should produce policies that perform at least as well as the closest-available policy.

To illustrate the benefits of an ADP policy in high-intensity conflict situations, we discuss the limitations of the closest-available policy. Consider the following three scenarios: *reserve, wait,* and *reject.* For the *reserve* scenario, we consider a system state wherein two MEDEVAC units are available for dispatch and a *routine* request from a high demand area has just been submitted to the system, which is shortly followed by an *urgent* request submission from the same area. Under the closest-available policy, the closest-available MEDEVAC unit is dispatched to the *routine* request. As such, the more distant MEDEVAC unit is dispatched to the subsequent *urgent* request. Clearly this sequence of events and actions is suboptimal since a higher reward is attained when the closest-available MEDEVAC unit is reserved for the *urgent* request and the more distant MEDEVAC unit is dispatched to the *routine* request. For the *wait* scenario, we consider a system state wherein a distant MEDEVAC unit is available for dispatch, one MEDEVAC unit is almost done servicing casualties from a previous request, and a request near the servicing MEDEVAC has just been submitted to the system. Under the closest-available policy, the available (but distant) MEDEVAC unit is dispatched to service the submitted request, which results in a lower reward attained by the system when compared to queueing the submitted request, waiting for the servicing MEDEVAC to become available, and assigning it to the queued submitted request. For the *reject* scenario, we consider a system state wherein a single MEDEVAC unit is available for dispatch in a high-demand area from where a high volume of *urgent* requests originate and a *routine* request near the available MEDEVAC unit is submitted to the system, which is shortly followed by an *urgent* request submission from the same area. Under the closest-available policy, the available MEDEVAC unit is dispatched to service the *routine* request leaving no available MEDEVAC units available to service the subsequent *urgent* request. This action results in a lower reward attained by the system when compared to rejecting the *routine* request from entering the system in anticipation of needing the MEDEVAC unit for the subsequent *urgent* request. Unlike the closest-available policy, the policies derived from our proposed ADP solution techniques utilize problem features to make decisions based on current *and* future events, resulting in better decisions and ultimately higher performance returns.

We observe that the superlative policies generated by the NN-API algorithm significantly outperform the superlative policies generated by the LSTD-API algorithm for each problem instance examined. This result can be explained due to the large number of free parameters (i.e., function mapping weights) embedded within the NN-API algorithm, which gives it the ability to more accurately fit highly complex data than other algorithms (e.g., LSTD-API). Unfortunately, the large number of free parameters also makes the NN-API algorithm a highly complex model that requires more computational running time than the LSTD-API algorithm. The results from the last two columns in Table 5 indicate that the LSTD-API algorithm is approximately 12 times faster than the NN-API algorithm, on average, when computing the top parameter settings for each problem instance. Even so, it is important to note that, once a satisfactory set of parameter settings is determined, MEDEVAC dispatching decisions can be computed nearly instantaneously.

We next compare the best performing NN-API algorithm policies to the closest-available policies for different MEDEVAC platforms and report the results in Tables 6-8. Except when otherwise noted, we utilize the baseline problem instance (i.e., $\frac{1}{\lambda} = 30$, $L =$ south, and $\mathbb{P}(\textit{urgent, priority, routine}) = (0.7, 0.2, 0.1)$) when performing the following analyses. The United States Army and United States Air Force respectively employ HH-60M Blackhawks and HH-60G Pave Hawks to conduct MEDEVAC missions. However, the Department of Defense (DoD) is aggressively pursuing the development of new rotary wing aircraft that are more capable to perform safe, reliable, and continuous operations worldwide (e.g., MEDEVAC missions) via the Future Vertical Lift (FVL) program (Drezner et al., 2017). The primary competitors for the FVL program are the Sikorsky-Boeing 1 (SB-1) Defiant and the Bell V-280 Valor. For comparison purposes, we consider the HH-60M closest-available policy data as our baseline case.

Starting from the left in Table 6, the first column indicates the MEDEVAC platform. The second column indicates the assumed aircraft speed in knots per hour (kph). Column three indicates the estimated cost per unit. Columns four and five respectively indicate the 95% confidence intervals of the performances for the NN-API and closest-available policies in terms of percent improvement over the baseline case. The last two columns respectively indicate the 95% confidence intervals of the MEDEVAC busy rates for the NN-API and closest-available policies.

Table 6. Dispatching Policy Performance Measures for the Selected MEDEVAC Platforms

			Improvement Over HH-60M Baseline (%)		MEDEVAC Busy Time (%)	
Platform	Speed (kph)	Unit Cost ($M)	NN-API	Closest-Available	NN-API	Closest-Available
HH-60M Blackhawk*	280	21.3	24.2 ± 0.7	-	97.4 ± 0.3	99.7 ± 0.1
HH-60G Pave Hawk	294.5	40.1	27.6 ± 0.7	5.7 ± 0.1	96.7 ± 0.3	99.7 ± 0.1
SB-1 Defiant	463	38	53.3 ± 0.8	47.4 ± 0.5	96.7 ± 0.3	99.7 ± 0.1
V-280 Valor	518.6	20	59.3 ± 0.8	54.8 ± 0.7	98.7 ± 0.2	99.7 ± 0.1

*Current platform employed by US Army.

The results from Table 6 indicate that, as airspeed increases, the overall performance for both NN-API and closest-available policies increases. Moreover, as airspeed increases the performance gap between NN-API and closest-available policies decreases. The last two columns reveal that the NN-API policies have significantly better (i.e., lower) MEDEVAC busy rates than the closest-available policies. However, we can observe that the differences in MEDEVAC busy rates have little-to-no practical significance since they are all nearly 100%. The MEDEVAC busy rates for both NN-API and closest-available policies are high in all cases due to the relatively high average MEDEVAC request arrival rate (i.e., $\lambda = \frac{1}{30}$). As mentioned previously, the SB-1 Defiant and V-280 Valor are currently competing against each other in the DoD's FVL program. While the official price per unit of the SB-1 Defiant is difficult to predict at this point in the research, development, and acquisition process, a fully burdened version of the SB-1 Defiant is currently estimated to be around \$38M-\$41M (Koucheravy, 2018). However, it is important to note that the estimate of the SB-1 Defiant is based on a number of assumptions and planning factors and most likely will be reduced after the final set of requirements are produced. Moreover, Bell has stated that it can offer the V-280 Valor for just \$20 million per unit, which is substantially less than the current estimate of the SB-1 Defiant (Adams, 2016). The unit costs reported in column three in Table 6 are provided to give decision makers a rough idea as to how each platform performs with its currently estimated cost. To further investigate how each MEDEVAC platform impacts performance, we next examine response times.

Starting from the left in Table 7, the first column indicates the MEDEVAC platform. The second and third columns respectively indicate the 95% confidence intervals of the *urgent* request response times for the NN-API and closest-available policies. Columns four and five respectively indicate the 95% confidence intervals for the *priority* request response times for the NN-API and closest-available policies. The last two columns respectively indicate the 95% confidence intervals of the *routine* request response times for the NN-API and closest-available policies.

Table 7. Dispatching Policy Response Times by Precedence Level for the Selected MEDEVAC Platforms

	Urgent		*Priority*		*Routine*	
	Response Time (Hours)		Response Time (Hours)		Response Time (Hours)	
Platform	NN-API	Closest-Available	NN-API	Closest-Available	NN-API	Closest-Available
HH-60M Blackhawk	0.90 ± 0.003	1.55 ± 0.1	0.79 ± 0.03	3.43 ± 0.4	0.83 ± 0.02	17.83 ± 2.7
HH-60G Pave Hawk	0.89 ± 0.003	1.44 ± 0.1	0.76 ± 0.01	3.05 ± 0.4	0.79 ± 0.05	14.37 ± 2.4
SB-1 Defiant	0.71 ± 0.002	0.86 ± 0.1	0.65 ± 0.01	1.13 ± 0.1	0.63 ± 0.02	3.45 ± 1.3
V-280 Valor	0.70 ± 0.002	0.79 ± 0.1	0.69 ± 0.01	0.96 ± 0.1	0.69 ± 0.01	2.60 ± 1.0

The results from Tables 6 and 7 show that, as airspeed increases, the response times decrease for all precedence categories for the closest-available policies. For the NN-API policies, this trend also holds for the *urgent* request response times but not for the *priority* or *routine* request response times. The gaps in response times between NN-API and closest-available policies decrease as airspeed increases. Moreover, we observe that the response times for the NN-API policies are significantly lower than the response times for the closest-available policies. This relationship can be explained in part by the NN-API algorithm having an admission control policy that allows the dispatching authority to reject incoming requests from entering the system, whereas the closest-available policy must service incoming requests if at least one MEDEVAC unit is available. We display the rejection rates of the NN-API and closest-available policies for each MEDEVAC platform in Table 8.

Within Table 8, the first column indicates the MEDEVAC platform. The second and third columns respectively indicate the 95% confidence intervals of the *urgent* request rejection rates for the NN-API and closest-available policies. Columns four and five respectively indicate the 95% confidence intervals for the *priority* request rejection rates for the NN-API and closest-available policies. Columns six and seven respectively indicate the 95% confidence intervals for the *routine* request rejection rates for the NN-API and closest-available policies. The last two columns respectively indicate the 95% total rejection rate confidence intervals for the NN-API and closest-available policies.

Table 8. Dispatching Policy Rejection Rates by Precedence Level for the Selected MEDEVAC Platforms

	Urgent Rejection Rate (%)		*Priority* Rejection Rate (%)		*Routine* Rejection Rate (%)		Total Rejection Rate (%)	
Platform	NN-API	Closest-Available	NN-API	Closest-Available	NN-API	Closest-Available	NN-API	Closest-Available
HH-60M Blackhawk	7.1 ± 0.3	33.9 ± 3.8	76.7 ± 1.0	35.0 ± 4.0	89.0 ± 2.9	39.7 ± 4.7	29.0 ± 0.5	34.7 ± 3.8
HH-60G Pave Hawk	5.3 ± 0.3	30.9 ± 3.8	76.5 ± 1.1	31.8 ± 4.1	97.3 ± 0.5	34.5 ± 4.5	28.8 ± 0.5	31.4 ± 3.9
SB-1 Defiant	3.5 ± 0.2	6.0 ± 2.5	62.7 ± 1.1	6.1 ± 2.6	96.3 ± 0.6	6.6 ± 2.8	24.7 ± 0.4	6.1 ± 2.6
V-280 Valor	1.3 ± 0.2	4.3 ± 2.1	20.8 ± 1.0	4.2 ± 2.1	93.4 ± 1.0	4.8 ± 2.4	10.7 ± 0.3	4.3 ± 2.1

The results from Tables 6 and 8 show that, as airspeed increases, the rejection rates decrease for all precedence categories for the closest-available policies. For NN-API policies, this relationship also holds for the *urgent* and *priority* request rejection rates but not for the *routine* request rejection rates. The gaps in rejection rates between NN-API and closest-available policies decrease as airspeed increases. Moreover, we observe that the *urgent* rejection rates for the NN-API policies are significantly lower than the *urgent* rejection rates for the closest-available policies. The opposite is true for *priority* and *routine* requests. These results reveal that the NN-API policies reject more *priority* and *routine* requests than the closest-available policies to allow for a higher number of *urgent* requests to be serviced. This result is intuitive since the reward attained by the MEDEVAC system for servicing *urgent* requests is substantially higher than servicing *priority* or *routine* requests, and there is a much higher proportion of *urgent* requests (i.e., 0.7) arriving to the system compared to *priority* requests (i.e., 0.2) and *routine* requests (i.e., 0.1). The last two columns in Table 8 show that the overall rejection rate for the HH-60M is significantly lower under the NN-API policy when compared against the closest-available policy. Moreover, there is not a statistically significant difference in the overall rejection rates between the NN-API and closest-available policies for the HH-60G Pave Hawk. However, the overall rejection rates for the FVL platforms (i.e., SB-1 Defiant and V-280 Valor) are significantly higher for NN-API policies when compared against closest-available policies. While these higher rejection rates may not align with expectations, admitting more *priority* and *routine* requests to the system will ultimately lead to longer *urgent* request response times since more MEDEVAC units will be busy servicing lower precedence requests. Hence, some rejection is desirable as it reserves dedicated MEDEVAC units for more life-threatening requests. It is important to recall that rejected requests are not simply discarded; they are redirected and serviced by non-medical command authorities aboard non-medical, combat support transportation assets (e.g., CASEVAC).

The results from Tables 6-8 demonstrate that NN-API policies have diminishing returns when compared to closest-available policies as airspeed increases. Even so, military medical planners should still consider the value of ADP policies and the high-quality dispatching solutions they have to offer.

3.7 Conclusions

This chapter examines the medical evacuation (MEDEVAC) dispatching problem. The intent of this research is to determine dispatching policies that improve the performance of a deployed Army MEDEVAC system and ultimately decrease the case fatality rate (i.e., percentage of fatalities among all combat casualties). We developed a discounted, infinite-horizon Markov decision process (MDP) model of the MEDEVAC dispatching problem, which enables examination of a variety of problem features relating to different military medical planning scenarios. Whereas the MDP model provides an appropriate framework for solving the MEDEVAC dispatching problem, the large size of our motivating problem instances required the design, development, and implementation of approximate dynamic programming (ADP) techniques to solve the problem in a tractable amount of time. As such, we developed and tested two ADP solution techniques that both utilize an approximate policy iteration (API) algorithmic framework. The first ADP solution technique considers least-squares temporal differences (LSTD) learning for policy evaluation, whereas the second ADP solution technique considers neural network (NN) learning for policy evaluation. Utilizing selected MEDEVAC dispatching problem features, we defined a set of basis functions to approximate the value function around the post-decision state for both of our proposed ADP solution techniques. We constructed a notional, representative planning scenario based on high-intensity combat operations in southern Azerbaijan to demonstrate the applicability of our MDP model and to examine the efficacy of our proposed ADP solution techniques.

