The Military Inventory Routing Problem: Utilizing Heuristics Within a Least Squares Temporal Differences Algorithm to Solve a Multiclass Stochastic Inventory Routing Problem with Vehicle Loss

DISSERTATION

Ethan L. Salgado, Captain, USAF

AFIT-ENS-PhD-18-DS-042

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENS-PhD-18-DS-042

THE MILITARY INVENTORY ROUTING PROBLEM: UTILIZING HEURISTICS WITHIN A LEAST SQUARES TEMPORAL DIFFERENCES ALGORITHM TO SOLVE A MULTICLASS STOCHASTIC INVENTORY ROUTING PROBLEM WITH VEHICLE LOSS

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 in Operations Research

Ethan L. Salgado, BS, MS

Captain, USAF

SEPTEMBER 2018

AFIT-ENS-PhD-18-DS-042

THE MILITARY INVENTORY ROUTING PROBLEM: UTILIZING HEURISTICS WITHIN A LEAST SQUARES TEMPORAL DIFFERENCES ALGORITHM TO SOLVE A MULTICLASS STOCHASTIC INVENTORY ROUTING PROBLEM WITH VEHICLE LOSS

Ethan L. Salgado, BS, MS
Captain, USAF

Committee Membership:

Lt Col Matthew J. Robbins, PhD
Co-Chair

Darryl K. Ahner, PhD
Co-Chair

Jeffery D. Weir, PhD
Member

Lt Col Jeremy D. Jordan, PhD
Member

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

AFIT-ENS-PhD-18-DS-042

Abstract

Military commanders currently resupply forward operating bases (FOBs) from a central location within an area of operations mainly via convoy operations in a way that closely resembles vendor managed inventory practices. Commanders must decide when and how much inventory to distribute throughout their area of operations while minimizing soldier risk. Technology currently exists that makes utilizing unmanned cargo aerial vehicles (CUAVs) for resupply an attractive alternative due to the dangers of utilizing convoy operations. Enemy actions in wartime environments pose a significant risk to a CUAV's ability to safely deliver supplies to a FOB. This dissertation develops a Markov decision process (MDP) model to examine this military inventory routing problem (MILIRP).

The first paper examines the structure of the MILIRP by considering a small problem instance and prove value function monotonicity when a sufficient penalty is applied. Moreover, this paper develops a monotone least squares temporal differences (MLSTD) algorithm that exploits this structure and demonstrate its efficacy for approximately solving this problem class. This work compares MLSTD to least squares temporal differences (LSTD), a similar ADP algorithm that does not exploit monotonicity. MLSTD attains a 3.05% optimality gap for a baseline scenario and outperforms LSTD by 31.86% on average in our computational experiments. The second paper expands the problem complexity with additional FOBs. This work generates two new algorithms, Index and Rollout, for the routing portion and implement an LSTD algorithm that utilized these to produce solutions 22% better than myopic generated solutions on average. The third paper greatly increases problem complexity with the addition of supply classes. This research formulates an MDP model to

handle the increased complexity and implement our LSTD-Index and LSTD-Rollout algorithms to solve this larger problem instance and perform 21% better on average than a myopic policy.

Keywords: stochastic inventory routing, approximate dynamic programming, Markov decision process, vendor managed inventory, least squares temporal differences

For my mother. She inspired me to pursue my dreams and gave me the confidence to achieve them.

Acknowledgements

I would like to express my sincere gratitude to my advisor, Dr. J.D. Robbins, for all his help. This would not have been possible without his mentorship, guidance, faith, and support.

Ethan L. Salgado

Table of Contents

Page

List of Figures

List of Tables

THE MILITARY INVENTORY ROUTING PROBLEM: UTILIZING HEURISTICS WITHIN A LEAST SQUARES TEMPORAL DIFFERENCES ALGORITHM TO SOLVE A MULTICLASS STOCHASTIC INVENTORY ROUTING PROBLEM WITH VEHICLE LOSS

I. Introduction

The brigade combat team (BCT) is the primary combined arms force that executes decisive actions for the United States Army. The BCT performs offensive, defensive, stability, and Defense Support of Civil Authorities tasks assigned to it by higher authority [11]. Military logistical planners must consider the timing, routing, and supply configuration of distribution assets when preparing for and executing routine resupply missions (i.e., distribution, replenishment, or sustainment operations) in support of BCT operations. The brigade support battalion (BSB) is the primary organization within the BCT that plans, coordinates, synchronizes, and executes sustainment operations. The BSB operations are accomplished by planning and executing missions within the context of the sustainment warfighting function and by applying the principles of sustainment when executing the support of decisive actions. Sustainment operations typically involve the establishment of a brigade support area (BSA) as the distribution center from which supplies are delivered to company- and platoon-sized units located at forward operating bases (FOBs) geographically dispersed throughout the BCT's area of operations [9]. The objective of sustainment in a wartime environment is to provide sufficient support to enable the BCT to conduct its four primary tasks: movement to contact, attack, exploitation, and pursuit [11]. Logistical planners at the BSB monitor the supply levels of the

FOBs utilizing logistics situation reports and automated sustainment data-gathering systems such as the Battle Command Sustainment Support System and the Force XXI Battle Command Brigade and Below logistical support system [11]. As such, the BSB knows the inventory level at all of the FOBs when making inventory routing decisions. At the beginning of each day, the BSB must decide which FOBs to resupply, how much of supplies to deliver to each FOB, how to combine FOBs into routes, and which routes to assign to each of the available delivery assets.

Distribution of assets that move supplies from the BSA to the FOBs include both ground assets (e.g., medium- and heavy-capacity cargo trucks and tanker trucks) and aerial assets (e.g., the CH-47 Chinook helicopter). While distribution via ground assets account for the majority of tonnage delivered, aerial delivery distribution provides an effective means of conducting distribution operations because it bypasses casualty-inducing enemy activities and reduces the need for route clearance of ground lines of communications (e.g., roads). Moreover, the development of unmanned cargo aerial vehicles (CUAVs) like Lockheed Martin's K-max further reduces troop exposure to potentially life threating enemy actions [19].

Aerial resupply poses its own risks that must be independently considered. North Atlantic Treaty Organization military forces must account for adversaries with the capability and intent to oppose and disrupt allied aerial assets [14]. Threat levels for aerial assets are classified based on the availability, accessibility, and probability of attack. Among the threats, man-portable air-defense systems (MANPADS) are already highly proliferated with an estimated $500,000$ to $750,000$ licensed units worldwide [14]. MANPADS are particularly effective against low or slow aircraft, which makes rotary wing assets particularly vulnerable during take-off and landing.

Military logistical planners face many important challenges when making daily inventory routing decisions in a combat environment. Poorly developed transporta–

tion infrastructure, adverse weather conditions and terrain, enemy threat and actions, and the availability of distribution assets all inhibit successful distribution of supplies from the BSA to the FOBs. Insurgent use of improvised explosive devices (IEDs) has greatly affected truck mobility throughout the operational environment and has been successful in disrupting replenishment procedures [25]. Since current resupply efforts operate mainly via convoys that have become costly and dangerous, this is of great concern. Successful distribution both to and from troop locations must be considered before a resupply decision can be made. Logisticians must decide what supplies (e.g., water, food, fuel, ammunition) should be sent and how much is required. Limiting factors may include distribution asset availability, convoy maintenance requirements, and current threat locations. Constantly evolving socio-political factors may cause a rapid change in current threat areas in the operational environment. Moreover, wartime logistics often do not have a short-term horizon, so logisticians must plan for sustainable resupply over an indefinite horizon.

The United States Department of Defense is interested in the design, development, and utilization of cargo unmanned aerial systems (CUASs) for resupply operations [23]. A CUAS is the collection of all components required to allow the operation of a cargo unmanned aerial vehicle (CUAV). A CUAS includes the operating crew (e.g., maintenance crew and pilot), required software, and vehicles. The United States Army intends to increase the utilization of CUASs as an integral component of inte–grated logistics aerial resupply. As such, examination of inventory routing decisions for CUASs across an austere combat environment is needed.

This dissertation considers a military variant of the stochastic inventory routing problem (MILIRP) in which the BSB of a deployed BCT must decide how many fully loaded CUAVs to dispatch to fulfill the demand requirements of a single com–pany-sized FOB. This research develops a Markov decision process (MDP) model of

the MILIRP. Moreover, it shows the special structural properties of the MILIRP by proving it has a monotone optimal value function when a sufficiently large penalty parameter is specified. When larger, multiple-FOB instances are considered, the high-dimensionality of the state and action space renders classical dynamic programming methods computationally intractable. Thus, this work designs, develops, and tests a monotone approximate dynamic programming (ADP) algorithm to solve the MILIRP. To demonstrate the efficacy of the proposed solution methodology, a notional, representative planning scenario based on an austere combat environment like that of Afghanistan is constructed. Comparing polices determined by the ADP algorithm to those generated by a well known ADP technique (i.e., least squares temporal differencing) and the optimal policy on small problem instances, demonstrates how exploiting structure improves results.

The unique military aspect of the MILIRP warrants further discussion. In contrast to much of the previous work on the inventory routing problem (IRP), this research effort explicitly accounts for the possible destruction of the delivery vehicles. The evolution of threat and weather and their attendant impact on the likelihood of vehicle delivery success is modeled. The lasting and permanent impact of vehicle destruction on the resupply operations over an uncertain horizon must also be modeled. Moreover, in a combat environment the military does not take into account various external costs commonly associated with IRPs. Thus, the MILIRP objective function focuses on total amount of supplies delivered over the life of the system and disregards holding, ordering, and transportation costs.

The remainder of this dissertation is organized as follows. Chapter II presents a review of relevant literature concerning vendor managed inventory practices and the IRP. The literature review also examines several ADP papers to inform the development of the solution methodology. Chapter III provides a description of the MILIRP

and introduces the monotone least squares temporal differences algorithm as an ef–fective ADP solution technique. Chapter IV expands the work done in Chapter III by expanding the number of FOBs considered and introduces the quiz problem heuristic as a means to solve more complex problem instances. Chapter V further increases problem complexity by increasing the number of supply classes considered, greatly expanding the state space.

II. Literature Review

The literature review focuses on two areas of research pertinent to the problem formulation and solution methodology. The first is the inventory routing problem (IRP), which has been widely researched. The second area of interest is approximate dynamic programming (ADP).

2.1 Inventory Routing Problem:

The IRP is an optimization problem wherein inventory is sent from a supplier to customers across a set of locations. The IRP arises from the idea of vendor managed inventory (VMI) replenishment, a centralized approach to inventory management used to reduce overall costs. Utilizing VMI, the IRP is a natural evolution from the vehicle routing problem and is an area of research that has been throughly studied in the operations research field because of the constant need to improve supply chain logistics. The IRP differs from vehicle routing problems because its routing decisions are based on customers' usage rather than customers' orders. The IRP integrates inventory management, vehicle routing, and delivery scheduling decisions. Inventory routing has been a topic of research in the operations research field for over 30 years [7].

VMI replenishment is a business practice wherein the vendor monitors the inventory levels of the customers. The objective of the vendor is the minimization of the sum of inventory and transportation costs over the entire network [18]. VMI is an alternative to traditional inventory management wherein customers keep track of their own inventory and determine when and how much to order from the supplier. The supplier receives orders and uses its vehicles to fill the orders. VMI is an attractive alternative because it is a mutually beneficial relationship between the supplier and

the customer; the supplier reduces transportation costs by deciding when and how much inventory to distribute to each customer, and the customer reduces costs by not allocating resources to monitoring inventory scheduling. There are three main advantages to utilizing VMI practices [17]. First, VMI may lead to reduced production and inventory costs by reducing variation and obtaining a more uniform utilization of resources for both the supplier and the customer. Second, proactive planning used in VMI may reduce transportation costs beyond that of more uniform utilization alone. It may be possible to increase low-cost full truckload shipments and decrease the frequency of high-cost less-than-full truckload shipments. Moreover, it may be possible to use more efficient routes by coordinating the replenishment of customers that are located close to each other. Third, VMI may increase service levels, measured in terms of reliability of product availability.

There are two requirements necessary to obtain the benefits of VMI: the availability of relevant, accurate, and timely data for the decision maker and the ability of the central decision maker to use an increased amount of information to make good decisions [17]. In order to succeed in VMI, an organization must not only have access to relevant information, such as current and past inventory levels at all customers, customer demand behavior, and customer location relative to the vendor and each other, but it must also have the ability to utilize that data in the construction of a relevant and useful distribution policy. This is a very complex task and many failures to implement VMI are a direct result of failing to meet one or both of the above requirements [17]. While a responsible vendor implementing VMI can save both time and money, misuse of VMI business practices can result in lost sales and revenue. Understanding VMI practices builds the knowledge base necessary to understand the IRP.

The IRP falls into a class of problems called NP-hard [7], meaning they are at

least as hard as the hardest non-deterministic polynomial-time problems. The IRP is inherently difficult because a supplier must make three simultaneous decisions: 1) when to serve a given customer, 2) how much to deliver to this customer when it is served, and 3) how to combine customers into vehicle routes [7]. A basic IRP seeks to minimize total inventory-distribution costs while meeting demand of each customer subject to the following constraints: inventory at each customer can never exceed its maximum capacity, inventory levels are not allowed to be negative, the supplier's vehicles can perform at most one route per time period with each starting and ending at the supplier, and vehicle capacities cannot be exceeded.

Coelho *et al.* [7] identify problem features to describe IRPs. These features include: time horizon, structure, routing, inventory policy, inventory decisions, fleet composition, and fleet size. Within the IRP framework time horizon is a problem dependent feature that can either be finite or infinite. With respect to structure, the number of suppliers and customers can vary and includes the following categories: one-to-one when there is only one supplier and one customer, one-to-many when there are many customers, or more rarely, many-to-many. Routing includes the following categories: direct when there is only one customer per route, multiple when there are several customers in the same route, or continuous when there is no central depot (i.e., maritime applications). Direct delivery involves the vehicle moving directly from the supplier to the customer and returning to the vendor immediately after delivery. Direct delivery greatly simplifies the IRP by removing the optimization of the routing portion of the problem. Direct delivery is appropriate for this application of the MILIRP due to the low carrying capacity of currently fielded CUAVs [19]. The two most common inventory polices are the maximum level or order-up-to level policies. The maximum level policy allows flexibility in deciding the amount to refill whereas the order-up-to level policy replenishes customers to a particular inventory level each

time the customer is visited. Inventory decisions can be modeled as lost sales when excess demand becomes lost revenue or as back-orders when demand can be filled at a later date. Fleet composition can either be homogeneous or heterogeneous, and fleet size can be single, limited, or unconstrained.

Coelho *et al.* [7] and Toth & Vigo [33] give a basic introduction to the stochastic variant of the basic IRP. In the stochastic inventory routing problem (SIRP), the sup–plier knows the customer demand only in a probabilistic sense. Demand stochasticity means shortages may occur. In order to discourage shortages, a penalty function is imposed whenever a customer runs out of stock and is usually modeled as unsatisfied demand. With no backlogging, unsatisfied demand is considered lost. There are sev–eral solution methods employed to solve the IRP, which include, but are not limited to, heuristic algorithms, link optimization, simulation, and dynamic programming.

Campbell *et al.* [6] and Minkoff [24] formulate their SIRP in a similar fashion. They both model the use of an unconstrained fleet (in terms of size) to meet demand across a network and allow for multiple routing. Campbell *et al.* [6] do not present a specific analysis of their SIRP formulation; instead, they develop challenging IRP test instances. Minkoff [24] applies a heuristic approach to solving the SIRP based on a decomposition of the problem by customer. The solutions to the customer subprob–lems generate the penalty functions that are applied within their master dispatching problem.

Adelman [1] and Kleywegt *et al.* [18] provide very similar SIRP formulations. They both formulate and solve infinite horizon problems with a one-to-many structure. While their solution methodologies differ, they both focus on multiple routing and maximum level inventory policies. They both employ homogeneous fleet composition without backlogging and with a fixed, limited fleet. Adelman [1] differs from Kleywegt *et al.* [18] in that he uses linear programming techniques to obtain his solution whereas

Kleywegt *et al.* [18] use ADP.

Two papers deserve more in-depth discussion because they both greatly informed this research and closely resemble this work in terms of methodology. Kleywegt *et al.* [17] model an IRP with the following characteristics: direct delivery, limited fleet size, stochastic demand, and deterministic vehicle supply. Similarly, Kleywegt *et al.* [18] model an IRP with multiple routing, limited fleet size, stochastic demand, and deterministic vehicle supply. This research differs from both papers in that the distinct military nature of this formulation yields a stochastic vehicle supply. The stochastic nature of the vehicle supply is discussed in more detail in Chapter 3.1.

Kleywegt *et al.* [17] and Kleywegt *et al.* [18] both employ ADP as a solution technique. Because of the complexity inherent in IRPs, ADP is an excellent solu–tion technique to produce high-quality inventory routing policies. Kleywegt *et al.* [17] employ an approximate policy iteration (API) algorithm with a parametric value function approximation. They construct a set of basis functions to create a linear approximation architecture around the pre-decision state. Kleywegt *et al.* [18] apply the same ADP solution technique for the first part of their optimization problem be–fore considering multiple delivery and then use a heuristic search method to determine additional delivery opportunities afterwards, if possible. Several differences exist that distinguish this dissertation's solution approach from theirs. First, this research con–structs a set of basis functions to create a linear approximation architecture around the post-decision state not the pre-decision state. Second, an ADP algorithm that enforces monotonicity within the value function approximation is employed. Third, the stochastic nature of the delivery and the possible loss of the delivery vehicles in the limited fleet are distinguishing features not present in other IRP research endeavors in the current literature.