The results from our computational experiments indicate that the respective policies generated by the NN-API and LSTD-API algorithms significantly outperform the closest-available benchmark policies in 27 (90%) and 24 (80%) of the 30 problem instances examined. Furthermore, the policies derived from the best NN-API parameter settings have performances that are significantly higher than the performances from policies derived from the best LSTD-API parameter settings. The advantage of NN-API results from its nonlinear architecture of nonlinear activation functions that can be effectively trained in an iterative manner, but this advantage comes at the cost of substantially increased computational time. Even so, the computational time required for the NN-API algorithm is not large enough to set it aside from consideration as a viable solution technique. We also determined that NN-API policies have diminishing returns when compared to closest-available policies as airspeed increases.

This research is of interest to both military and civilian medical planners. Medical planners can apply our MDP model and ADP solution techniques to compare different dispatching policies for a variety of planning scenarios that have fixed medical treatment facility and MEDEVAC staging locations (i.e., hospital and ambulance locations for the civilian sector). Moreover, medical planners can evaluate different location schemes for medical assets to determine the best allocation of resources if they are not already fixed or the emplacement of new assets is being considered. Although implementing an ADP policy into active dispatching operations may require the use of a decision support system to alleviate the burden to a dispatcher, the point of this research is to show that the closest-available dispatching policy is not always superlative and that there exist operations research techniques that may improve MEDEVAC system performance.

Of the MEDEVAC platforms examined, the FVL V-280 Valor performance is superlative, an expected result due to its airspeed. Both the military medical planning community and the military acquisition community (i.e., those personnel responsible for implementing new technology into the military) can leverage the results from this research to examine how the currently employed MEDEVAC platforms (i.e., HH-60M Blackhawk and HH-60G Pave hawk) and the currently competing FVL platforms (i.e., SB-1 Defiant and V-280 Valor) compare in a high-intensity, representative planning scenario. Such comparisons inform the implementation and modification of current and future MEDEVAC tactics, techniques, and procedures, as well as the design and purchase of future aerial MEDEVAC assets.

IV. Approximate Dynamic Programming for the Military Aeromedical Evacuation Dispatching, Preemption-Rerouting, and Redeployment Problem

4.1 Introduction

A fundamental objective of the military's health service support system is to ensure effective and efficient delivery of healthcare across the continuum of military operations. It is important for senior military leaders and medical planners to continuously seek enhanced healthcare methods to eliminate current shortfalls and satisfy future requirements. In particular, the enhancement of the treatment and evacuation of combat casualties to medical treatment facilities (MTFs) is a key objective of the United States (US) Army (Department of the Army, 2016). It is well known that effective evacuation of combat casualties relies heavily on the reduction of time between injury and treatment (Kotwal et al., 2016; Howard et al., 2018). To improve combat casualty survivability, the military's health service support system utilizes aeromedical evacuation (MEDEVAC) as the primary link between roles of medicare care on the battlefield since it is comprised of dedicated, standardized air evacuation platforms designed, equipped, and staffed to rapidly respond to requests for medical support during combat operations and provide necessary en route care to casualties being delivered to an appropriate MTF (Department of the Army, 2016). This chapter leverages operations research and machine learning techniques to determine high-quality MEDEVAC policies that increase the effectiveness and efficiency of deployed US Army MEDEVAC systems.

As a critical component of deployed military forces, MEDEVAC systems serve as a force multiplier, providing the necessary connectivity between roles of medical care. Effective and efficient MEDEVAC systems utilize limited assets, minimize long term injuries, and maximize casualty survivability rates by ensuring the highest levels of medical care are available when needed. Moreover, MEDEVAC systems that provide prompt evacuation and timely critical care of casualties raise morale of deployed military personnel, ultimately improving the effectiveness of operational combat units by demonstrating medical assistance is quickly available upon request (Department of the Army, 2016). Several important decisions must be considered when designing a MEDEVAC system. These decisions include: where to locate MEDEVAC staging facilities, how many MEDEVAC units (i.e., aeromedical helicopters) to allocate to each staging facility, which MEDEVAC unit to assign (i.e., dispatch or preempt-and-reroute) to respond to a request for service, and where a MEDEVAC unit redeploys after transferring casualty care to an MTF's staff.

Given a set of fixed staging facilities and MTFs as well as a predefined allocation of MEDEVAC units to each staging facility, the identification of a policy that determines which MEDEVAC units to assign to respond to requests for service and where MEDEVAC units redeploy after finishing a service request (i.e., upon successful transfer of casualty care to an MTF's staff), herein referred to as the MEDEVAC dispatching, preemption-rerouting, and redeployment (DPR) problem, is vital to the success of a deployed MEDEVAC system and is the primary focus of this chapter. The US military typically utilizes a closest-available dispatching policy, which assigns the closest-available MEDEVAC unit to respond to a request for service, regardless of the request's characteristics (e.g., location and severity) or the MEDEVAC system state (e.g., location and availability of MEDEVAC units), and it does not consider MEDEVAC unit redeployment (i.e., MEDEVAC units return to their home staging facilities after completing a service request). Besides the lack of redeployment, one of the primary drawbacks of this policy is that it only allows idle MEDEVAC units (i.e., MEDEVAC units awaiting mission tasking at their staging facility) to be dispatched in response to incoming requests for service. Moreover, although the closest-available dispatching policy may perform well in low intensity conflicts wherein the average arrival rate of MEDEVAC requests is low (e.g., one request every two hours), many researchers (e.g., Keneally et al. (2016), Rettke et al. (2016), and Jenkins et al. (2018)) have shown that, as the average arrival rate of requests increases (e.g., in high intensity conflicts), policies that disregard important information concerning incoming requests and the state of the system, such as the closest-available dispatching policy, perform poorly.

The objective of this research is to generate high-quality policies that dispatch, preempt-and-reroute, and redeploy MEDEVAC units in a way that improves upon the currently practiced closest-available dispatching policy for large-scale, highly kinetic, and intense military conflicts. A discounted, infinite-horizon Markov decision process (MDP) model of the MEDEVAC DPR problem is formulated and solved via an approximate dynamic programming (ADP) strategy that utilizes a support vector regression (SVR) value function approximation scheme within an approximate policy iteration (API) framework. The objective is to maximize expected total discounted reward attained by the system. The locations of MEDEVAC staging facilities and MTFs are known, as is the allocation of MEDEVAC units to each staging facility. Moreover, it is assumed that each MEDEVAC unit has the capability to satisfy the mission requirements of any request for service.

Indeed, the literature on civilian and military emergency medical service (EMS) response systems is quite rich and, as such, only highly related papers that incorporate ADP strategies are discussed; for an extensive review of related literature see Swersey (1994), Brotcorne et al. (2003), Ingolfsson (2013), and Aringhieri et al. (2017). ADP research involving civilian EMS systems primarily focuses on ambulance redeployment, relocation, and dispatching. Maxwell et al. (2010) solve the ambulance redeployment problem (i.e., determining to where ambulances are moved after completing service at a hospital) via a least-squares policy evaluation (LSPE) value function approximation scheme within an API algorithmic framework. Schmid (2012) considers both ambulance dispatching and redeployment. The author utilizes a spatial and temporal aggregation value function approximation scheme within an approximate value iteration algorithmic framework to develop ambulance dispatch-redeployment policies. Lastly, Nasrollahzadeh et al. (2018) consider an ADP solution methodology for ambulance dispatching, relocation, and redeployment decisions. Similar to Maxwell et al. (2010), the authors employ an ADP strategy that utilizes an LSPE value function approximation scheme within an API algorithmic framework. The effectiveness of these ADP strategies is demonstrated via applications to EMS response scenarios, of which, all ADP strategies show improvements over benchmark policies.

While similarities between civilian and military EMS response systems do exist, there also exists substantive differences that require consideration when examining the performance of a military-specific EMS response system. For example, the travel, load, and unload times are much greater during military MEDEVAC operations due to the complex nature of each mission (e.g., threat level concerns, number and severity of casualties, and remote terrain areas) (Jenkins et al., 2018). Moreover, most civilian EMS response systems enforce a response time threshold (RTT) of nine minutes or less, so service preemption is typically not modeled because, in doing so, computational complexity significantly increases while only providing a marginal benefit to system performance (Maxwell et al., 2010). Such moderate improvement is not the case for military MEDEVAC missions. Of 4,542 casualties transported via aeromedical helicopters from the battlefield to an MTF in Afghanistan between 2001 and 2014, the average response time was 73.1 minutes (Kotwal et al., 2016). The RTTs of military casualties varies by precedence level (i.e., severity level) and range from 1 to 24 hours (Department of the Army, 2016). Thus, modeling preemption rules when examining military EMS response systems is necessary. It is important to understand when preemption is needed and how much it can improve system performance. More examples of differences between civilian and military EMS response systems are presented by Keneally et al. (2016), Rettke et al. (2016), and Jenkins et al. (2019). Due to these differences, the investigation of military-specific EMS response systems (i.e, military MEDEVAC systems) is warranted.

ADP research on military MEDEVAC systems exists, but it focuses on the MEDEVAC dispatching problem (i.e., determining which MEDEVAC unit to dispatch in response to a request for service). Of the existing research, Rettke et al. (2016) are responsible for the first paper that utilizes ADP to improve upon the US military's closest-available dispatching policy. The authors design, develop, and test an ADP strategy that utilizes a least-squares temporal differences (LSTD) value function approximation scheme within an API algorithmic framework. Robbins et al. (2018) and Jenkins et al. (2019) follow Rettke et al. (2016) and also utilize ADP strategies to improve upon the US military's closest available dispatching policy. More specifically, Robbins et al. (2018) utilize a hierarchical aggregation value function approximation scheme within an API algorithmic framework, whereas Jenkins et al. (2019) utilize LSTD and neural network value function approximation schemes within API algorithmic frameworks. Within all three works, the ADP policies perform significantly better than the currently practiced closest-available dispatching policy.

This research makes the following contributions. First, it defines the military MEDEVAC DPR problem, extending prior work on the MEDEVAC dispatching problem by simultaneously determining dispatching decisions in conjunction with preemption-rerouting and redeployment decisions. Preemption has yet to be addressed in either the military or civilian EMS literature. A realistic MDP model of the MEDEVAC DPR problem is formulated that includes: queueing; prioritized MEDEVAC requests; fuel constraints; dispatching decisions; preemption-rerouting decisions; and redeployment decisions. Most notably, the incorporation of fuel constraints, preemption-rerouting decisions, and redeployment decisions are significant as they have not been examined in related efforts regarding military MEDEVAC systems (e.g., Keneally et al. (2016), Rettke et al. (2016), Jenkins et al. (2018), Robbins et al. (2018), and Jenkins et al. (2019)). Second, this research provides the first proposal and demonstration of the relative efficacy of an ADP strategy that utilizes an SVR value function approximation scheme within an API algorithmic framework for military MEDEVAC problems. Moreover, this solution methodology has not been applied within the literature regarding civilian EMS response systems. Third, it examines the applicability of the MDP model and highlights the efficacy of the proposed solution methodology for a notional, representative planning scenario based on high-intensity combat operations in Azerbaijan. Computational experimentation is performed to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated DPR policies. Fourth, the modeling and solution approach presented in this work is transferable to civilian EMS response systems, particularly those located in areas that rely heavily on air evacuation platforms as the primary means for service (e.g., vast rural regions) or for aeromedical natural disaster response.

The remainder of this chapter is organized as follows. Section 4.2 sets forth the MDP model formulation of the MEDEVAC DPR problem. Section 4.3 presents the proposed ADP solution methodology. Section 4.4 demonstrates the applicability of the MDP model and examines the efficacy of the proposed ADP solution methodology via designed computational experiments. Section 4.5 concludes the chapter.

4.2 Problem Formulation

This section presents a discounted, infinite-horizon Markov decision process (MDP) model of the military aeromedical evacuation (MEDEVAC) dispatching, preemption-rerouting, and redeployment (DPR) problem.

4.2.1 State Space

The state of the system is a minimally dimensioned function that is both necessary and sufficient to compute the decision function, transition function, and the contribution function (Powell, 2011). Let $S = (\tau, e, M, R) \in \mathcal{S}$ denote the state of the MEDEVAC system, wherein τ is the current system time, e is the current system event type, M is the MEDEVAC unit status tuple, R is the request status tuple, and $\mathcal{S}$ is the set of all possible states. Throughout this chapter, $\tau(S)$ and $e(S)$ respectively denote the system time and event type when the MEDEVAC system is in state S.