This dissertation is not the first research effort on the MILIRP. The MILIRP was

initially formulated by McCormack [20]. McKenna [22] expanded this research by extending the number of forward operating bases while maintaining direct delivery. Salgado [31] adds stochastic demand to the direct delivery MILIRP. This research builds off of these previous works through structural analysis of the unique problem features of the MILIRP, creation of new algorithms better suited to solve the MILIRP, and relaxation of the direct delivery constant.

2.2 Approximate Dynamic Programming

Inventory routing decisions in a combat environment involve sequential decision making under uncertain conditions. Because of enemy threats, the routing of a cargo unmanned aerial vehicle (CUAV) to replenish supplies has an uncertain outcome. The loss of a CUAV impacts the ability of the Brigade Support Battalion (BSB) to replenish supplies in the future. Thus, the safety of the CUAV in the formulation must be accounted for. This dissertation formulates the MILIRP as a Markov decision process (MDP). However, due to the high dimensionality of this problem when practical instances are considered, it is unable to be solved exactly using classical dynamic programming solution techniques. To overcome the curse of dimensionality, an approximate dynamic programming (ADP) methodology is implemented in order to solve the MILIRP. ADP is being concurrently developed by multiple different communities to include engineering controls, computer science (artificial intelligence), and operations research. For a more detailed introduction to ADP from an operations research perspective, the reader is referred to Powell [26, 27, 28]. For a different ADP outlook, the reader is referred to Bertsekas & Tsitsiklis [4] (engineering control theory) or Sutton & Barto [32] (artificial intelligence).

The API algorithmic strategy involves utilizing the post-decision state to construct a linear architecture based on an appropriate set of basis functions while maintaining

monotonicity in the value function. Van Roy *et al.* [34] first introduced post-decision state approximation as a way to modify Bellman's equation to obtain an equivalent, deterministic expression. Using the post-decision state is computationally helpful be–cause it addresses the outcome state portion of the curse of dimensionality and allows one to average the optimal value rather than estimate the expected value as a function of the decision. API consists of two basic steps: policy improvement and policy eval–uation. Within the policy improvement step of the API algorithm, the value function approximation is updated for a fixed policy using least squares temporal differencing (LSTD). Bradtke & Barto [5] introduced LSTD as a computationally efficient method for estimating the adjustable parameters when using a linear architecture with fixed basis functions to approximate the value function for a fixed policy. LSTD updates its estimate of the expected contribution and projects this over the infinite horizon [27]. A variant of the LSTD algorithm similar to Rettke *et al.* [30], Davis *et al.* [8], and McKenna *et al.* [21] is implemented. This dissertations' variant distinguishes itself by utilizing a post-decision state value function approximation and a monotonicity projection operator to maintain value function monotonicity. The development of the monotonicity projection operator was greatly informed by Jiang & Powell [16], who examine a finite horizon problem and construct an ADP approach that attains high-quality solutions within a relatively small number of iterations.

2.3 Rollout Algorithms:

Rollout algorithms produce heuristic solutions that, when implemented efficiently, have been shown to yield considerable savings in computation over optimal algorithms on stochastic control problems with combinatorial decision spaces like the *quiz problem* [3]. The military inventory routing problem (MILIRP) is a stochastic control problem with a combinatorial decision space, and as such, studying the *quiz problem*

variant referred to as the *stochastic quiz problem* does produce a useful heuristic for solving the MILIRP as shown in Chapter 4.3. The unique nature of the MILIRP allowing for vehicle destruction is analogous to a stopping rule set to a quiz taker answering a question incorrectly on a sequential exam given individual probabilities of success. Bertsekas & Castanon [3] discuss how to use rollout algorithms for the *stochastic quiz problem* where, as in the MILIRP, there is no optimal open-loop policy (i.e., an optimal order for the questions (FOBs) does depend on the random outcome of the earlier questions). These problems can only be solved exactly with dynamic programming, but their optimal solutions are prohibitively difficult to determine be–cause the states over which dynamic programing must be executed are subsets of questions, and the number of their subsets increases exponentially with the num–ber of questions [3]. Since the *quiz problem heuristic* only applies to deterministic quiz problems and its variants, it cannot be directly applied for the MILIRP without reservation. However, as with all heuristic methods, it may result in computational savings that provide value to solving the MILIRP.

III. A Stochastic Inventory Routing Problem with Vehicle Loss

The first paper examines the structure of the MILIRP by considering small prob–lem instances and prove value function monotonicity when a sufficient penalty is applied. Moreover, this research effort develops a monotone least squares temporal differencing (MLSTD) algorithm that exploits this structure and show its advantages on the MILIRP. This section first provides a brief problem description and discussion concerning MDP methodology.

3.1 Problem Description

A basic understanding of the U.S. Army replenishment structure is central to understanding the military inventory routing problem (MILIRP). The brigade combat team (BCT) is the highest echelon organization able to act independently in regional combat operations. The BCT is responsible for a number of forward operating bases (FOBs) within its area of operations. The interaction between BCT and FOB parallels the supplier-to-customer relationship seen in vendor managed inventory practices. Within the BCT, a sub-organization called a brigade support battalion (BSB) is responsible for replenishment of FOBs in the BCT's area of operations.

The BSB plans, coordinates, synchronizes, and executes replenishment operations in support of BCT operations [12]. The BSB is the organization within the BCT that establishes and operates the brigade support area (BSA), a central location utilized to resupply its customers (i.e., FOBs) at locations of varying distances. The BSB is responsible for the periodic resupply of the BCT's subordinate units, which closely mirrors vendor managed inventory (VMI) practices used in the civilian sector. To accomplish its responsibility, the BSB is kept informed of inventory levels at the FOBs through regular reporting and automated data systems. VMI practices allow

the BSB to decide when, where, and how much supplies to send to FOBs. The routing and resupply operation of the FOB can be formulated as a variant of the inventory routing problem (IRP) due to these similarities.

Replenishment during combat operations includes difficult, deliberate, and time-sen–sitive operations conducted to replenish forward companies with essential supplies to sustain the pace of operations [12]. The U.S. Army employs trucks, manned air assets, and now cargo unmanned aerial vehicles (CUAVs) to perform replenishment opera–tions. Many operational issues must be considered when utilizing these distribution assets.

Operating the cargo unmanned aerial system (CUAS) presents challenges that must be addressed. The CUAS is a complex system because of the many influenc–ing factors required for successful operation: remote pilot (operator) (for emergency and combat purposes), maintenance requirements, maintenance crew, aircraft fuel, required software, and CUAV. The CUAV is inherently a more complicated vehicle than typically observed in applications of the IRP. In this paper, we condense all influencing factors for CUAV operation into two categories: CUAV and crew. We refer to 'crew' as all other factors required for CUAS operation other than the CUAV itself.

Due to the possibility of vehicle destruction, the MILIRP necessarily takes into account the stochasticity of supply allocation decisions. CUAV routing decisions are influenced by the current threat conditions because a destroyed CUAV cannot be replaced; losing CUAVs has a permanent and lasting impact on the ability of the BSB to deliver supplies.

Lack of road infrastructure within the area of operations and enemy attacks make resupply via ground transport inherently dangerous. General Dynamics reports that improvised explosive devices caused 18% of all deployed fatalities between November

2002 and March 2009, all occurring during sustainment operations [13]. If CUAV re–supply is unable to meet supply requirements, FOBs must be supplied mostly through ground convoy operations. Due to the human capital expenditure risk necessary to resupply FOBs via ground convoy, we impose a penalty on the system if the CUAS is unable to fulfill FOB demand requirements (i.e., a FOB's inventory level falls below a specified safety-stock level).

3.2 Methodology

This section describes the Markov decision process (MDP) model formulation of the military inventory routing problem (MILIRP). The objective of the MILIRP with stochastic demand is to determine the optimal resupply of a single, large forward op–erating base (FOB) via inventory routing decisions in order to maintain inventory. The reward function maintains increasing monotonicity with respect to supplies de–livered to the FOB until it reaches the FOB's maximum holding capacity, after which additional supplies delivered yield no reward. We assume the inventory level at the FOB is known at the start of each period and that supply demand has a known historical average with some variability, modeled as an independent and identically distributed error term. Inherent in this formulation is the assumption that no other external event (e.g., enemy action, fire, expiration of supplies) other than demand causes a loss of inventory.

A brigade combat team (BCT) is responsible for the FOBs within its area of op–erations. In this analysis, we only consider the resupply of a single, company-sized FOB. The BCT contains a brigade support battalion (BSB) that manages resupply efforts for the FOB. The BSB distributes supplies to the FOB utilizing V identical cargo unmanned aerial vehicles (CUAVs). Each CUAV has an identical load capacity of V^{cap} tons. The FOB requires $\hat{D}_t$ tons of supplies per time period t, a stochas–

tic demand with a mean demand $\bar{d}$ and an independent and identically distributed exogenous error term $\hat{\epsilon}$. The FOB also has a finite maximum holding quantity R^{max}.

Given an austere combat environment, there is potential for delivery failure due to extrinsic uncontrollable factors (e.g., enemy action, mechanical failure, extreme weather conditions). The probability of a CUAV being destroyed depends on the current threat conditions. A set of M threat maps models the periodic changes in risk throughout the BCT's area of operations. Under threat map $m = 1, 2, ..., M$, the parameter ψ_m denotes the probability of a successful one-way trip from (to) the brigade support area (BSA) to (from) the FOB. A CUAV may be destroyed either on its way to a FOB or after delivering supplies on the return route back to the depot at the BSA.

We proceed by describing the MDP model formulation of the MILIRP. With respect to a conventional inventory routing formulation, CUAVs are vehicles, the FOB is a customer, and the centralized BSB is the supplier. Table 1 located at the end of this section provides a summary of the notation.

The MILIRP is formulated as an infinite horizon Markov decision problem wherein at each decision epoch $t \in \mathcal{T} = \{1, 2, ...\}$ an inventory routing decision is made. During each time period a CUAV refuels, resupplies, receives maintenance, travels to the FOB, unloads, and returns to the BSB. It is assumed that the FOB is within the CUAV's range when fully loaded and that this route is serviceable in one time period. Current CUAV limitations validate this assumption [19].

The state space includes three components: the inventory level at the FOB, the number of operational CUAVs, and the threat map index number. The inventory at the FOB is defined as r_t, where $r_t \in (0, R^{max})$ is the number of tons of supplies at the FOB at time t. Moreover, R^{max} is the maximum inventory capacity for the FOB, and $R^{min} \in (0, R^{max})$ is the minimum threshold inventory level that must be

exceeded (i.e., the safety stock level). If $r_t \leq R^{min}$ then resupply via convoy ground lines of communication (GLOC) is required. The number of operational CUAVs able to perform resupply operations at time t is defined as v_t. The threat map index number at time t is defined as $m_t \in \{1, 2, ..., M\}$. The threat map impacts the flight risk associated with successfully completing sorties between the FOB and the BSA. The threat map information m_t is available at time t. The threat map information m_{t+1} available at time $t+1$ is conditioned on m_t and is unknown at time t. Utilizing these components, we define $s_t = \big(r_t, v_t, m_t\big) \in \mathcal{S}$ as the state of the system at time t, where $\mathcal{S}$ is the set of all possible states.

We let $\mathcal{X}(s_t)$ be the set of all feasible actions when the system is in state s_t. Let $x_t = (x_t^d, x_t^{GLOC}) \in \mathcal{X}(s_t)$ denote an inventory routing decision wherein $x_t^d \in \mathbb{N}^0$ denotes the number of fully loaded CUAVs dispatched to resupply the FOB and $x_t^{GLOC} \in \{0, 1\}$ denotes whether a ground convoy is dispatched to resupply the FOB, which results in its inventory level increasing to capacity. Only CUAV resupply is available if the inventory level is greater than the safety stock threshold (i.e., $r_t > R^{min}$). Only GLOC resupply is available if the inventory level is less than or equal to the safety stock threshold (i.e., $r_t \leq R^{min}$). Two constraints impact the CUAV routing decision: first, the number of CUAVs deployed cannot exceed the number of operational CUAVs (i.e., $x_t^d \leq v_t$); second, the number of CUAVs deployed cannot exceed the number of crews available (i.e., $x_t^d \leq V^{crew}$). We assume that each CUAV carries a maximum capacity load of V^{cap}. The policy (i.e., decision function) $X^\pi(s_t)$ returns a decision $x_t \in \mathcal{X}(s_t)$ as a function of the system state $s_t \in \mathcal{S}$. After a routing decision is made, delivery is performed within one time period.

Transition probabilities are defined for each dimension of the state space to in–clude the inventory level at the FOB, number of remaining CUAVs, and threat map. Inventory transitions are based on the routing decision x_t and the current state of the

system s_t. When CUAVs are routed to the FOB there are three possible outcomes (governed by a trinomial distribution): first, a CUAV may successfully travel to and from the FOB; second, a CUAV may successfully deliver its supplies and be destroyed upon returning to the BSA; third, a CUAV may be destroyed before successfully de–livering its supplies. Let ψ_m^2, $\psi_m(1-\psi_m)$, and $(1-\psi_m)$ respectively denote the probabilities of a successful two-way delivery (SS), successful one-way delivery (SF), and failure (F) for a single CUAV routed to resupply the FOB during the threat conditions of map $m = 1, 2, ..., M$. Since we are interested in a particular outcome of a routing decision, we proceed by defining the binomial marginal distributions for each outcome type (i.e., SS, SF, F). With the assumption that each outcome of a resupply mission to a FOB is independent of other missions and recalling that x_t in–cludes the decision to route x_t^d CUAVs to the FOB (each carrying a full supply load), we let $\hat{Z}_{t+1}^{SS}(\psi_m^2, x_t^d)$ denote the binomial random variable with parameters ψ_m^2 and x_t^d that indicates the number of successful two-way CUAV deliveries to the FOB during time interval $[t, t+1)$ on map m. Let $\hat{Z}_{t+1}^{SF}(\psi_m(1-\psi_m), x_t^d)$ and $\hat{Z}_{t+1}^{F}((1-\psi_m), x_t^d)$ be similarly defined. For compactness, we refer to the set of random variables that indicate resupply mission outcomes as follows:

$$\hat{Z}_{t+1} = \left(\hat{Z}_{t+1}^{SS}, \hat{Z}_{t+1}^{SF}, \hat{Z}_{t+1}^{F}\right). \tag{1}$$

The inventory level at the FOB is limited by the maximum holding quantity R^{max}. Moreover, if the FOB supply level is less than or equal to a safety stock threshold, R^{min}, the FOB must be fully resupplied via ground convoy. Equation 2 is the inventory transition function for the FOB.

$$r_{t+1} = \begin{cases} R^{max} & \text{if } x_t^{GLOC} = 1 \\ \min\left(r_t + V^{cap}(\hat{Z}_{t+1}^{SS} + \hat{Z}_{t+1}^{SF}) - \hat{D}_{t+1}, R^{max}\right) & \text{if } x_t^d > 0 \\ r_t - \hat{D}_{t+1} & \text{otherwise.} \end{cases} \tag{2}$$

In the first case, convoy resupply is selected, and the FOB is resupplied to capacity. In the second and third cases, the FOB inventory level changes according to supplies received and realized demand. The minimization in the second case enforces the FOB capacity constraint.

CUAV transition is contingent on the number of CUAVs that fail to return to the BSA after attempting to travel to the FOB. The number of CUAVs transition according to Equation 3.

$$v_{t+1} = v_t - (\hat{Z}^{SF}_{t+1} + \hat{Z}^{F}_{t+1}) \tag{3}$$

The map transition function is a representation of the uncontrolled stochastic aspect of the combat environment. The set of all maps captures the threat level of the operational environment. The map transitions are representative of the changing environment. For relatively static combat conditions, the map transition probability would be relatively low. More dynamic combat environments yield a relatively higher map transition probability. The BCT intelligence teams gather information on threat conditions based on information such as enemy action, season, historical trends, and weather.

The contribution function rewards the system based on the amount of supplies delivered by CUAV to the FOB. The amount of supplies delivered is bounded above by the maximum inventory quantity at the FOB, constraining any excess supplies delivered from affecting the system behavior. An immediate penalty is applied if the FOB's inventory level is less than or equal to the safety stock threshold R^{min} due to the human risk associated with ground convoy resupply. The below-threshold penalty function for the FOB

$$\tau(r) = \begin{cases} 0 & \text{if } r > R^{min}, \\ \bar{\tau} & \text{if } r \leq R^{min} \end{cases} \tag{4}$$

allows the application of a penalty that can capture the difficulty of resupplying the

FOB via ground convoy. We present our contribution function in Equation 5.

$$C(s_t, x_t) = \mathbb{E}\Big\{ \min\Big(R^{max} - r_t + \hat{D}_{t+1}, V^{cap} \cdot (\hat{Z}^{SS}_{t+1} + \hat{Z}^{SF}_{t+1})\Big) - \tau(r_t)\Big| s_t, x_t \Big\} \quad (5)$$

The single-period contribution (i.e., reward) is determined by the amount of supplies successfully delivered to the FOB. However, the system is not rewarded for excess supplies (i.e., the FOB cannot take delivery of supplies in excess of its capacity).

The objective of this MDP is to maximize the expected total discounted reward over an infinite horizon. By definition, the transitions are Markovian. All decisions made at time t depend only on the current state of the system. To obtain the policy that maximizes the expected total discounted reward, Bellman's optimality equation is solved:

$$J(s_t) = \max_{x \in \mathcal{X}(s_t)} \Big(C(s_t, x) + \lambda \mathbb{E}\{J(s_{t+1})|s_t, x\}\Big). \quad (6)$$

The value of being in state s_t results from choosing the action that maximizes the sum of the expected immediate contribution and the discounted expected value of the state of the system at time $t+1$. The parameter $\lambda \in [0, 1)$ denotes the discount factor. Using this MDP formulation, an approximate dynamic programming algorithm can be developed to obtain policies for resupplying the FOB via CUAVs.