The MEDEVAC unit status tuple is defined as

$$M = (M_m)_{m \in \mathcal{M}} \equiv (M_1, M_2, \ldots, M_{|\mathcal{M}|}),$$

wherein $\mathcal{M} = \{1, 2, \ldots, |\mathcal{M}|\}$ denotes the set of MEDEVAC units in the system and M_m contains information pertaining to MEDEVAC unit $m \in \mathcal{M}$. The state of MEDEVAC unit m is represented by $M_m = (\zeta_m, l_m, \mu_m, d_m, p_m, c_m)$, wherein ζ_m is the status of the MEDEVAC unit, l_m is the location of the MEDEVAC unit, μ_m is the expected system time in which the care of the MEDEVAC unit's onboard casualties is transferred to an MTF's staff, d_m is the remaining distance the MEDEVAC unit can travel before violating its refueling threshold, p_m is the precedence level of the request being serviced by the MEDEVAC unit, and c_m is the reward (i.e., contribution) attained by the system for the MEDEVAC unit's most recent incomplete request assignment. For the purposes of this work, it is sufficient to consider nine possibilities for the status of a MEDEVAC unit, i.e., $\zeta_m \in \{1, 2, \ldots, 9\}$, wherein $\zeta_m = 1$ shows that MEDEVAC unit m is idle at a staging facility, $\zeta_m = 2$ shows that MEDEVAC unit m is en route to a casualty collection point (CCP), $\zeta_m = 3$ shows that MEDEVAC unit m is servicing a request at the request's CCP (i.e., loading casualties onto MEDEVAC unit), $\zeta_m = 4$ shows that MEDEVAC unit m has completed service at the CCP (i.e., loaded casualties onto MEDEVAC unit and prepared to travel to nearest MTF), $\zeta_m = 5$ shows that MEDEVAC unit m is en route to the nearest MTF, $\zeta_m = 6$ shows that MEDEVAC unit m is servicing a request at an MTF (i.e., transferring casualty care to an MTF's staff), $\zeta_m = 7$ shows that MEDEVAC unit m has completed the service of a request (i.e., transferred casualty care to an MTF's staff), $\zeta_m = 8$ shows that MEDEVAC unit m is en route to a staging facility, and $\zeta_m = 9$ shows that MEDEVAC unit m is refueling and re-equipping at a staging facility. Note that, if MEDEVAC unit m is not currently servicing a request (i.e., $\zeta_m \in \{1, 7, 8, 9\}$), then $\mu_m = p_m = 0$. Moreover, if MEDEVAC unit m is currently servicing a request, then $p_m \in \{1, 2\}$, wherein $p_m = 1$ shows that MEDEVAC unit m is currently servicing a lower-precedence (i.e., *priority*) request and $p_m = 2$ shows that MEDEVAC unit m is currently servicing a higher-precedence (i.e., *urgent*) request. Lastly, $c_m > 0$ only when $\zeta_m = 2$ and MEDEVAC unit m has just been preempted-and-rerouted to service a new request, otherwise $c_m = 0$.

The request status tuple is defined as

$$R = (R_r)_{r \in \mathcal{R}} \equiv (R_1, R_2, \ldots, R_{|\mathcal{R}|}),$$

wherein $\mathcal{R} = \{1, 2, \ldots, |\mathcal{R}|\}$ denotes the set of possible requests and R_r contains information pertaining to request $r \in \mathcal{R}$. The state of request r is represented by $R_r = (\delta_r, o_r, t_r, \rho_r, n_r)$, wherein δ_r is the status of the request, o_r is the location of the request's CCP, t_r is the time at which the request entered the system, ρ_r is the precedence of the request (i.e., priority or urgent), and n_r is the number of casualties within the request. The status of request r is either "queued" (i.e., $\delta_r = 0$) or "assigned to MEDEVAC unit m" (i.e., $\delta_r = m : m \in \mathcal{M}$). A request is removed from the system once its servicing MEDEVAC unit transfers the care of the request's casualties to an MTF's staff. If request r does not exist, then $R_r = (0, 0, 0, 0, 0)$. Let r^{max} denote the maximum number of requests that can be tracked in the system such that $|\mathcal{R}| = r^{max}$. This assumption is not restrictive because one may assume a relatively large r^{max}.

The dynamics of the system are based on an event driven process. This research considers seven distinct event types, i.e., $e(S) \in \{1, 2, \ldots, 7\}$, wherein $e(S) = 1$ represents a request entering the system, $e(S) = 2$ represents a MEDEVAC unit arriving at a CCP, $e(S) = 3$ represents a MEDEDAC unit completing service at a CCP, $e(S) = 4$ represents a MEDEDAC unit arriving at an MTF, $e(S) = 5$ represents a MEDEDAC unit completing service at an MTF, $e(S) = 6$ represents a MEDEDAC unit arriving at a staging facility, and $e(S) = 7$ represents a MEDEVAC unit becoming idle at a staging facility.

4.2.2 Action Space

To formulate the action space, consider the definitions for the following sets and decision variables:

Sets

- Let $\mathcal{H} = \{1, 2 \ldots, |\mathcal{H}|\}$ denote the set of staging facilities
- Let $\mathcal{Q}(S) = \{r : r \in \mathcal{R}, \delta_r = 0\}$ denote the set of requests waiting in the queue for a MEDEVAC unit assignment when the system is in state S

- Let $\mathcal{A}_1(S) = \{m : m \in \mathcal{M}, \zeta_m \in \{1, 7\}\}$ denote the set of MEDEVAC units available for dispatching when the system is in state S
- Let $\mathcal{A}_2(S) = \{m : m \in \mathcal{M}, \zeta_m \in \{2, 8\}, p_m \neq 2\}$ denote the set of MEDEVAC units available for preemption-rerouting when the system is in state S
- Let $\mathcal{A}_3(S) = \{m : m \in \mathcal{M}, \zeta_m = 7\}$ denote the set of MEDEVAC units available for redeployment when the system is in state S

Decision Variables

- Let $X_{mr} = 1$ if MEDEVAC unit m is dispatched to service request r, 0 otherwise
- Let $Y_{mr} = 1$ if MEDEVAC unit m is preempted and rerouted to service request r, 0 otherwise
- Let $Z_{mh} = 1$ if MEDEVAC unit m is redeployed to staging facility h, 0 otherwise

Given this framework, the action space is described in four cases.

Case 1. If $\mathcal{Q}(S) \neq \emptyset$ and $e(S) \in \{1, 2, 3, 4, 6, 7\}$, the decision maker has two types of decisions: (1) which MEDEVAC units (if any) should immediately dispatch to service requests waiting in the queue for a MEDEVAC unit assignment and (2) which MEDEVAC units (if any) should immediately preempt-and-reroute to service requests waiting in the queue for a MEDEVAC unit assignment. Note that the decision maker is not required to immediately dispatch or preempt-and-reroute MEDEVAC units to service new or preexisting requests. Moreover, in this case, the decision maker is bounded by five constraints. The first constraint,

$$\sum_{m \in \mathcal{A}_1(S)} X_{mr} + \sum_{m \in \mathcal{A}_2(S)} Y_{mr} \leq 1, \forall\, r \in \mathcal{Q}(S), \tag{46}$$

ensures each request has no more than one MEDEVAC unit dispatched or rerouted to it when the system is in state S. The second and third constraints,

$$\sum_{r \in \mathcal{Q}(S)} X_{mr} \leq 1, \forall\, m \in \mathcal{A}_1(S) \tag{47}$$

and

$$\sum_{r \in \mathcal{Q}(S)} Y_{mr} \leq 1, \forall\, m \in \mathcal{A}_2(S), \tag{48}$$

ensure each MEDEVAC unit is dispatched or rerouted to no more than one request when the system is in state S. Let $\bar{d}_{mr}$ denote the total distance required for MEDEVAC m to transport request r's casualties to the nearest MTF. The last two constraints,

$$X_{mr}\bar{d}_{mr} \leq d_m, \forall\, m \in \mathcal{A}_1(S), \forall\, r \in \mathcal{Q}(S) \tag{49}$$

and

$$Y_{mr}\bar{d}_{mr} \leq d_m, \forall\, m \in \mathcal{A}_2(S), \forall\, r \in \mathcal{Q}(S), \tag{50}$$

ensure that a MEDEVAC unit can only be dispatched or rerouted to a request if the MEDEVAC unit has sufficient fuel when the system is in state S. Let

$$X = (X_{mr})_{m \in \mathcal{A}_1(S), r \in \mathcal{Q}(S)} \equiv (X_{1,1}, X_{1,2}, \ldots, X_{1,|\mathcal{Q}(S)|}, X_{2,1}, \ldots, X_{|\mathcal{A}_1(S)|,|\mathcal{Q}(S)|})$$

denote the dispatch decision tuple and

$$Y = (Y_{mr})_{m \in \mathcal{A}_2(S), r \in \mathcal{Q}(S)} \equiv (Y_{1,1}, Y_{1,2}, \ldots, Y_{1,|\mathcal{Q}(S)|}, Y_{2,1}, \ldots, Y_{|\mathcal{A}_2(S)|,|\mathcal{Q}(S)|})$$

denote the preempt-and-reroute decision tuple. The action space is given by

$$\mathcal{X}_1(S) = \{(X, Y) : \text{Constraints (46)-(50)}\}.$$

Case 2. If $\mathcal{Q}(S) = \emptyset$ and $e(S) = 5$ (i.e., MEDEVAC unit m has just finished transferring casualty care to an MTF's staff), the decision maker must decide where to redeploy MEDEVAC unit m. Note that in this case $A_3(S) = \{m\}$. Moreover, let $\bar{d}_{mh}$ denote the total distance required for MEDEVAC m to travel to staging facility h and $\mathbb{I}_{\{\bar{d}_{mh} \leq d_m\}}$ denote an indicator function that takes the value of one if the distance to staging facility h is less than or equal to the remaining distance MEDEVAC unit m can travel before violating its refueling threshold. The constraint,

$$\sum_{h \in \mathcal{H}} \mathbb{I}_{\{\bar{d}_{mh} \leq d_m\}} Z_{mh} = 1, \forall\ m \in \mathcal{A}_3(S), \tag{51}$$

requires each redeployable MEDEVAC unit to redeploy to a single staging facility. Each MTF is collocated with a staging facility, and it is assumed that a MEDEVAC unit will always have enough fuel to travel from an MTF to the MTF's collocated staging facility. Let

$$Z = (Z_{mh})_{m \in \mathcal{A}_3(S), h \in \mathcal{H}} \equiv (Z_{1,1}, Z_{1,2}, \ldots, Z_{1,|\mathcal{H}|}, Z_{2,1}, \ldots, Z_{|\mathcal{A}_3(S)|,|\mathcal{H}|})$$

denote the redeployment decision tuple. The action space is given by

$$\mathcal{X}_2(S) = \{Z : \text{Constraint (51)}\}.$$

Case 3. If $\mathcal{Q}(S) \neq \emptyset$ and $e(S) = 5$ (i.e., MEDEVAC unit m has just finished transferring casualty care to an MTF's staff), the decision maker has three types of decisions: (1) which MEDEVAC units (if any) should immediately dispatch to service requests waiting in the queue for a MEDEVAC unit assignment, (2) which MEDEVAC units (if any) should immediately preempt-and-reroute to service requests waiting in the queue for a MEDEVAC unit assignment, and (3) where to redeploy MEDEVAC unit m if it is not dispatched to a request waiting in the queue. Note that in this case $A_3(S) = \{m\}$. The constraint,

$$\sum_{r \in \mathcal{Q}(S)} X_{mr} + \sum_{h \in \mathcal{H}} \mathbb{I}_{\{\bar{d}_{mh} \leq d_m\}} Z_{mh} = 1, \forall\ m \in \mathcal{A}_3(S), \tag{52}$$

requires redeployable MEDEVAC units to redeploy to a staging facility if they are not dispatched to a request waiting in the queue. The action space is given by

$$\mathcal{X}_3(S) = \{(X, Y, Z) : \text{Constraints (46)-(50), (52)}\}.$$

Case 4. If an event type occurs and does not satisfy the criteria of any of the cases listed above, then the action space is $\mathcal{X}_4(S) = \emptyset$.

4.2.3 Transitions

Let S_k denote the state of the system when the kth event occurs. The evolution of the MEDEVAC system from state S_k to state S_{k+1} is characterized by action x_k, a random element $\omega(S_k, x_k)$ that captures the information that arrives to the system in state S_{k+1}, and a state transition function S^M. As such, the following recursion models the dynamics of the MEDEVAC system: $S_{k+1} = S^M(S_k, x_k, \omega(S_k, x_k))$.

4.2.4 Contributions

The MEDEVAC system earns contributions (i.e., rewards) when the decision maker assigns (i.e., dispatches or preempts-and-reroutes) MEDEVAC units to service requests. The total contribution earned depends on several factors (e.g., request precedence levels, expected service times, and request entry times). Let $C(S_k, x_k)$ denote the immediate expected contribution attained by the system when the system is in state S_k and action x_k is taken; it is computed as follows

$$C(S_k, x_k) = \sum_{m \in \mathcal{A}_1(S_k)} \sum_{r \in \mathcal{R}} w_{\rho_r} \Psi_\rho(\mu_m - t_r) X_{mr} +$$
$$\sum_{m \in \mathcal{A}_2(S_k)} \sum_{r \in \mathcal{R}} \left(w_{\rho_r} \Psi_\rho(\mu_m - t_r) - c_m \right) Y_{mr},$$

wherein w_{ρ_r} is a trade-off parameter that scales the contribution attained by the system based on the precedence level of request r; $\Psi_\rho(\mu_m - t_r)$ is a precedence-ρ based utility function Ψ_ρ that is monotonically decreasing with respect to the difference in time between the expected service time μ_m and the entry time t_r of request r; and c_m is the reward (i.e., contribution) attained by the system for MEDEVAC unit m's most recent incomplete request assignment.