3.3 Structural Properties

We examine the special structure of the MILIRP to inform our solution methodology. We define the partial order over the state space $\mathcal{S} \ni s_t = (r_t, v_t, m_t)$ as

$$s \succeq \tilde{s} \iff r \geq \tilde{r}, v = \tilde{v}, m = \tilde{m}.$$

Table 1. Table of Notation

C	=	Contribution Function
d	=	daily FOB demand
F	=	failure to deliver supplies
I	=	indicator variable
J	=	total expected reward (cost-to-go) function
k	=	policy evaluation loop counter
K	=	number of policy evaluation loops
M	=	number of threat maps
n	=	policy improvement loop counter
N	=	number of policy improvement loops
r	=	supplies on hand
R^{min}	=	supply threshold
R^{max}	=	FOB holding capacity
s	=	state of system
SF	=	one-way successful trip
SS	=	two-way successful trip
t	=	time epoch
v	=	current number of CUAVs
V	=	number of inital CUAVs
V^{cap}	=	CUAV holding capacity
V^{crew}	=	number of crews
w	=	exogenous information process
x	=	actions
Z	=	set of random variables corresponding to the number of possible SS, SF, and F events
$\mathcal{F}$	=	set of basis function features
$\mathcal{S}$	=	state space
$\mathcal{T}$	=	set of time epochs
$\mathcal{X}$	=	action space
α	=	stepsize
β	=	probability of remaining in the high threat map
θ	=	vector of weights
λ	=	discount factor
π	=	policy
τ	=	penalty cost
ψ	=	one-way probability a CUAV successfully reaches its destination
ϕ	=	basis function
Φ	=	matrix of fixed basis functions
Ω	=	probability of remaining in the low threat map

We prove herein that the optimal value function J^* is order preserving over the state space (i.e., $s \succeq \tilde{s} \implies J^*(s) \geq J^*(\tilde{s})$). Such a result provides justification for the development of our monotone least squares temporal difference (MLSTD) approximate dynamic programing (ADP) algorithm that exploits the monotonicity of the value function to improve the quality of solutions attained.

Theorem 1. The optimal value function $J^*(s)$ is nondecreasing in r_t for fixed v_t and m_t for $t \in \mathcal{T}$ when $\bar{\tau} \geq R^{max}$.

Proof. The claim is shown by demonstrating that the following three conditions [29] are satisfied.

1. Using $s_t = (r_t, v_t, m_t)$ and the partial ordering $s_t \succeq \tilde{s}_t$, consider $r_t \geq \tilde{r}_t$. It suffices to show that

$$C((r_t, v_t, m_t), x_t) \geq C((\tilde{r}_t, v_t, m_t), x_t) \; \forall \; s_t, \tilde{s}_t \in \mathcal{S} \text{ and } x_t \in \mathcal{X}'$$

First, consider the trivial case where $r = \tilde{r}$. It follows that $C((r_t, v_t, m), x_t) = C((\tilde{r}_t, v_t, m), x_t)$ holds true. Now consider $r_t > \tilde{r}_t$. Using the expected immediate contribution function

$$C(s_t, x_t) = \mathbb{E}\Big\{ \big[\min \big(R^{max} - r_t + \hat{D}_{t+1}, V^{cap}(\hat{Z}^{SS}_{t+1} + \hat{Z}^{SF}_{t+1}) \big) \big] - \tau(r_t) | s_t, x_t \Big\} =$$

$$\sum_{d=0}^{\infty} \sum_{z^{SS}=0}^{x_t} \sum_{z^{SF}=0}^{x_t} [\mathbb{P}\Big(\hat{D}_{t+1} = d, \hat{Z}^{SS}_{t+1} = z^{SS}, \hat{Z}^{SF}_{t+1} = z^{SF}\Big) \Big(\min \big(R^{max} - r_t + d, V^{cap}(z^{SS} + z^{SF}) \big) - \tau(r_t) \Big)],$$

it suffices to show that

$$C((r_t, v_t, m_t), x_t) \geq C((\tilde{r}_t, v_t, m_t), x_t) \iff$$

$$\sum_{d=0}^{\infty} \sum_{z^{SS}=0}^{x_t} \sum_{z^{SF}=0}^{x_t} [\mathbb{P}\Big(\hat{D}_{t+1} = d, \hat{Z}^{SS}_{t+1} = z^{SS}, \hat{Z}^{SF}_{t+1} = z^{SF}\Big) \Big(\min \big(R^{max} - r_t + d, V^{cap}(z^{SS} + z^{SF}) \big) - \tau(r_t) \Big)] \geq$$

$$\sum_{d=0}^{\infty}\sum_{z^{SS}=0}^{x_t}\sum_{z^{SF}=0}^{x_t}[\mathbb{P}\big(\hat{D}_{t+1}=d,\hat{Z}_{t+1}^{SS}=z^{SS},\hat{Z}_{t+1}^{SF}=z^{SF}\big)\big(\min\left(R^{max}-\tilde{r}_t+d,V^{cap}(z^{SS}+z^{SF})\right)-\tau(\tilde{r}_t)\big)]\iff$$

$$\sum_{d=0}^{\infty}\sum_{z^{SS}=0}^{x_t}\sum_{z^{SF}=0}^{x_t}[\mathbb{P}\big(\hat{D}_{t+1}=d,\hat{Z}_{t+1}^{SS}=z^{SS},\hat{Z}_{t+1}^{SF}=z^{SF}\big)\big(\tau(\tilde{r}_t)-\tau(r_t)\big)]\geq$$

$$\sum_{d=0}^{\infty}\sum_{z^{SS}=0}^{x_t}\sum_{z^{SF}=0}^{x_t}[\mathbb{P}\big(\hat{D}_{t+1}=d,\hat{Z}_{t+1}^{SS}=z^{SS},\hat{Z}_{t+1}^{SF}=z^{SF}\big)\big(\min\left(R^{max}-\tilde{r}_t+d,V^{cap}(z^{SS}+z^{SF})\right)-\min\left(R^{max}-r_t+d,V^{cap}(z^{SS}+z^{SF})\right)\big)]\iff$$

$$\tau(\tilde{r}_t)-\tau(r_t)\geq$$

$$\sum_{d=0}^{\infty}\sum_{z^{SS}=0}^{x_t}\sum_{z^{SF}=0}^{x_t}[\mathbb{P}\big(\hat{D}_{t+1}=d,\hat{Z}_{t+1}^{SS}=z^{SS},\hat{Z}_{t+1}^{SF}=z^{SF}\big)\big(\min\left(R^{max}-\tilde{r}_t+d,V^{cap}(z^{SS}+z^{SF})\right)-\big(\min\left(R^{max}-r_t+d,V^{cap}(z^{SS}+z^{SF})\right)$$

Consider the extreme values of this final expression. Letting $r_t = R^{max}$ and $\tilde{r}_t = 0$, we have

$$\tau(0) - \tau(R^{max}) \geq R^{max} + d - d = R^{max}.$$

Since $\tau(0) = \bar{\tau}$ and $\tau(R^{max}) = 0$, we have that, in the most extreme case, $C(s_t, x_t) \geq C(\tilde{s}_t, x_t) \iff \bar{\tau} \geq R^{max}$, which is a valid statement.

2. $q(k|s_t, x_t)$ is nondecreasing in r_t for fixed v_t and m_t for all $k \in \mathcal{R} = [0, R^{max}]$ and $x_t \in \mathcal{X}'$.

Using $s_t = (r_t, v_t, m_t)$ and the partial ordering $s_t \succeq \tilde{s}_t$, consider $r_t \geq \tilde{r}_t$. For the trivial case where $r = \tilde{r}$, $q(k|(r_t, v_t, m_t), x_t) = q(k|(\tilde{r}_t, v_t, m_t), x_t)$. For $r > \tilde{r}$ it suffices to show

$$q(k|(r_t, v_t, m_t), x_t) \succeq q(k|(\tilde{r}_t, v_t, m_t), x_t) \iff$$

$$\sum_{j \in \{\mathcal{S}|j \geq k\}} \mathbb{P}(j|r_t, x_t) \geq \sum_{j \in \{\mathcal{S}|j \geq k\}} \mathbb{P}(j|\tilde{r}_t, x_t)$$

where $\sum_{j \in \{\mathcal{S}|j \geq k\}} \mathbb{P}(j|r_t, x_t)$ represents the probability that the inventory level at time $t+1$ is greater than or equal to k. This expression must hold for five possible cases of inventory transitions. Each case is analyzed for fixed action $(x_t^d = 0, x_t^{GLOC} = 1)$ and $(x_t^d, x_t^{GLOC} = 0)$.

Case 1. $k > r_t + V^{cap}(Z^{SS}_{t+1} + Z^{SF}_{t+1}) - \hat{D}_{t+1}$

$$\sum_{j \in \{\mathcal{S} | j \geq k, k > r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (x^d_t, 0) \geq \sum_{j \in \{\mathcal{S} | j \geq k, k > r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | \tilde{r}_t, (x^d_t, 0)) \iff$$

$$0 \geq 0$$

$$\sum_{j \in \{\mathcal{S} | j \geq k, k > r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (0, x^{GLOC}_t = 1)) \geq \sum_{j \in \{\mathcal{S} | j \geq k, k > r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | \tilde{r}_t, (0, x^{GLOC}_t = 1)) \iff$$

$$1 \geq 1$$

Case 2. $\tilde{r}_t + V^{cap}(Z^{SS} + Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS} + Z^{SF}) - \hat{D}_{t+1}$

$$\sum_{j \in \{\mathcal{S} | j \geq k, \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (x^d_t, 0)) \geq$$

$$\sum_{j \in \{\mathcal{S} | j \geq k, \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | \tilde{r}_t, (x^d_t, 0)) \iff$$

$$\sum_{j \in \{\mathcal{S} | j \geq k, \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (x^d_t, 0)) \geq 0$$

$$\sum_{j \in \{\mathcal{S} | j \geq k, \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (0, x^{GLOC}_t = 1)) \geq$$

$$\sum_{j \in \{\mathcal{S} | j \geq k, \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1} < k < r_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | \tilde{r}_t, (0, x^{GLOC}_t = 1)) \iff$$

$$1 \geq 1$$

Case 3. $k < \tilde{r}_t + V^{cap}(Z^{SS} + Z^{SF}) - \hat{D}_{t+1}$

$$\sum_{j \in \{\mathcal{S} | j \geq k, k < \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | r_t, (x^d_t, 0)) \geq \sum_{j \in \{\mathcal{S} | j \geq k, k < \tilde{r}_t + V^{cap}(Z^{SS}+Z^{SF}) - \hat{D}_{t+1}\}} \mathbb{P}(j | \tilde{r}_t, (x^d_t, 0)) \iff$$

$$1 \geq 1$$

$$\sum_{j\in\{\mathcal{S}|j\geq k,k<\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|r_t,(0,x_t^{GLOC}=1))\geq\sum_{j\in\{\mathcal{S}|j\geq k,k<\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|\tilde{r}_t,(0,x_t^{GLOC}=1))\iff$$

$$1\geq 1$$

Case 4. $k=r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}$

$$\sum_{j\in\{\mathcal{S}|j\geq k,k=r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|r_t,(x_t^d,0))\geq\sum_{j\in\{\mathcal{S}|j\geq k,k=r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|\tilde{r}_t,(x_t^d,0))\iff$$

$$\mathbb{P}(j\geq r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|r_t,(x_t^d,0))\geq\mathbb{P}(j\geq r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|\tilde{r}_t,(x_t^d,0))\iff$$

$$1\geq 0$$

$$\sum_{j\in\{\mathcal{S}|j\geq k,k=r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|r_t,(0,x_t^{GLOC}=1))\geq\sum_{j\in\{\mathcal{S}|j\geq k,k=r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|\tilde{r}_t,(0,x_t^{GLOC}=1))\iff$$

$$\mathbb{P}(j\geq r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|r_t,(0,x_t^{GLOC}=1))\geq\mathbb{P}(j\geq r_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|\tilde{r}_t,(0,x_t^{GLOC}=1))\iff$$

$$1\geq 1$$

Case 5. $k=\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}$

$$\sum_{j\in\{\mathcal{S}|j\geq k,k=\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|r_t,(x_t^d,0))\geq\sum_{j\in\{\mathcal{S}|j\geq k,k=\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|\tilde{r}_t,(x_t^d,0))\iff$$

$$\mathbb{P}(j\geq \tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|r_t,(x_t^d,0))\geq\mathbb{P}(j\geq \tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|\tilde{r}_t,(x_t^d,0))\iff$$

$$1\geq 1$$

$$\sum_{j\in\{\mathcal{S}|j\geq k,k=\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|r_t,(0,x_t^{GLOC}=1))\geq\sum_{j\in\{\mathcal{S}|j\geq k,k=\tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}\}}\mathbb{P}(j|\tilde{r}_t,(0,x_t^{GLOC}=1))\iff$$

$$\mathbb{P}(j\geq \tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|r_t,(0,x_t^{GLOC}=1))\geq\mathbb{P}(j\geq \tilde{r}_t+V^{cap}(Z^{SS}+Z^{SF})-\hat{D}_{t+1}|\tilde{r}_t,(0,x_t^{GLOC}=1))\iff$$

$1 \geq 1$

3. $C_T(s_T)$ is nondecreasing in r_t for fixed v_t and m_t for an arbitrarily large $T \in \mathcal{T}$.

Since the contribution function only rewards supplies delivered and not supplies on hand, there is no terminal contribution and $C_T(s_T) = 0, \forall\ s \in \mathcal{S}$. $\square$

3.4 Solution Methodology

This section introduces a monotone least squares temporal differences (MLSTD) algorithm that exploits the special structure of the military inventory routing problem (MILIRP) to more effectively solve this variant of the stochastic inventory routing problem (IRP).

ADP Formulation.

Our monotone approximate dynamic programming (ADP) algorithm utilizes an approximate policy iteration (API) framework with a least squares temporal differences (LSTD) value function approximation scheme similar to work by Rettke *et al.* [30] and Davis *et al.* [8]. API mirrors the exact policy iteration algorithm closely. Instead of using the one-step transition matrix that is difficult to utilize when solving problems with high dimensionality, our API algorithm approximates and updates the value function after simulating system trajectories. We utilize the post-decision state, which is the state of the system immediately after a decision is made but before the exogenous information processes are realized. This allows the expectation to be moved outside of the maximization operator, altering our value function to the form

$$J^x(s_t^x) = \mathbb{E}\Big\{ \max_{x \in \mathcal{X}(s_{t+1})} \big(C(s_{t+1}, x) + \gamma J^x(s_{t+1}^x)\big) | s_t^x \Big\}.$$

LSTD utilizes a set of basis functions that capture relevant information in the system thus reducing the dimensionality of the state space and providing an approximate solution [27]. Let $\phi_f(s)$, $f \in \mathcal{F}$, denote a basis function where $\mathcal{F}$ is a set of features. The value function approximation is given by

$$\bar{J}^x(s_t^x|\theta) = \sum_{f \in \mathcal{F}} \theta_f \phi_f(s_t^x),$$

wherein $\theta = (\theta_f)_{f \in \mathcal{F}}$ is a column vector of weights with one coefficient for each basis function. Because we choose the number of features to be fewer than the size of the state space, it is computationally efficient to estimate the value function using basis functions. Although classical linear regression methods can be used to estimate θ, choosing an appropriate set of basis functions can be challenging. LSTD updates θ iteratively during execution of the API algorithm.

LSTD iteratively updates the value function approximation for a fixed policy and projects it over an infinite horizon. LSTD differences the current value of being in a state with the updated value of being in a state at the following iteration. Alternatively, this procedure can be viewed as a batch algorithm that operates by collecting samples of temporal differences and then using least squares regression to find the best linear fit [27]. LSTD obtains a least squares regression fit so that the sum of the temporal differences over the simulation is equal to zero.

Within the construct of LSTD, a total of K temporal difference sample realizations are collected in each policy evaluation loop where the kth temporal difference is denoted $C(s_{t,k}, X_\theta^\pi(s_{t,k})) + \gamma\theta^\top\phi(s_{t,k}^x) - \theta^\top\phi(s_{t-1,k}^x)$ where $\phi(\cdot)$ is a column vector of basis function evaluations and the policy (i.e., decision function) $X_\theta^\pi(s_{t,k})$ is defined below

$$X_\theta^\pi(s_{t,k}) = \underset{x \in \mathcal{X}(s_t)}{\arg\max}\ C(s_t, x) + \gamma \bar{J}(s_t^x|\theta).$$

To solve the approximate dynamic program, we need to solve the inner maximization problem. Although the inner maximization problem can be solved exactly using complete enumeration for smaller problem instances, our approach is intended to solve larger problem instances wherein enumeration is not tractable. We formulate the inner maximization problem as an integer program (IP) because only an integer number of CUAVs can be sent for resupply. We define our IP as follows:

Decision Variables:

x^d, integer number of fully loaded CUAVs sent to resupply the FOB.

Parameters:

θ_f, coefficient value corresponding to action taken.

θ_0, coefficient value corresponding to the number of CUAVs available.

V^{cap}, CUAV holding capacity.

V^{crew}, number of crews available.

v_t, number of CUAVs available at time t.

λ, time discount factor.

IP:

$$\max_{x^d} : x^d(\psi_m V^{cap} + \lambda \sum_{f \in \mathcal{F}} (\theta_f - \theta_0)) \tag{7}$$

subject to:

$$x^d \leq \min\left(V^{crew}, v_t\right) \tag{8}$$

$$\psi_m V^{cap} x^d - R^{max} + r_t + \bar{d} \leq 0 \tag{9}$$

$$x^d \in \mathbb{N}^0 \tag{10}$$

The objective function 7 balances current rewards and future expected rewards for the FOB. Constraint 8 limits our actions to utilizing at most the total number of CUAVs available as dictated by the CUAS crew limitations. Constraint 9 limits the expected amount of supplies delivered to be no greater than the FOB capacity. The final constraint enforces integer restrictions on the x^d decision variable.