4.2.5 Objective Function and Optimality Equation

Let $D^\pi(S_k) \in \mathcal{X}(S_k)$ represent the decision function that maps the state space to the action space, indicating the action (i.e., x_k) to take when the system is in state S_k based on a given policy π. The MDP model seeks to determine the optimal policy π^* from the class of policies $(D^\pi(S_k))_{\pi \in \Pi}$ to maximize the expected total discounted reward earned by the system over an infinite horizon. Accordingly, the objective is given by

$$\max_{\pi \in \Pi} \mathbb{E}^\pi \left[\sum_{k=1}^{\infty} \gamma^{\tau(S_k)} C\left(S_k, D^\pi(S_k)\right) \right],$$

wherein $\gamma \in (0, 1]$ is a fixed discount factor. The optimal policy π^* maximizes the expected total discounted contribution and satisfies the following optimality equation

$$V(S_k) = \max_{x_k \in \mathcal{X}(S_k)} \left(C(S_k, x_k) + \gamma^{(\hat{\tau}(S_{k+1}) - \tau(S_k))} \mathbb{E}\left[V(S_{k+1}) | S_k, x_k \right] \right), \qquad (53)$$

wherein $\hat{\tau}(S_{k+1})$ denotes the time at which the system visits state S_{k+1}.

4.3 Solution Methodology

Solving Equation (53) via exact dynamic programming methods (e.g., value iteration or policy iteration) is impractical due to what is commonly referred to as the *curse of dimensionality* (i.e., high dimensionality and/or uncountable state space). To overcome the curse of dimensionality, this research leverages an approximate dynamic programming (ADP) strategy to approximate the value function around the post-decision state variable.

Similar to Rettke et al. (2016) and Jenkins et al. (2019), a post-decision state convention is adopted to reduce the dimensionality of the state space and to avoid computing expectations explicitly within Equation (53) (Powell, 2011; Ruszczynski, 2010). The post-decision state captures the state of the system immediately after a decision is made but before any new information has arrived. Let S_k^x denote the post-decision state variable. The evolution of the aeromedical evacuation (MEDEVAC) system from pre-decision state S_k to post-decision state S_k^x is characterized by action x_t and a deterministic function $S^{M,x}$. As such, the post-decision state is computed as follows: $S_k^x = S^{M,x}(S_k, x_k)$. Utilizing this information, let

$$V^x(S_k^x) = \mathbb{E}\left[V(S_{k+1})|S_k^x\right] \tag{54}$$

denote the value of being in post-decision state S_k^x. By substituting Equation (54) into Equation (53), the optimality equation is modified to

$$V(S_k) = \max_{x_k \in \mathcal{X}(S_k)} \left(C(S_k, x_k) + \gamma^{(\hat{\tau}(S_{k+1}) - \tau(S_k))} V^x(S_k^x) \right). \tag{55}$$

Note that the value of being in post decision state S_{k-1}^x is given by

$$V^x(S_{k-1}^x) = \mathbb{E}\left[V(S_k)|S_{k-1}^x\right]. \tag{56}$$

By substituting Equation (55) into Equation (56), the optimality equation around the post-decision state is given by

$$V^x(S^x_{k-1}) = \mathbb{E}\left[\max_{x_k \in \mathcal{X}_{S_k}} \left(C(S_k, x_k) + \gamma^{(\hat{\tau}(S_{k+1})-\tau(S_k))} V^x(S^x_k)\right) \big| S^x_{k-1}\right]. \tag{57}$$

Despite the computational advantages attained via the post-decision state convention, solving the updated optimality equation (i.e., Equation (57)) remains impractical due to the size and dimensionality of the state space. One of the most powerful and visible methods for solving complex dynamic programs involves replacing the true value function with a statistical approximation (Powell, 2011). This research utilizes basis functions to capture important features of the post-decision state variable and builds a value function approximation from the resulting quantities by leveraging support vector regression (SVR) within an approximate policy iteration (API) algorithmic framework. Basis functions are appealing due to their relative simplicity, but the development of appropriate basis functions and features is an art form and depends on the problem being examined (Powell, 2011). As such, one of the key challenges in this research is to design basis functions and features that accurately contribute to the quality of the solution for the MEDEVAC dispatching, preemption-rerouting, and redeployment (DPR) problem.

Let $\phi_f(S^x_k)$ denote a basis function, where $f \in \mathcal{F}$ is a feature that draws information from post-decision state S^x_k, and $\mathcal{F}$ is the set of features. The basis functions utilized herein are presented next. The first set of basis functions,

$$\phi_f(S^x_k) = \zeta_m,$$

describes the status of each MEDEVAC unit and includes one feature per MEDEVAC unit $m \in \mathcal{M}$. Let $\mathbb{I}_{\{\delta_r = m\}}$ denote an indicator function that takes the value of one if

MEDEVAC unit m is currently servicing request r. The second set of basis functions,

$$\phi_f(S_k^x) = \mathbb{I}_{\{\delta_r = m\}}(\mu_m - \tau(S_k^x)),$$

captures the expected remaining service time associated with each active MEDEVAC unit-request pairing and includes one feature per MEDEVAC unit $m \in \mathcal{M}$, request $r \in \mathcal{R}$ pairing. Note that a pairing of MEDEVAC unit $m \in \mathcal{M}$ with request $r \in \mathcal{R}$ is considered active if MEDEVAC unit m is currently servicing request r. The third set of basis functions,

$$\phi_f(S_k^x) = \mathbb{I}_{\{\delta_r = m\}} p_m,$$

identifies the precedence level associated with each active MEDEVAC unit-request pairing and includes one feature per pairing. The fourth set of basis functions,

$$\phi_f(S_k^x) = \mathbb{I}_{\{\delta_r = 0\}} \rho_r,$$

captures the precedence level of each request currently *not* being serviced and includes one feature per request $r \in \mathcal{R}$. The fifth set of basis functions,

$$\phi_f(S_k^x) = (\tau(S_k^x) - t_r),$$

identifies the total time each request has been in the system and includes one feature per request $r \in \mathcal{R}$. The last set of basis functions,

$$\phi_f(S_k^x) = \bar{d}_{mr},$$

captures the total distance required for each MEDEVAC unit to service request r for each $r \in \mathcal{R}$ and includes one feature per MEDEVAC unit $m \in \mathcal{M}$, request

$r \in \mathcal{R}$ pairing. This basis function informs the system which MEDEVAC units have enough fuel to be dispatched or preempted-and-rerouted to service requests waiting for assignment.

Utilizing these basis functions, the post-decision state value function $V^x(S_k^x)$ is replaced with a linear approximation architecture given by

$$\bar{V}^x(S_k^x|\theta) = \sum_{f \in \mathcal{F}} \theta_f \phi_f^s(S_k^x) \equiv \theta^T \phi^s(S_k^x), \tag{58}$$

wherein $\theta = (\theta_f)_{f \in \mathcal{F}}$ is a column vector of basis functions weights and $\phi^s(S_k^x) = (\phi_f^s(S_k^x))_{f \in \mathcal{F}}$ is a column vector of scaled (i.e., normalized) basis function evaluations. Utilizing continuous data that cover different ranges can cause difficulty for some machine learning strategies (e.g., SVR). Normalization techniques can be used to transform continuous data to a specified range while maintaining the relative differences between the values (Kelleher et al., 2015). As such, this research adopts a normalization strategy that scales the value of each basis function evaluation between zero and one to make the data format suitable for the SVR strategy utilized. Substituting Equation (58) into Equation (57) yields the post-decision state value function approximation

$$\bar{V}^x(S_{k-1}^x|\theta) = \mathbb{E}\left[C\left(S_k, D^\pi(S_k|\theta)\right) + \gamma^{(\hat{\tau}(S_{k+1}) - \tau(S_k))} \bar{V}^x(S_k^x|\theta) \middle| S_{k-1}^x\right], \tag{59}$$

wherein the action x_k is determined via the decision function

$$D^\pi(S_k|\theta) = \underset{x_k \in \mathcal{X}_{S_k}}{\operatorname{argmax}} \left\{C(S_k, x_k) + \gamma^{(\hat{\tau}(S_{k+1}) - \tau(S_k))} \bar{V}^x(S_k^x|\theta)\right\}. \tag{60}$$

Having defined the decision function and the value function approximation upon which it is based, the manner in which the value function approximation is updated is presented next. This research employs an API algorithmic strategy, the general structure of which is derived from exact policy iteration, wherein a sequence of value function approximations and policies are generated from two repeated, alternating phases: policy evaluation and policy improvement. Within a policy evaluation phase, the value of a fixed policy is approximated via simulation. Within a policy improvement phase, a new policy is generated by leveraging the data collected in the previous policy evaluation phase. The API algorithm utilized herein is adapted in part from Jenkins et al. (2019) and is displayed in Algorithm 3.

Algorithm 3 Approximate Policy Iteration Algorithm

```
Initialize $\theta$.
for $i = 1$ to $I$ do
    for $j = 1$ to $J$ do
        Generate a random post-decision state $S^x_{k-1,j}$.
        Compute and record $\phi^s(S^x_{k-1,j})$.
        Simulate transition to next pre-decision state $S_{k,j}$.
        Compute and record $\hat{v}_j$ utilizing Equation (61).
    end for
    Compute $\hat{\theta}$ by solving the optimization problem given in (62)-(65).
    Update $\theta$ utilizing Equation (66).
end for
Return the approximate value function $\bar{V}^x(\cdot|\theta)$.
```

Algorithm 3 starts by initializing θ, which represents an initial base policy. The algorithm then enters a policy evaluation phase wherein, for each iteration $j = 1, \ldots, J$, the following steps occur. A post-decision state $S^x_{k-1,j}$ is randomly generated, and the associated vector of scaled basis function evaluations $\phi^s(S^x_{k-1,j})$ is computed and recorded. Next, the algorithm simulates the transition from post-decision state $S^x_{k-1,j}$ to pre-decision state $S_{k,j}$ and collects a sample realization of the value attained from the current policy via

$$\hat{v}_j = \max_{x_k \in \mathcal{X}_{S_{k,j}}} \left(C(S_{k,j}, x_k) + \gamma^{(\hat{\tau}(S_{k+1,j}) - \tau(S_k))} \bar{V}^x(S^x_{k,j}|\theta) \right). \tag{61}$$

After J iterations have been performed, the algorithm enters a policy improvement phase. Within each policy improvement phase, the algorithm leverages the training data set from the most recent policy evaluation phase (i.e., $(\phi^s(S^x_{k-1,j}), \hat{v}_j)$ for $j = 1, \ldots, J$) to compute a sample estimate of θ (i.e., $\hat{\theta}$) via SVR. The essential idea of SVR is to identify a function (e.g., $\bar{V}^x(\cdot|\hat{\theta})$) defined by the weights $\hat{\theta}$ such that most of the observed responses (e.g., $\hat{v}_j$) deviate from the function by a value no greater than ε for each input pattern (e.g., $\phi^s(S^x_{k-1,j})$) and at the same time is flat as possible. That is, SVR does not care about errors as long as they are within a pre-defined ε-insensitive zone; for more details and examples of SVR, see Smola and Schölkopf (2004), Hastie et al. (2009), and Cherkassky (2013). To find the sample estimate $\hat{\theta}$, SVR solves the following quadratic optimization problem:

$$\text{minimize } \frac{1}{2}||\hat{\theta}||^2 + \eta \sum_{j=1}^{J} (\xi_j + \xi_j^*) \tag{62}$$

$$\text{subject to } \hat{v}_j - \hat{\theta}^T \phi^s(S^x_{k-1,j}) \leq \varepsilon + \xi_j, \text{ for } j = 1, \ldots, J, \tag{63}$$

$$-\hat{v}_j + \hat{\theta}^T \phi^s(S^x_{k-1,j}) \leq \varepsilon + \xi_j^*, \text{ for } j = 1, \ldots, J, \tag{64}$$

$$\xi_j, \xi_j^* \geq 0 \text{ for } j = 1, \ldots, J. \tag{65}$$

Here, the regularization parameter η controls the trade-off between model complexity (i.e., the flatness of the function) and the margin-based error (i.e., the degree to which sample deviations beyond the ε-insensitive zone are tolerated). Note that each training sample j (i.e., $(\phi^s(S^x_{k-1,j}), \hat{v}_j)$) is associated with slack variables ξ_j and ξ_j^*, corresponding to sample deviations above and below the ε-insensitive zone, respectively. For training samples that fall inside the ε-insensitive zone, $\xi_j = \xi_j^* = 0$. For training samples that fall outside the ε-insensitive zone, only one of these coefficients is positive, and the other is zero.

Once $\hat{\theta}$ is computed, θ is updated via the following equation

$$\theta \leftarrow \alpha_i \hat{\theta} + (1 - \alpha_i)\theta, \tag{66}$$

wherein the θ on the right hand side of the equation is the previous estimate based on the first $i - 1$ policy improvement iterations and α_i is a stepsize rule. This research incorporates a polynomial stepsize rule given by

$$\alpha_i = \frac{1}{i^\beta},$$

wherein $\beta \in (0.3, 1]$ controls the rate at which the stepsizes decline. The best value of β depends on the problem being examined and must be tuned (Powell, 2011).