Let Φ_{t-1} and Φ_t consist of rows of basis function evaluations of the sampled post-decision states and C_t as the contribution vector for the sampled events as shown in Equation 11. The sample realization $\hat{\theta}$ is calculated using linear regression for each policy evaluation loop $n = 1, 2, ..., N$. A harmonic step-size rule is applied to smooth θ during implementation.

$$\Phi_{t-1} \triangleq \begin{bmatrix} \phi(s^x_{t-1,1})^\top \\ \vdots \\ \phi(s^x_{t-1,K})^\top \end{bmatrix}, \Phi_t \triangleq \begin{bmatrix} \phi(s^x_{t,1})^\top \\ \vdots \\ \phi(s^x_{t,K})^\top \end{bmatrix}, C_t \triangleq \begin{bmatrix} C(s_{t,1}, x_t) \\ \vdots \\ C(s_{t,K}, x_t) \end{bmatrix} \tag{11}$$

Jiang & Powell [16] develop a Monotone-ADP algorithm that handles finite-hori–zon MDPs utilizing an approximate value iteration algorithmic framework. Although they discuss possible extensions of the Monotone-ADP algorithm that can be utilized for infinite-horizon cases, they do not explicitly develop an extension of the Mono–tone-ADP algorithm that can handle cases wherein the use of value functions based on look-up tables is intractable. We first delineate their initial idea of how to ex–tend Monotone-ADP to handle infinite-horizon problems using a linear architecture of basis functions, then discuss our proposed solution methodology.

Recall that the value function approximation is given by

$$\bar{J}(s|\theta) = \sum_{f \in \mathcal{F}} \theta_f^n \phi_f(s),$$

where the reliance on the parameter vector θ and the associated policy improvement iteration counter n is made more explicit here. The basis for expanding the Mono–tone-ADP algorithm to the infinite horizon version is accomplished by examining the linear architecture where updates are given by

$$\theta^n \in \arg\min_{\theta} \{\|\theta - \theta^{n-1}\|_2 : \ \bar{J}(s_t^n|\theta) = z^n(s_t^n) \text{ and } \bar{J}(s_t^n|\theta) \text{ is monotone}\},$$

wherein $z^n(s_t^n)$ is the the approximated value smoothed with the value from the pre–ceding iteration. The problem with this approach is that there is no easily computable solution to this update function. Since this update happens within the loop structure of the algorithm, this approach also poses computational concerns. Jiang & Powell [16] point out that special cases may exist that make solving this problem quickly feasible.

We propose constrained linear least-squares optimization as a monotone projection operator to enforce value function monotonicity within the construct of LSTD. For constrained linear least-squares problems, we solve a convex optimization problem of the following form:

$$\min_{\theta} \frac{1}{2}||E\theta - e||_2^2 \ s.t. \ G\theta \leq g. \tag{12}$$

The E matrix is the matrix of sampled observations generated from the LSTD sam–pling (i.e., $(\Phi_{t-1} - \gamma\Phi_t)$) with e being the observed value (i.e., C_t). Herein we explore two possible ways to generate the constraint matrix G with the associated constant g. The first involves generating $K-1$ constraints for the K observations by the following

formula:

$$(a_i - a_{i+1})\theta \leq 0,\ i = 1, 2, ..., K-1$$

where a_i is a monotone ordering of sampled observations. Although this constraint matrix enforces monotonicity, it may result in memory issues due to the need to generate $K-1$ constraints for each dimension. Alternatively, Ahrens [2] shows the necessary and sufficient conditions that ensure global monotonicity in one or more dimensions based on the intuition derived from calculus (i.e., a function is monotone if $\frac{d}{dx}f(x) \geq 0$). With each dimension arranged in non-decreasing order and scaled to lie in the interval $[0, 1]$, the necessary and sufficient conditions for non-decreasing monotonicity of the quadratic response surface is given by Equation 13:

$$\theta_f^{linear} + 2\min(\theta_f^{quadratic}, 0) + \sum_{f' \in \mathcal{F}, f \neq f'} \min(\theta_{ff'}^{cross}, 0) \geq 0,\ \forall f \in \mathcal{F} \tag{13}$$

wherein $\theta^{linear}, \theta^{quadratic}$, and θ^{cross} represent all linear, quadratic, and cross product features, respectively.

Our algorithm takes the batch sampling from the LSTD methodology and gener–ates the sample realization $\hat{\theta}$, enforcing monotonicity projection by solving Equation 12 and utilizing Equation 13 to generate monotonicity constraints. Moreover, once our algorithm has generated a policy, we perform a post-processing policy improve–ment search based on well-performing polices. The monotone least squares temporal differences (MLSTD) pseudo code is summarized in Algorithm 1.

We then apply a harmonic stepsize rule to smooth in the new observation $\hat{\theta}$ with the previous estimate θ during implementation. The stepsize rule α_n is a function of the outer loop iteration count and is defined below.

$$\alpha_n = \frac{1}{n} \tag{14}$$

Algorithm 1 Approximate Policy Iteration Using Monotone Least Square Temporal Differences (MLSTD)

Step 0. Initilize θ^0
Step 1. **For** n=1 to N **(Policy Improvement Loop)**
  Step 2. **For** k=1 to K **(Policy Evaluation Loop)**
    a. Generate a random post-decision state, $s^x_{t-1,k}$
    b. Record $\phi(s^x_{t-1,k})$
    c. Simulate transition to next event; obtain pre-decision state $s_{t,k}$
    d. Determine decision $x = X^{\pi}_{\theta^{n-1}}(s_{t,k})$
    e. Record contribution $C(s_{t,k}, x)$
    f. Record basis function evaluation $\phi(s^x_{t,k})$
    End
  Step 3. Compute θ^n using monotone projection operator Equation 12 and smoothing rule Equation 15
  End
Step 4. Perform post-processing policy improvement Equation 16

The stepsize rule α_n greatly influences the rate at which the API algorithm con–verges, thus impacting the attendant solutions. Utilizing the harmonic stepsize rule, we update our θ in the following way:

$$\theta^n \leftarrow \theta^{n-1}(1 - \alpha_n) + \hat{\theta}(\alpha_n). \tag{15}$$

Equation 15 shows that the updated θ^n is weighted most heavily by our current estimate, θ^{n-1}, and then moved toward our new estimate, $\hat{\theta}$, by an incremental amount proportional to α_n. Initially, greater emphasis is placed on $\hat{\theta}$, but as the number of iterations increases the incremental effect of $\hat{\theta}$ is lessened. Moreover, as the number of iterations increases, any single $\hat{\theta}$ has less influence than the estimate based on information from the first $n - 1$ iterations.

Upon obtaining an updated parameter vector θ, we have completed one policy improvement iteration of the algorithm. The parameters N and K are tunable, where N is the number of policy improvement iterations completed and K is the number of policy evaluation iterations completed.

The post-processing policy improvement step in our algorithm allows for well–known policies to affect our ADP-generated policy. This adjustment allows for subject matter expertise to be applied in decision making to improve solution quality. We tailor our policy improvement to balance FOB supply needs and vehicle risk. For each threat map m, we specify that a minimum number of CUAVs, $x_m^{min} \leq V^{crew}$, must be tasked. When the threat map is relatively safer, we send relatively more CUAVs. When the threat map is more dangerous, however, we only send CUAVs when inventory is either extremely low or we have a large number of CUAVs remaining. The post-processing policy improvement step is shown in Equation 16.

$$X_\theta^{\pi'}(s_t) = \min\{\max\{X_\theta^\pi(s_t), x_{m_t}^{min}\}, v_t\} \tag{16}$$

After completion of the post-processing step, our algorithm has generated a policy and terminates.

3.5 Analysis

Utilizing the Markov decision process (MDP) formulation discussed in Section 3.2, we can find a policy for a single, battalion-sized forward operating base (FOB) problem instance. Moreover, we run a designed experiment to find the algorithmic and model parameters that yield the best results for our approximate dynamic programming (ADP) algorithm.

MDP Parameterization.

The military inventory routing problem (MILIRP) is formulated as an infinite horizon MDP wherein a single period represents a 6-hour interval. We assume that during each period the cargo unmanned aerial vehicle (CUAV) can complete all mission preparation tasks and perform the assigned mission. We assume the FOB has

stochastic demand.