The algorithm completes a single policy improvement phase once θ is updated. If $i < I$, the algorithm starts a new policy evaluation phase. The algorithmic parameters I, J, ε, η, and β are tunable, where I is the number of policy improvement phase iterations; J is the number of policy evaluation phase iterations; ε is a parameter that controls the degree of overfitting via the width of the ε-insensitive zone; η is a regularization term that determines the trade-off between the flatness of approximate value function defined by the basis function weights and the degree to which sample deviations beyond the ε-insensitive zone are tolerated; and β is the polynomial stepsize parameter that controls the algorithm's rate of convergence. Concluding after I policy improvement phases, the algorithm provides the recommended policy θ and the corresponding approximate value function $\bar{V}^x(\cdot|\theta)$.

4.4 Testing, Results, & Analysis

This section utilizes a representative military aeromedical evacuation (MEDEVAC) planning scenario to demonstrate the applicability of the Markov decision process (MDP) model of the dispatching, preemption-rerouting, and redeployment (DPR) problem and to examine the efficacy of the proposed approximate dynamic programming (ADP) solution strategy. Computational experiments are designed and conducted to examine how different algorithmic parameter settings impact the performance of the ADP strategy. Comparisons are made between the ADP-generated policies and two benchmark policies, one of which is the currently practiced closest-available dispatching policy. This research utilizes a dual Intel Xeon E5-2650v2 workstation having 128 GB of RAM and invokes the commercial solver CPLEX 12.6 while leveraging MATLAB's Parallel Computing Toolbox to conduct the experiments and analyses presented herein.

4.4.1 Representative Scenario

This research utilizes the notional, representative military MEDEVAC planning scenario presented in Jenkins et al. (2019) wherein the United States (US) military is performing high-intensity combat operations in support of the Azerbaijan government. The scenario considers two main operating bases (i.e., bases that host both a medical treatment facility (MTF) and a MEDEVAC staging facility) and two forward operating bases (i.e., bases that only host a MEDEVAC staging facility) as well as 55 casualty cluster centers (i.e., areas in which casualty events are likely to occur), all of which are depicted in Figure 9 (Jenkins et al., 2019).

The characteristics (e.g., location, entry time, and precedence) of requests for MEDEVAC service are modeled by simulating a Poisson cluster process. The proportion of requests originating from a particular casualty cluster center depends on the casualty cluster center's location. This research considers a baseline instance wherein casualty events are more likely to occur in the south due to an invasion by a notional aggressor. The distribution of requests from a particular casualty cluster center is

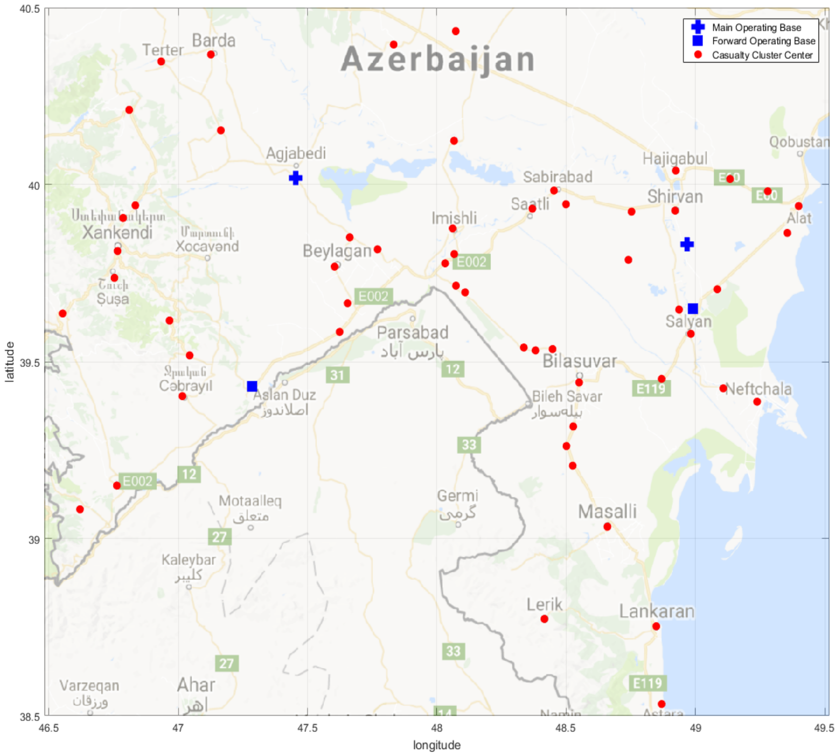


Figure 9. Representative Military MEDEVAC Planning Scenario Disposition

generated on a uniform distribution with respect to the distance of the request to the casualty cluster center. Moreover, requests enter the system sequentially over time according to a Poisson process with parameter λ. This assumption is reasonable within the military MEDEVAC context and is utilized in related efforts; see Keneally et al. (2016), Rettke et al. (2016), Jenkins et al. (2018), Robbins et al. (2018), and Jenkins et al. (2019) for more details.

The baseline instance considers an average request arrival rate of $\lambda = \frac{1}{40}$. That is, the average arrival rate of requests for MEDEVAC service is one request per 40 minutes. To portray a highly kinetic conflict, resulting in a large number of life-threatening (i.e., urgent) casualty events, the baseline proportions of priority and urgent requests are set to 0.2 and 0.8, respectively (i.e., $\mathbb{P}(\text{priority, urgent}) = (0.2, 0.8)$).

This research leverages distributions provided in Bastian (2010), Keneally et al. (2016), and Jenkins et al. (2018) to model the variability in the times associated with each simulated MEDEVAC mission (e.g., mission preparation, travel, and loading and unloading casualties). Moreover, unlike previous efforts, this research explicitly models the range of each MEDEVAC unit and only allows MEDEVAC units to service requests if they have enough fuel to reach the destination (i.e., MTF) and have a planned fuel reserve of 30 minutes at cruise speed, which aligns with US Army flight regulations (Department of the Army, 2018).

The trade-off parameters within the contribution function that are associated with servicing priority and urgent requests are set to $w_1 = 0.1$ and $w_2 = 1$, respectively, to prioritize urgent requests more than priority requests. Additionally, the discount factor $\gamma = 0.99$ is utilized within the computational experiments. This choice of γ exhibits relatively low discounting of operational values associated with future events and motivates the system to position itself to efficiently respond to future requests for service. Lastly, the maximum number of requests that can be tracked in the system is set to ten (i.e., $r^{max} = 10$).

4.4.2 Experimental Design

For comparison purposes, two benchmark policies are considered. The first benchmark policy (i.e., Benchmark 1) utilizes the default MEDEVAC dispatching policy in practice (i.e., the closest-available dispatching policy). Recall that this policy assigns the closest-available MEDEVAC unit to respond to a request for service regardless of the request's characteristics (e.g., location and precedence) or the MEDEVAC system state (e.g., location and status of MEDEVAC units). Moreover, this policy considers a MEDEVAC unit available only when the MEDEVAC unit is not servicing a request. As such, Benchmark 1 does not allow preemption-and-rerouting of MEDEVAC units.

The second benchmark policy (i.e., Benchmark 2) extends Benchmark 1 by allowing preemption-and-rerouting of MEDEVAC units. That is, Benchmark 2 allows the dispatching authority to preempt a MEDEVAC mission and re-route the MEDEVAC unit to service a more urgent, time sensitive request. Both Benchmark 1 and Benchmark 2 utilize a myopic approach, which is commonly utilized in resource allocation problems, that makes decisions based on the immediate expected reward and does not consider forecasted information or decisions that must be made in the future (Powell, 2011). Moreover, it is important to note that neither Benchmark 1 or Benchmark 2 consider MEDEVAC unit redeployment since there is no immediate reward associated with redeployment decisions.

The computational experiments herein measure performance as the percent increase of total discounted reward obtained over an approximated infinite horizon to that obtained via the Benchmark 1 and Benchmark 2 policies. The efficacy of ADP-generated policies varies based on different problem and algorithmic features. An important problem feature of interest is the average request arrival rate λ. The algorithmic features of interest include the number of policy improvement phase iterations, I; the number of policy evaluation phase iterations, J; the regularization term within the support vector regression (SVR) quadratic optimization problem, η; the ε-insensitive zone parameter, ε; and the polynomial stepsize parameter, β.

The optimal values associated with η and ε depend on the training data set. Each policy evaluation phase generates a new training data set and, as such, prescribing a fixed value for ε and η for every training data set may yield low quality solutions. Instead, this research utilizes estimation techniques presented in Cherkassky (2013) to determine the values for η and ε for each training data set. The value of η is computed via

$$\eta = \max\{|\bar{\hat{v}} + 3\sigma_{\hat{v}}|, |\bar{\hat{v}} - 3\sigma_{\hat{v}}|\},$$

wherein $\bar{\hat{v}}$ is the mean and $\sigma_{\hat{v}}$ is the standard deviation of the observed training responses, $\hat{v}$. The value of ε is computed via

$$\varepsilon = 3\hat{\sigma}\sqrt{\frac{\ln J}{J}},$$

wherein $\hat{\sigma}$ is the estimated standard deviation of additive noise. The term $\hat{\sigma}$ is determined by first solving the optimization problem given in (62)-(65) with $\varepsilon = 0$ and then applying the model to estimate the noise variance of training samples; see Cherkassky and Ma (2004) for more details.

A full factorial computational experiment is designed and conducted to examine the remaining algorithmic features of interest (i.e., I, J, and β) with a baseline average request arrival rate of $\lambda = \frac{1}{40}$. Table 9 shows the factor levels considered. To achieve the desired confidence level, 50 simulation runs are performed for each experimental design point. Moreover, each simulation run considers a 1,000-minute trajectory, which is a reasonable approximation because the model utilizes a discounting scheme and the $\gamma^{\tau(S_k)}$ term in the objective function becomes small enough when $\tau(S_k) = 1000$ such that a longer simulation does not impact the measure of performance. These experiments inform the selection of appropriate algorithmic parameter settings and provide general insight regarding the performance of the ADP-generated policies for a high-intensity MEDEVAC DPR problem instance.

Table 9. Experimental Design Factor Levels

Feature	Parameter Settings
I	$\{1, 2, \ldots, 40\}$
J	$\{500, 1000, 2000, 4000\}$
β	$\{0.3, 0.5, 0.7, 0.9\}$

4.4.3 Experimental Results

Table 10 reports the results from the experimental design. The three leftmost columns report the factor levels associated with the algorithmic parameter settings.

Note that Table 10 only reports the $I \in \{1, 2, \ldots, 40\}$ factor level corresponding to the best algorithmic performance for each (J, β) factor level combination. The remaining columns report the attendant solution qualities of the ADP policies determined at the experimental design points. These solution qualities provide a measure of algorithm efficacy and are expressed in terms of the 95% confidence intervals of percent improvement over the Benchmark 1 and Benchmark 2 policies with respect to total discounted reward over an approximated infinite horizon.

Table 10. Experimental Design Results

Parameter Settings			Percent Improvement Over	
I	J	β	Benchmark 1	Benchmark 2
30	500	0.3	8.2 ± 0.2	5.3 ± 0.2
35	1000	0.3	8.6 ± 0.2	5.6 ± 0.1
27	2000	0.3	9.5 ± 0.2	6.5 ± 0.2
17	4000	0.3	9.3 ± 0.2	6.4 ± 0.2
10	500	0.5	8.3 ± 0.2	5.3 ± 0.1
8	1000	0.5	8.3 ± 0.2	5.3 ± 0.1
27	**2000**	**0.5**	**9.6 ± 0.2**	**6.6 ± 0.2**
12	4000	0.5	8.2 ± 0.2	5.2 ± 0.1
10	500	0.7	8.4 ± 0.2	5.4 ± 0.1
7	1000	0.7	8.0 ± 0.1	5.1 ± 0.1
25	2000	0.7	9.0 ± 0.2	6.0 ± 0.2
4	4000	0.7	8.4 ± 0.2	5.4 ± 0.1
30	500	0.9	8.1 ± 0.2	5.2 ± 0.1
7	1000	0.9	7.8 ± 0.1	4.9 ± 0.1
33	2000	0.9	9.4 ± 0.2	6.4 ± 0.2
25	4000	0.9	8.4 ± 0.1	5.4 ± 0.1

The results from Table 10 indicate that the ADP algorithm is able to generate policies that significantly outperform both the Benchmark 1 and Benchmark 2 policies when $\lambda = \frac{1}{40}$. In particular, the best ADP policy occurs when $I = 27, J = 2000$, and $\beta = 0.5$, resulting in $9.6\% \pm 0.2\%$ and $6.6\% \pm 0.2\%$ improvements over the Benchmark 1 and Benchmark 2 policies, respectively.

Figure 10 illustrates the percent improvements over Benchmark 1 graphically and indicates that the best performing ADP policies for each β factor level occur when

$J = 2000$. The general trend in Figure 10 shows that performance improves as the level of J increases to 2000 and declines when $J = 4000$. This trend illustrates that large training data sets (e.g., $J = 4000$) may result in SVR models that overfit the data and, as such, result in lower-quality ADP policies. Moreover, the results from Table 10 and Figure 10 show that smaller β-levels typically correspond to greater improvements over the benchmark policies. Note that smaller β-levels (e.g., 0.3 and 0.5) have slower convergence rates, which improve the responsiveness in the presence of initial transient conditions (Powell, 2011).

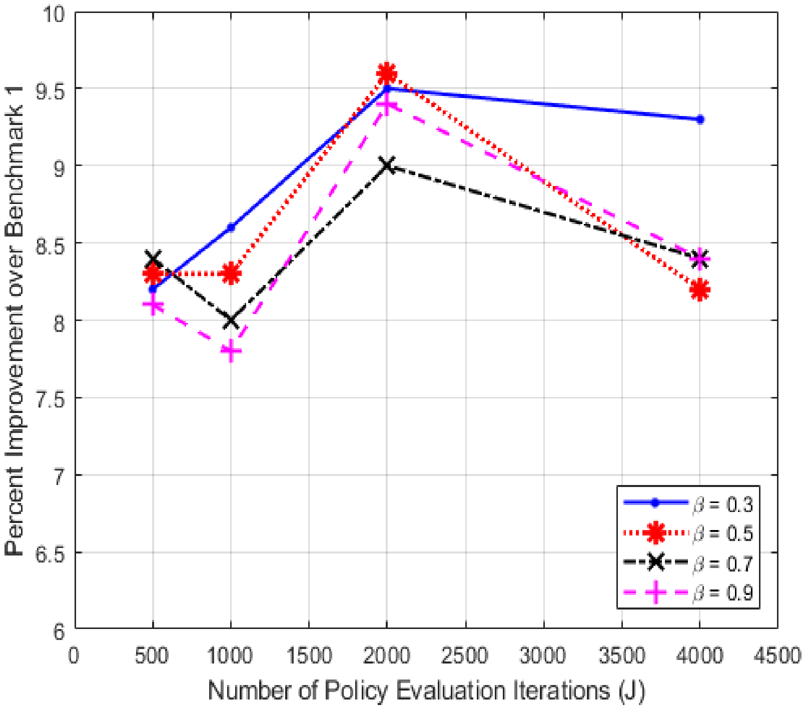


Figure 10. ADP Policies' Percent Improvement over Benchmark 1 Policy

The average waiting times associated with high-precedence (i.e., *urgent*) and low-precedence (i.e., *priority*) requests for the best performing ADP policy and the benchmark policies are reported in Table 11 as well as each policy's solution quality. Starting from the left, the first column indicates the policy. The second and third columns respectively report the the average response times for urgent and priority requests in terms of their 95% confidence intervals. The last column reports solution quality, measured in terms of the 95% confidence interval over the Benchmark 1 policies with respect to total discounted reward over an approximated infinite horizon.

Table 11. Performance of ADP and Benchmark Policies

	Service Times (min.)		Percent Improvement Over
Policy, π	Urgent	Priority	Benchmark 1
Benchmark 1	56.2 ± 0.2	56.8 ± 0.2	-
Benchmark 2	53.4 ± 0.1	57.6 ± 0.2	2.8 ± 0.1
ADP	48.1 ± 0.1	57.9 ± 0.1	9.6 ± 0.2

The results from Table 11 indicate that the incorporation of preemption-and-rerouting of MEDEVAC units significantly improves system performance when $\lambda = \frac{1}{40}$. Recall that the Benchmark 1 policy does not allow for preemption-and-rerouting of MEDEVAC units whereas both the Benchmark 2 and ADP policies do. As shown in Table 11, allowing the dispatching authority to preempt a low-precedence MEDEVAC mission to reroute the MEDEVAC unit to service a high-precedence request results in better system performance. Another interesting result from Table 11 is that the average service time of urgent requests decreases whereas the average service time of priority requests increases as the performance over the Benchmark 1 policy increases. This result aligns with intuition since the act of preempting priority MEDEVAC missions to reroute MEDEVAC units to service urgent requests will increase service time for priority requests and decrease service time for urgent requests.

4.4.4 On the Value of More Information

To illustrate the benefits of utilizing an ADP policy over the benchmark policies, consider the following vignettes: *wait*, *preempt-and-reroute*, and *redeploy*. For the *wait* vignette, consider a system state wherein a request for service has just entered the system, a distant MEDEVAC unit is available for dispatch, and a busy MEDEVAC unit is almost done completing service nearby the just submitted request for service. Under the Benchmark 1 and Benchmark 2 policies, the available (but distant) MEDEVAC unit is dispatched to service the new request, which results in a lower reward attained by the system when compared to queueing the request and waiting for the busy MEDEVAC to complete service and then assigning it to the queued submitted request.

For the *preempt-and-reroute* vignette, consider a system state wherein a MEDEVAC unit has just been tasked to service a low-precedence request in a high demand area, which is shortly followed by a high-precedence request being submitted from the same area. Under the Benchmark 1 policy, the busy MEDEVAC unit must complete service before the dispatching authority can task it to service the new high-precedence request. Clearly the system benefits by preempting the busy MEDEVAC unit and rerouting it to service the more time-sensitive, high-precedence request, resulting in a higher reward attained by the system.

For the *redeploy* vignette, consider a system state wherein a busy MEDEVAC unit is almost finished transferring casualty care to an MTF's staff and a new request enters the system that is just outside of the MEDEVAC units distance threshold. Under the Benchmark 1 policy, the MEDEVAC unit must return to its original staging area to refuel prior to being tasked to service the new request, resulting in a lower reward attained by the system when compared to redeploying the MEDEVAC to a closer staging facility to refuel and then rapidly servicing the new request. Unlike the benchmark policies, the policies generated by the ADP algorithm utilize important features of the system state, which are captured via the basis functions, to make decisions based on both current and future events, yielding better decisions and higher system performance returns.

The impact each basis function has on ADP-generated policies is determined via their weights (i.e., θ-values). Examination of the θ-values corresponding to the basis functions for the best ADP-generated policy when $\lambda = \frac{1}{40}$ reveals the following insights. The fourth set of basis functions, which corresponds to the precedence level of requests not being serviced, has the largest impact on the policy. This relative impact is assessed via the relatively larger θ-values associated with the fourth set of basis functions, as compared to the θ-values associated with the other sets of basis functions. Moreover, requests that have been waiting longer have a higher impact on the policy (i.e., they have larger θ-values) for the fourth set of basis functions.

The analysis also reveals that the last set of basis functions, which reveals which requests are reachable for each MEDEVAC unit, has the second largest impact on the policy. Interestingly, the weights associated with MEDEVAC units located *farther* away from the high demand request area (i.e., the two bases on the left hand side of the map in Figure 9) have relatively larger θ-values for the last set of basis functions as compared to the other bases.

The second set of basis functions, which captures the remaining time in the system of requests currently being serviced, also has a large impact on the policy. The θ-values for the second set of basis functions reveal that MEDEVAC units located *closer* to the high demand request area (i.e., the two bases on the right hand side of the map in Figure 9) are substantially larger than the remaining bases.

The last three sets of basis functions have relatively low θ-values, indicating that they have little-to-no impact on the best performing ADP-generated policy. The θ-values associated with each basis function changes when important problem features, such as λ, changes. As the value of λ approaches zero, the θ-values associated with each set of basis functions decrease. Moreover, as θ-values decreases towards zero the ADP-generated policy perform more similarly to that of a myopic approach. That is, as the average rate at which requests enter the system decreases, the difference between ADP-generated and myopic-generated policies decreases.

4.4.5 Sensitivity Analysis on Request Arrival Rate

As stated earlier, an interesting problem feature is the average rate at which requests enter the system (i.e., λ). Figure 11 plots the percent improvement of the ADP and Benchmark 2 policies over the Benchmark 1 policy for when the value of $\frac{1}{\lambda}$ varies between 20 and 60.

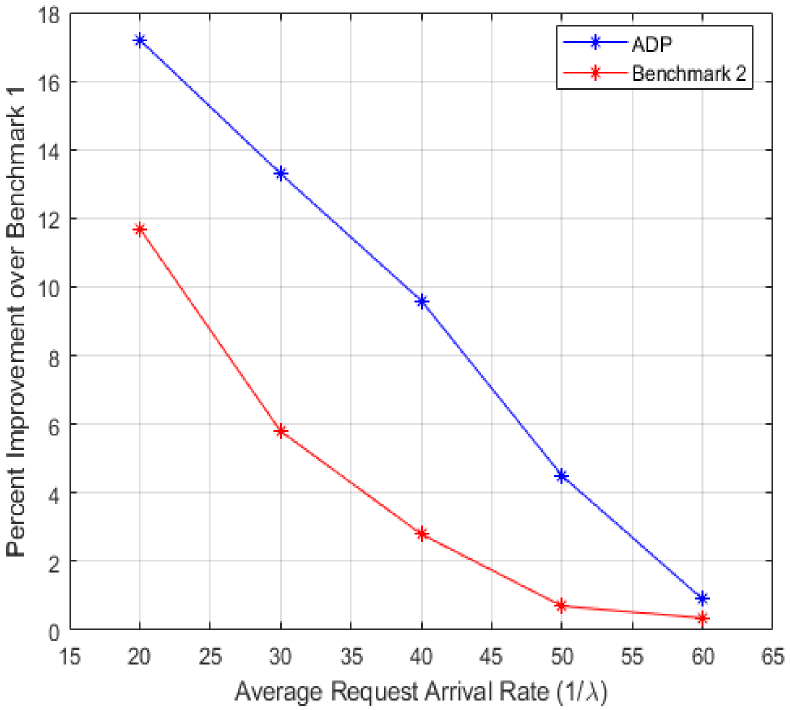


Figure 11. Sensitivity of Performance to Average Request Arrival Rate

The results from Figure 11 show that as λ decreases, the frequency in which requests enter the system decreases, and the performance improvements over the Benchmark 1 policy decreases for both the ADP and Benchmark 2 policies. As requests enter the system at a slower rate, the advantage of waiting, preempting-and-rerouting, and redeploying diminishes. Accordingly, implementing an ADP policy in low-intensity conflicts wherein the average request arrival rate is relatively low (e.g., $\lambda = \frac{1}{60}$ for this planning scenario) will likely result in marginal performance gains over the Benchmark 1 and Benchmark 2 policies. Conversely, as requests enter the system at a faster rate, the advantage of waiting, preempting-and-rerouting, and redeploying increases, revealing the benefits of adopting an ADP policy when conflict intensity is high.

4.5 Conclusions

This chapter examines the military aeromedical evacuation (MEDEVAC) dispatching, preemption-rerouting, and redeployment (DPR) problem. The intent of this research is to determine high-quality DPR policies that improve the performance of United States Army MEDEVAC systems and hence increase the combat casualty survivability rate. A discounted, infinite-horizon Markov decision process (MDP) model of the MEDEVAC DPR problem is formulated and solved via an approximate dynamic programming (ADP) strategy that utilizes a support vector regression (SVR) value function approximation scheme within an approximate policy iteration (API) framework. The objective is to maximize expected total discounted reward attained by the system. The applicability of the MDP model is examined via a notional, representative planning scenario based on high-intensity combat operations in Azerbaijan. Computational experimentation is performed to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated DPR policies and to highlight the efficacy of the proposed solution methodology.

The results from the computational experiments indicate the ADP-generated policies significantly outperform the two benchmark policies considered. Moreover, the results reveal that the average service time of high-precedence, time sensitive (i.e., *urgent*) requests decreases when an ADP policy is adopted during high-intensity conflicts. As the rate in which requests enter the system increases, the performance gap between the ADP policy and the Benchmark 1 policy (i.e., the currently practiced closest-available dispatching policy) increases substantially. Conversely, as the rate in which request enter the system decreases, the ADP performance improvement over both benchmark policies decreases, revealing that the ADP policy provides little-to-no benefit over a myopic approach (e.g., as utilized in the benchmark policies) when the intensity of a conflict is low. This result comports with similar research efforts (e.g., Rettke et al. (2016) and Jenkins et al. (2019)) examining MEDEVAC dispatching models having lesser operational fidelity (and which used different algorithmic techniques).

This research can be applied to the analysis of both civilian and military emergency medical service (EMS) response systems. Comparisons can be made between currently practiced policies and the ADP polices determined via the approach proposed in this chapter for a variety of planning scenarios that have fixed staging facilities (i.e., ambulance bases) and medical treatment facilities (i.e., hospitals). Moreover, medical planners can examine different location schemes and apply this methodology to determine the best allocation of medical resources. Such comparisons inform the implementation and modification of current and future tactics, techniques, and procedures for both civilian and military EMS response systems.

V. Conclusions and Recommendations

This dissertation considers the importance of optimizing deployed military MEDEVAC systems and utilizes operations research techniques to develop models that allow military medical planners to analyze different strategies regarding the management of MEDEVAC assets in a deployed environment. For optimization models relating to selected subproblems of the MEDEVAC enterprise, the work herein leverages integer programming, multi-objective optimization, Markov decision processes (MDPs), approximate dynamic programming (ADP), and machine learning, as appropriate, to identify relevant insights for aerial MEDEVAC operations. Moreover, realistic, but notional, computational examples are utilized to illustrate the impact and relevance of the models developed in this dissertation.

The research presented in this dissertation is of interest to both military and civilian medical planners. Medical planners can apply the models within the dissertation to compare different location, allocation, relocation, dispatching, preempting-rerouting, and redeployment MEDEVAC policies for a variety of planning scenarios. Such comparisons inform the implementation and modification of current and future MEDEVAC tactics, techniques, and procedures, as well as the design, evaluation, and purchase of future aerial MEDEVAC assets.

5.1 Conclusions

Chapter II examines the MEDEVAC location-allocation problem wherein military medical planners must decide where to locate mobile aeromedical staging facilities (MASFs) and, implicitly, co-located MTFs as well as identify how many aeromedical helicopters to allocate to each located MASF throughout the phases of a deployment. An integer mathematical programming formulation is constructed to determine the location and allocation of MEDEVAC assets for each phase of a deployment. Whereas the identification of an optimal coverage solution for each phase may require a large number of located MASFs and a significant number of MEDEVAC asset relocations as a force transitions between phases, the model also seeks to minimize both the maximum number of located MASFs in any deployment phase and the total number of MASF relocations throughout the deployment. More specifically, the model seeks to address the multi-objective problem of maximizing the total expected coverage of demand as a measure of solution effectiveness, minimizing the maximum number of located MASFs in any deployment phase as a measure of solution efficiency, and minimizing the total number of MASF relocations throughout the deployment as a measure of solution robustness.

The location modeling research presented in Chapter II makes the following contributions. First, it formulates a representative mathematical programming formulation and identifies an accompanying solution methodology to assess and recommend improvements to deployed military MEDEVAC systems designed to provide large-scale emergency medical response for contingency operations that range in casualty-inducing intensity over the phases of a deployment. Second, the research illustrates the application of the model for a realistic, synthetically generated medical planning scenario in southern Azerbaijan. Comparisons are made between the models (multi-phase) optimal solution and the phase-specific optimal solutions that disregard concerns of solution robustness.

Chapter III examines the MEDEVAC dispatching problem wherein a dispatching authority must decide which (if any) MEDEVAC unit to dispatch in response to a submitted 9-line MEDEVAC request. A discounted, infinite-horizon MDP model of the MEDEVAC dispatching problem is formulated to maximize the expected total discounted reward attained by the system. Whereas the MDP model provides an appropriate mathematical framework for solving the MEDEVAC dispatching problem, classical dynamic programming techniques (e.g., policy iteration or value iteration) are computationally intractable due to the high dimensionality and uncountable state space of practical scenarios (i.e., large-scale problem instances). As such, two ADP strategies are designed, tested, and employed to produce high-quality dispatching policies relative to the currently practiced dispatching policy (i.e., closest-available dispatching policy). The first ADP strategy utilizes least-squares temporal differences learning within an approximate policy iteration (API) algorithmic framework, whereas the second strategy leverages neural network (NN) learning within an API algorithmic framework. Utilizing features from the MEDEVAC dispatching problem, a set of basis functions is defined to approximate the value function around the post-decision state for both ADP strategies. A notional, representative planning scenario based on high-intensity combat operations in southern Azerbaijan is constructed to demonstrate the applicability of the MDP model and to examine the efficacy of the proposed ADP strategies. Moreover, designed computational experiments are conducted to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated dispatching policies.

An important difference between the military MEDEVAC dispatching research presented in Chapter III and similar research efforts is the incorporation of redeployment. This aspect gives the dispatching authority the ability to task a MEDEVAC unit to service incoming or queued requests directly after the MEDEVAC unit completes service at an MTFs co-located MEDEVAC staging area (i.e., completes refueling or aircraft and re-equipping of MEDEVAC supplies). This relaxes the restriction that MEDEVAC units must return to their own staging areas to refuel and re-equip after delivering combat casualties to an MTF prior to being tasked with another service request, which is a recognized limitation of previous work. Since MTFs are co-located with MEDEVAC staging areas, it is reasonable to assume that MEDEVAC units can refuel and re-equip at an MTF's co-located MEDEVAC staging area immediately after the MEDEVAC unit transfers combat casualties to the MTF staff, especially during high-intensity combat operations. In addition to the incorporation of redeployment, Chapter III jointly considers the relevant problem features examined in earlier research efforts, including admission control, queueing, and explicit modeling of the number of casualties per casualty event. Lastly, Chapter III demonstrates the improved efficacy of an NN-based ADP strategy for the MEDEVAC dispatching problem, as compared to an ADP strategy previously offered in the literature (e.g., in Rettke et al. (2016)).

Chapter IV examines the MEDEVAC dispatching, preemption-rerouting, and redeployment (DPR) problem wherein a decision maker seeks a policy that determines which MEDEVAC units to assign (i.e., dispatch or preempt-and-reroute) to respond to requests for service and where MEDEVAC units redeploy after finishing a service request (i.e., successfully transferred casualty care to an MTF's staff). A discounted, infinite-horizon MDP model of the MEDEVAC DPR problem is formulated and solved via an ADP strategy that utilizes a support vector regression (SVR) value function approximation scheme within an API framework. The objective is to generate high-quality policies that dispatch, preempt-and-reroute, and redeploy MEDEVAC units in a way that improves upon the currently practiced, closest-available dispatching policy for large-scale, highly kinetic, and intense military conflicts.

The military MEDEVAC dispatching research presented in Chapter IV makes the following contributions. First, it defines the military MEDEVAC DPR problem, which extends prior work on the MEDEVAC dispatching problem by simultaneously determining dispatching decisions in conjunction with preemption-rerouting and redeployment decisions. Preemption has yet to be addressed in either the military or civilian EMS literature. A realistic MDP model of the MEDEVAC DPR problem is formulated that includes: queueing; prioritized MEDEVAC requests; fuel constraints; dispatching decisions; preemption-rerouting decisions; and redeployment decisions. Most notably, the incorporation of fuel constraints, preemption-rerouting decisions, and redeployment decisions are significant as they have not been examined in related efforts regarding military MEDEVAC systems (e.g., Keneally et al. (2016), Rettke et al. (2016), and Jenkins et al. (2018), Robbins et al. (2018), Jenkins et al. (2019)). Second, this chapter provides the first proposal and demonstration of the relative efficacy of an ADP strategy that utilizes an SVR value function approximation scheme within an API algorithmic framework for military MEDEVAC problems. Moreover, this solution methodology has not been applied within the literature regarding civilian EMS response systems. Third, it examines the applicability of the MDP model and highlights the efficacy of the proposed solution methodology for a notional, representative planning scenario based on high-intensity combat operations in Azerbaijan. Computational experimentation is performed to determine how selected problem features and algorithmic features impact the quality of solutions attained by the ADP-generated DPR policies. Fourth, the modeling and solution approach presented in this work is transferable to civilian EMS response systems, particularly those located in areas that heavily rely on air evacuation platforms as the primary means for service (e.g., vast rural regions) or for aeromedical natural disaster response.

5.2 Recommendations for Future Research

Whereas the models presented in this dissertation prove to be useful to the medical planning community, there also exists several opportunities for future research. For example, the location modeling research can be extended via the consideration of different objectives (e.g., minimize the maximum distance traveled) and the exploration of different solution methodologies (e.g., weighted sum and lexicographic). Moreover, the incorporation of precedence levels, response times, and service times can enhance the accuracy and applicability of the integer programming formulation, ultimately improving the reliability of the trade-off analysis generated.

Several limiting assumptions are made in the MEDEVAC dispatching research that can be relaxed in future work. For example, it is assumed that every MTF has the capability and capacity to handle all incoming requests for service and each request is transported to the nearest MTF available. These assumptions can be removed by explicitly modeling the level and capacity of each MTF and developing a framework that determines locations to which casualties are delivered. Moreover, high-intensity conflicts may result in casualty events wherein the number of casualties within a single request for service may exceed the capacity of currently utilized MEDEVAC platforms, requiring the dispatch of more than one MEDEVAC unit. This phenomenon is not modeled in this research, but can be incorporated in an extension. Another important extension is the modeling of different types of MEDEVAC platforms and their associated attributes (e.g., speed and capacity). Within this extension, one could investigate the trade-offs between employing smaller capacity, but faster MEDEVAC platforms versus employing larger capacity, but slower MEDEVAC platforms. Furthermore, precedence (i.e., severity) classification errors during the initial triage of casualties can result in improper use of MEDEVAC assets and should be considered in future work when developing dispatching policies.

Lastly, the examination of different ADP strategies (e.g., least squares policy evaluation, kernel regression, Q-learning, and SARSA) and ADP parameter tuning approaches (e.g., response surface methodology) may reveal superlative results as compared to those presented in this research and are the most prominent areas for future research.

Bibliography

Adams, E. (2016), 'The US may finally get a cheap alternative to the $70 million V-22 Osprey', `https://www.wired.com/2016/07/bell-v280-valor-v22-osprey-replacement/`. Retrieved on 9 April 2018.

Aringhieri, R., Bruni, M. E., Khodaparasti, S. and van Essen, J. (2017), 'Emergency medical services and beyond: Addressing new challenges through a wide literature review', *Computers & Operations Research* **78**, 349–368.

Baker, J. R., Clayton, E. R. and Taylor III, B. W. (1989), 'A non-linear multi-criteria programming approach for determining county emergency medical service ambulance allocations', *Journal of the Operational Research Society* **40**(5), 423–432.

Bandara, D., Mayorga, M. E. and McLay, L. A. (2014), 'Priority dispatching strategies for EMS systems', *Journal of the Operational Research Society* **65**(4), 572–587.

Bastian, N. D. (2010), 'A robust, multi-criteria modeling approach for optimizing aeromedical evacuation asset emplacement', *The Journal of Defense Modeling and Simulation* **7**(1), 5–23.

Batta, R., Dolan, J. M. and Krishnamurthy, N. N. (1989), 'The maximal expected covering location problem: Revisited', *Transportation Science* **23**(4), 277–287.

Bélanger, V., Ruiz, A. and Soriano, P. (2019), 'Recent optimization models and trends in location, relocation, and dispatching of emergency medical vehicles', *European Journal of Operational Research* **272**(1), 1–23.

Berlin, G. N. and Liebman, J. C. (1974), 'Mathematical analysis of emergency ambulance location', *Socio-Economic Planning Sciences* **8**(6), 323–328.

Berman, O. (1981), 'Repositioning of distinguishable urban service units on networks', *Computers & Operations Research* **8**(2), 105–118.

Bérubé, J.-F., Gendreau, M. and Potvin, J.-Y. (2009), 'An exact ε-constraint method for bi-objective combinatorial optimization problems: Application to the traveling salesman problem with profits', *European Journal of Operational Research* **194**(1), 39–50.

Bradley, M., Nealiegh, M., Oh, J. S., Rothberg, P., Elster, E. A. and Rich, N. M. (2017), 'Combat casualty care and lessons learned from the past 100 years of war', *Current Problems in Surgery* **54**(6), 315–351.

Brotcorne, L., Laporte, G. and Semet, F. (2003), 'Ambulance location and relocation models', *European Journal of Operational Research* **147**(3), 451–463.

Burland, B. R. (2008), *Digital Warfighter Experiment 2008 (DWE08) Final Report: Battle Command Battle Laboratory Leavenworth*, Fort Leavenworth, Kansas.

Carrizosa, E., Ushakov, A. and Vasilyev, I. (2015), 'Threshold robustness in discrete facility location problems: a bi-objective approach', *Optimization Letters* **9**(7), 1297–1314.

Carter, G. M., Chaiken, J. M. and Ignall, E. (1972), 'Response areas for two emergency units', *Operations Research* **20**(3), 571–594.

Chaiken, J. M. and Larson, R. C. (1972), 'Methods for allocating urban emergency units: a survey', *Management Science* **19**(4), 110–130.

Chankong, V. and Haimes, Y. Y. (2008), *Multiobjective Decision Making: Theory and Methodology*, Courier Dover Publications, New York.

Chanta, S., Mayorga, M. E. and McLay, L. A. (2014), 'Improving emergency service in rural areas: a bi-objective covering location model for EMS systems', *Annals of Operations Research* **221**(1), 133–159.

Cherkassky, V. (2013), *Predictive Learning*, VCtextbook, Minneapolis, Minnesota.

Cherkassky, V. and Ma, Y. (2004), 'Practical selection of svm parameters and noise estimation for svm regression', *Neural Networks* **17**(1), 113–126.

Church, R. and ReVelle, C. (1974), 'The maximal covering location problem', *Papers in Regional Science* **32**(1), 101–118.

Clarke, J. E. and Davis, P. R. (2012), 'Medical evacuation and triage of combat casualties in Helmand Province, Afghanistan: October 2010-April 2011', *Military Medicine* **177**(11), 1261–1266.

Cohon, J. L. (2013), *Multiobjective Programming and Planning*, Courier Corporation, Mineola, New York.

Current, J. R., ReVelle, C. and Cohon, J. L. (1985), 'The maximum covering/shortest path problem: A multiobjective network design and routing formulation', *European Journal of Operational Research* **21**(2), 189–199.

Daskin, M. S. and Stern, E. H. (1981), 'A hierarchical objective set covering model for emergency medical service vehicle deployment', *Transportation Science* **15**(2), 137–152.

De Lorenzo, R. A. (2003), Military casualty evacuation: MEDEVAC, *in* 'Aeromedical Evacuation', Springer, pp. 45–59.

Deb, K. (2014), Multi-objective optimization, *in* 'Search methodologies', Springer, pp. 403–449.

Department of Defense, U. S. (2012), *Joint Publication 4-02, Health Service Support*, Washington, DC.

Department of Defense, U. S. (2017), *Joint Publication 5-0, Joint Operation Planning*, Washington, DC.

Department of the Army, U. S. (2016), *Field Manual 4-02.2, Medical Evacuation.*

Department of the Army, U. S. (2018), *Army Regulation 95-1 Aviation Flight Regulations.*

Drezner, J. A., Roshan, P. and Whitmore, T. (2017), *Enhancing Management of the Joint Future Vertical Lift Initiative*, Rand Corporation.

Eastridge, B. J., Mabry, R. L., Seguin, P., Cantrell, J., Tops, T., Uribe, P., Mallett, O., Zubko, T., Oetjen-Gerdes, L., Rasmussen, T. E. et al. (2012), 'Death on the battlefield (2001–2011): implications for the future of combat casualty care', *Journal of Trauma and Acute Care Surgery* **73**(6), S431–S437.

Ehrgott, M. (2013), *Multicriteria Optimization*, Vol. 491, Springer Science & Business Media, Heidelberg, Germany.

Elster, E. A. and Bailey, J. (2017), 'Prehospital blood transfusion for combat casualties', *JAMA* **318**(16), 1548–1549.

Erkut, E., Ingolfsson, A. and Erdoğan, G. (2008), 'Ambulance location for maximum survival', *Naval Research Logistics* **55**(1), 42–58.

Fulton, L. V., Lasdon, L. S., McDaniel Jr, R. R. and Coppola, M. N. (2010), 'Two-stage stochastic optimization for the allocation of medical assets in steady-state combat operations', *The Journal of Defense Modeling and Simulation* **7**(2), 89–102.

Garrett, M. X. (2013), USCENTCOM review of MEDEVAC procedures in Afghanistan, Technical report, United States Central Command.

Grannan, B. C., Bastian, N. D. and McLay, L. A. (2015), 'A maximum expected covering problem for locating and dispatching two classes of military medical evacuation air assets', *Optimization Letters* **9**(8), 1511–1531.

Green, L. and Kolesar, P. (1984), 'A comparison of the multiple dispatch and M/M/c priority queueing models of police patrol', *Management Science* **30**(6), 665–670.

Green, L. V. and Kolesar, P. J. (2004), 'Anniversary article: Improving emergency responsiveness with management science', *Management Science* **50**(8), 1001–1014.

Gutjahr, W. J. and Nolz, P. C. (2016), 'Multicriteria optimization in humanitarian aid', *European Journal of Operational Research* **252**(2), 351–366.

Hall, W. K. (1972), ‘The application of multifunction stochastic service systems in allocating ambulances to an urban area’, *Operations Research* **20**(3), 558–570.

Hastie, T., Tibshirani, R. and Friedman, J. (2009), *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, 2nd edn, Springer, New York.

Howard, J. T., Kotwal, R. S., Santos-Lazada, A. R., Martin, M. J. and Stockinger, Z. T. (2018), ‘Reexamination of a battlefield trauma golden hour policy’, *Journal of Trauma and Acute Care Surgery* **84**(1), 11–18.

Ignall, E., Carter, G. and Rider, K. (1982), ‘An algorithm for the initial dispatch of fire companies’, *Management Science* **28**(4), 366–378.

Ingolfsson, A. (2013), EMS planning and management, *in* ‘Operations Research and Health Care Policy’, Springer, pp. 105–128.

Jenkins, P. R., Robbins, M. J. and Lunday, B. J. (2018), ‘Examining military medical evacuation dispatching policies utilizing a Markov decision process model of a controlled queueing system’, *Annals of Operations Research* **271**(2), 641–678.

Jenkins, P. R., Robbins, M. J. and Lunday, B. J. (2019), Approximate dynamic programming for military medical evacuation dispatching policies, Technical report, Air Force Institute of Technology.

Kelleher, J. D., Mac Namee, B. and D’arcy, A. (2015), *Fundamentals of Machine Learning for Predictive Data Analytics: Algorithms, Worked Examples, and Case Studies*, MIT Press.

Keneally, S. K., Robbins, M. J. and Lunday, B. J. (2016), ‘A Markov decision process model for the optimal dispatch of military medical evacuation assets’, *Health Care Management Science* **19**(2), 111–129.

Khodaparasti, S., Maleki, H., Bruni, M. E., Jahedi, S., Beraldi, P. and Conforti, D. (2016), ‘Balancing efficiency and equity in location-allocation models with an application to strategic ems design’, *Optimization Letters* **10**(5), 1053–1070.

Knight, V. A., Harper, P. R. and Smith, L. (2012), ‘Ambulance allocation for maximal survival with heterogeneous outcome measures’, *Omega* **40**(6), 918–926.

Kolesar, P. and Walker, W. E. (1974), ‘An algorithm for the dynamic relocation of fire companies’, *Operations Research* **22**(2), 249–274.

Kotwal, R. S., Howard, J. T., Orman, J. A., Tarpey, B. W., Bailey, J. A., Champion, H. R., Mabry, R. L., Holcomb, J. B. and Gross, K. R. (2016), ‘The effect of a golden hour policy on the morbidity and mortality of combat casualties’, *JAMA Surgery* **151**(1), 15–24.

Koucheravy, R. (2018), Personal Communication. Rich Koucheravy is the business development director for the future vertical lift aircraft being developed by Sikorsky, the SB-1 Defiant.

Kuisma, M., Holmström, P., Repo, J., Määttä, T., Nousila-Wiik, M. and Boyd, J. (2004), 'Prehospital mortality in an EMS system using medical priority dispatching: a community based cohort study', *Resuscitation* **61**(3), 297–302.

Larson, R. C. (1974), 'A hypercube queuing model for facility location and redistricting in urban emergency services', *Computers & Operations Research* **1**(1), 67–95.

Lejeune, M. A. and Margot, F. (2018), 'Aeromedical battlefield evacuation under endogenous uncertainty in casualty delivery times', *Management Science* **64**(12), 5461–5480.

Majzoubi, F., Bai, L. and Heragu, S. S. (2012), 'An optimization approach for dispatching and relocating EMS vehicles', *IIE Transactions on Healthcare Systems Engineering* **2**(3), 211–223.

Malsby III, R. F., Quesada, J., Powell-Dunford, N., Kinoshita, R., Kurtz, J., Gehlen, W., Adams, C., Martin, D. and Shackelford, S. (2013), 'Prehospital blood product transfusion by US Army MEDEVAC during combat operations in Afghanistan: a process improvement initiative', *Military Medicine* **178**(7), 785–791.

Marler, R. T. and Arora, J. S. (2004), 'Survey of multi-objective optimization methods for engineering', *Structural and Multidisciplinary Optimization* **26**(6), 369–395.

Marler, R. T. and Arora, J. S. (2010), 'The Weighted Sum Method for multi-objective optimization: new insights', *Structural and Multidisciplinary Optimization* **41**(6), 853–862.

Maxwell, M. S., Restrepo, M., Henderson, S. G. and Topaloglu, H. (2010), 'Approximate dynamic programming for ambulance redeployment', *INFORMS Journal on Computing* **22**(2), 266–281.

Mayorga, M. E., Bandara, D. and McLay, L. A. (2013), 'Districting and dispatching policies for emergency medical service systems to improve patient survival', *IIE Transactions on Healthcare Systems Engineering* **3**(1), 39–56.

McLay, L. A. and Mayorga, M. E. (2010), 'Evaluating emergency medical service performance measures', *Health Care Management Science* **13**(2), 124–136.

McLay, L. A. and Mayorga, M. E. (2013*a*), 'A dispatching model for server-to-customer systems that balances efficiency and equity', *Manufacturing & Service Operations Management* **15**(2), 205–220.

McLay, L. A. and Mayorga, M. E. (2013*b*), 'A model for optimally dispatching ambulances to emergency calls with classification errors in patient priorities', *IIE Transactions* **45**(1), 1–24.

Nasrollahzadeh, A. A., Khademi, A. and Mayorga, M. E. (2018), 'Real-time ambulance dispatching and relocation', *Manufacturing & Service Operations Management* **20**(3), 467–480.

Nicholl, J., Coleman, P., Parry, G., Turner, J. and Dixon, S. (1999), 'Emergency priority dispatch systems - a new era in the provision of ambulance services in the UK', *Pre-Hospital Immediate Care* **3**(2), 71–75.

Paul, N. R., Lunday, B. J. and Nurre, S. G. (2017), 'A multiobjective, maximal conditional covering location problem applied to the relocation of hierarchical emergency response facilities', *Omega* **66**, 147–158.

Powell, W. B. (2011), *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, 2nd edn, John Wiley & Sons, Princeton, New Jersey.

Rath, S. and Gutjahr, W. J. (2014), 'A math-heuristic for the warehouse location–routing problem in disaster relief', *Computers & Operations Research* **42**, 25–39.

Rettke, A. J., Robbins, M. J. and Lunday, B. J. (2016), 'Approximate dynamic programming for the dispatch of military medical evacuation assets', *European Journal of Operational Research* **254**(3), 824–839.

Robbins, M. J., Jenkins, P. R., Bastian, N. D. and Lunday, B. J. (2018), 'Approximate dynamic programming for the aeromedical evacuation dispatching problem: Value function approximation utilizing multiple level aggregation', *Omega* **(in press)**.

Ruszczynski, A. (2010), 'Commentary–post-decision states and separable approximations are powerful tools of approximate dynamic programming', *INFORMS Journal on Computing* **22**(1), 20–22.

Schmid, V. (2012), 'Solving the dynamic ambulance relocation and dispatching problem using approximate dynamic programming', *European Journal of Operational Research* **219**(3), 611–621.

Smola, A. J. and Schölkopf, B. (2004), 'A tutorial on support vector regression', *Statistics and Computing* **14**(3), 199–222.

Sudtachat, K., Mayorga, M. E. and Mclay, L. A. (2016), 'A nested-compliance table policy for emergency medical service systems under relocation', *Omega* **58**, 154–168.

Swersey, A. J. (1994), 'The deployment of police, fire, and emergency medical units', *Handbooks in Operations Research and Management Science* **6**, 151–200.

Toregas, C., Swain, R., ReVelle, C. and Bergman, L. (1971), 'The location of emergency service facilities', *Operations Research* **19**(6), 1363–1373.

van Barneveld, T., Bhulai, S. and van der Mei, R. D. (2016), 'The effect of ambulance relocations on the performance of ambulance service providers', *European Journal of Operational Research* **252**(1), 257–269.

Zeto, J., Yamada, W. and Collins, G. (2006), Optimizing the emplacement of scarce resources to maximize the expected coverage of a geographically variant demand function, Technical report, US Center for Army Analysis, Ft. Belvoir, VA.

Zhang, Z.-H. and Jiang, H. (2014), 'A robust counterpart approach to the bi-objective emergency medical service design problem', *Applied Mathematical Modelling* **38**(3), 1033–1040.

Vita

Captain Phillip R. Jenkins accomplished his undergraduate studies at Ohio University and earned a Bachelor of Science degree in Mathematics in December 2012. Phillip commissioned into the United States Air Force as an Operations Research Analyst in December 2012.

Captain Jenkins' first assignment was to Air Force Global Strike Command A9 at Barksdale Air Force Base, Louisiana. While stationed at Barksdale, Phillip served as a Long Range Systems Strike Analyst and as the Chief of Force Structure Assessments.

In August 2015, Phillip entered the Air Force Institute of Technology's Graduate School of Engineering and Management at Wright-Patterson AFB, Ohio. He earned his Master's degree in Operations Research in March 2017 as a distinguished graduate. He continued his research as a Doctor of Philosophy student with a specialty area in optimization and a minor in applied statistics. Upon earning his Doctorate degree in Operations Research, Phillip will continue his Air Force career as an Assistant Professor in the Department of Operational Sciences at the Air Force Institute of Technology, Wright-Patterson AFB, Ohio.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	**2. REPORT TYPE**	**3. DATES COVERED** *(From — To)*
13–06–2019	Doctoral Dissertation	March 2017 — June 2019

4. TITLE AND SUBTITLE Strategic Location and Dispatch Management of Assets in a Military Medical Evacuation Enterprise	**5a. CONTRACT NUMBER** **5b. GRANT NUMBER** **5c. PROGRAM ELEMENT NUMBER**
6. AUTHOR(S) Jenkins, Phillip R., Capt, USAF	**5d. PROJECT NUMBER** **5e. TASK NUMBER** **5f. WORK UNIT NUMBER**
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Air Force Institute of Technology Graduate School of Engineering and Management (AFIT/EN) 2950 Hobson Way WPAFB OH 45433-7765	**8. PERFORMING ORGANIZATION REPORT NUMBER** AFIT-ENS-DS-J-037
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) US Army Medical Evacuation Proponency Division Colonel Stephen A. Barnes, Chief, MEPD 4103 Gladiator Street Fort Rucker, AL 36305 stephen.a.barnes.mil@mail.mil	**10. SPONSOR/MONITOR'S ACRONYM(S)** MEPD **11. SPONSOR/MONITOR'S REPORT NUMBER(S)**

12. DISTRIBUTION / AVAILABILITY STATEMENT

DISTRIBUTION STATEMENT A:
APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

This dissertation considers the importance of optimizing deployed military medical evacuation (MEDEVAC) systems and utilizes operations research techniques to develop models that allow military medical planners to analyze different strategies regarding the management of MEDEVAC assets in a deployed environment. For optimization models relating to selected subproblems of the MEDEVAC enterprise, the work herein leverages integer programming, multi-objective optimization, Markov decision processes, approximate dynamic programming, and machine learning, as appropriate, to identify relevant insights for aerial MEDEVAC operations.

15. SUBJECT TERMS

Military medical evacuation (MEDEVAC), integer programming, multi-objective optimization, Markov decision process, approximate dynamic programming, machine learning

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** Dr. Brian J. Lunday, AFIT/ENS
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**			**19b. TELEPHONE NUMBER** *(include area code)*
U	U	U	UU	155	(937) 255-3636, x4624; brain.lunday@afit.edu