Each battalion is made of subordinate platoons that each have a consumption rate and storage capacity based on the number of personnel on site. Based on a General Dynamics report [13], the expected daily consumption requirements of a platoon is $7,482$ pounds. We round up as a conservative estimate to an 8,000 pound daily average consumption. With four periods in one day, about one ton of supplies per period is required for sustainment. For our testing, we model the stochastic demand using this known historical average $\bar{d}$ and a randomly generated error term, $\hat{\epsilon}$, uniformly distributed on the interval $[-0.5, 0.5]$. We also make the conservative assumption that the FOB has a maximum holding capacity of three times the daily average requirement, totaling 12 tons. We assume that there are no logistical failures limiting the amount of supplies available at the centralized brigade support battalion (BSB). This assumption is reasonable since the BSB is supplied via fixed wing aircraft from outside the theater of operations.

Lockheed Martin's K-MAX CUAV has delivered two tons at $15,000$ feet above ground level (AGL) with more tonnage delivered at lower altitudes [19]. Thus, we chose a conservative two ton carrying capacity for CUAV resupply. We also chose the number of CUAVs and crews to be eight and four respectively, which mirrors operations for tactical unmanned aircraft system (TUAS) platoons [10]. As the re–quirements for CUAV resupply increase, we expect to see the number of CUAVs and crews the BSB utilizes to increase. As such, we parameterize the CUAVs and crews as multiples of TUAS platoon ratios. For example, if three TUAS platoons are deployed at the BSB, the number of CUAVs would be 12 and the number of crews 6.

Recall from Section 3.2 that ψ_m denotes the probability of a successful one-way trip from (to) the brigade support area (BSA) to (from) the FOB on map m. An intelligence team would ideally assign risk values to each zone in the tessellated area of

operations (i.e., region). This number would account for threats, to include, but not limited to: weather, enemy action, and mechanical breakdown. Transition between maps can be created by leveraging observed trends specific to the region of interest. These problems influence threat levels, which might include time of the year. For our example, we chose to use $M = 2$ threat maps and parameters $\psi_1 = 0.99$, and $\psi_2 = 0.95$. We vary the transition probability in our experimental design to investigate the effects of region volatility on algorithmic outcome. We select an initial probability of 0.8 to remain in low threat map and 0.2 to remain in a high threat map (i.e., $\Omega = 0.8$ and $\beta = 0.2$) to model the current instability of the region.

When the FOB's supply level falls below a predetermined minimum threshold, the FOB must be resupplied via ground convoy to regain full capacity. When a convoy is sent, the penalty is immediately applied. The penalty represents the increased human capital risk inherent in ground convoy operations along with the risk of a FOB stock out. The penalty associated with resupplying the FOB would ideally be supplied by a subject matter expert who knows the terrain and enemy activity levels associated with the FOB. For example, a FOB further away from the BSA across rough terrain would have a higher penalty than a closer and more readily accessible FOB. This penalty creates a strong incentive to ensure the FOB is resupplied by CUAV when possible. To ensure our problem maintains monotonicity in the value function we apply the result from Section 3.3 by setting $\bar{\tau} = R^{max}$, which is applied when the inventory level is less than or equal to the R^{min} threshold.

We chose $\lambda = 0.98$ to be a discount factor that balances future needs with current needs. We utilized the above described MDP parameterization to create policies using exact, LSTD, and MLSTD solution techniques.

Optimal Policy.

The advantage of examining the single FOB instance is that computing the optimal value function is tractable, and we can use the solution to compare our ADP results. An optimal policy is determined using policy iteration.

ADP Policies.

The ADP policy is obtained from our monotone least squares temporal differences (MLSTD) algorithm using least squares temporal differences and the monotonicity enforcement operator as explained in Section 3.2. We compare MLSTD to an approximate policy iteration algorithm using least squares temporal differences (LSTD). The challenge with both these algorithms is developing basis functions that accurately approximate the optimal value function. These two algorithms are employed with the system initialized at full capacity for the FOB.

We develop ADP policies using our integer program in both LSTD and MLSTD algorithms. Our basis function includes first order effects for current inventory level, CUAVs not deployed, and number of CUAVs deployed. Moreover, we also chose to include the second order inventory effect. The simpler inner maximization problem allows us to perform a designed experiment with more breadth in a reasonable amount of time.

Baseline Instance.

We selected a representative baseline instance as a reference point for testing the MLSTD algorithm's performance. The baseline instance has 12 CUAVs, 2 crews, probability of staying in a low threat map of 0.8, probability of staying in a high threat map of 0.2, average period demand of 1 ton, with associated algorithmic features $K = 5000$ and $N = 30$. Due to the small problem instance, LSTD-, and MLSTD-generated

policies were evaluated exactly. The baseline performance is shown in Table 2. For the baseline instance, MLSTD outperformed LSTD by 4.84% with an optimality gap of only 3.05%. We then expanded our experimental region of search with a designed experiment.

Experimental design.

We created an experimental design to test the robustness of our ADP algorithm and find the parameter settings that allow our proposed algorithmic approach to achieve the best performance. We focused our response variable on the total value of the system when initialized at full capacity. In each experimental run, we simultaneously assess four problem features and three algorithmic features. The four problem features of interest were chosen based on what we thought might have the most effect on the system performance. The problem features we chose to investigate are number of crews available (V^{crew}), probability of staying in a low threat map (Ω), probability of staying in a high threat map (β), and the average demand ($\bar{d}$). The three algorithmic features we chose to experiment on are inner loop iteration count (K), outer loop iteration count (N), and a categorical variable where -1 denotes MLSTD and 1 denotes LSTD. We compared the exact value of each ADP policy to the optimal policy.

Each of the four problem features are considered to be continuous. We chose the crew level to be levels associated with deploying two, three, and four TUAS platoons at the BSB. This was done under the assumption that as commanders increasingly

Table 2. Baseline Instance

Algorithm	Value	% Optimal
LSTD	46.0015	92.11 %
MLSTD	48.4205	96.95 %
Optimal	49.9428	

value CUAV resupply, TUAS platoons will be sent in greater numbers to support brigade operations. The transition probabilities, Ω and β, are parameterized to explore how regional volatility affects the value function. The lower value, 0.2, denotes a low chance of transitioning to a different threat map condition. The higher value, 0.8, denotes a high probability of transitioning to a different threat map condition. Demand was chosen to range from current consumption rates (i.e., 1 ton every 6 hours) to a considerably larger rate of consumption (i.e., 2 tons every 6 hours) to explore how demand schedule increases would affect the system.

The three algorithmic features were chosen to best explore the experimental space. The inner loop count was set to a low of $5,000$ and a high of $15,000$ based on initial testing. The center run is the midpoint of the upper and lower bounds and allows us to determine if our response variable demonstrates nonlinearity. The outer loop iteration counter was similarly chosen, allowing for a large upper bound to achieve the most accurate value function approximation for the basis functions we chose. Table 3 shows the problem and algorithmic settings for our experimental design.

We implemented a 2^{7-2} resolution VII fractional factorial design with two center runs, totaling 66 runs. In a resolution VII design, all first-, second-, and third-order effects are free from being aliased with other first-, second-, or third-order interactions. For each design run, an optimal, MLSTD, and LSTD policy is determined. Recall that an ADP policy utilizes θ coefficients that correspond to selected basis functions.

Table 3. Factorial Design Settings

Description	Factor	Low	Center	High
Number of crews	V^{crew}	2	4	6
Probability of remaining low threat	Ω	0.2	0.5	0.8
Probability of remaining high threat	β	0.2	0.5	0.8
Average Demand	$\bar{d}$	1	1.5	2
Number of inner loops	K	5000	10000	15000
Number of outer loops	N	10	20	30
Algorithm	MLSTD	LSTD	-	MLSTD

After the θ coefficients are determined, we compute the exact values of the resulting MLSTD and LSTD policies and compare them to the optimal policy.

Results.

Tables 4 and 5 show the results from the experiment. The MLSTD policy outperformed the LSTD policy, performing at least as well as LSTD in all design runs. The problem features for which the MLSTD algorithm performed best include when probability of staying in a low threat map and demand were at their high levels, and when number of crews and probability of staying in a high threat map were at their low levels. This intuitive result is caused by MLSTD's ability to send more CUAVs in this relatively safer operating condition via the post-processing step in a way LSTD does not. This treatment achieved an optimality gap of 3.0%. Interestingly the policy value was invariant to outer and inner loop count. We found that MLSTD would quickly converge to the estimated optimal policy. Moreover, although subsequent iterations would update θ^n, the generated policy would be invariant to the resulting change.

MLSTD performs most poorly with a low probability of remaining in a low threat condition and a high probability of remaining in a high threat condition, resulting in a much lower optimality gap of 17.93%. The less conservative strategy is penalized more heavily when conditions are more consistently dangerous for CUAV operation.

Both MLSTD and LSTD perform better when a lower number of crews are considered due to the limited CUAV deployment. Moreover, MLSTD greatly outperformed LSTD in many cases. Particularly, in runs 40, 41, 46, 47, 58, 59, 61, and 64 LSTD had an optimality gap in excess of 66%. LSTD poorly estimated the value function for problem instances with many of the same problem features as MLSTD but with a much greater optimality gap. On average, MLSTD performed 31.86% better than

LSTD as shown in Table 6. These experimental results indicate which parameters are influential in the structure of the MILIRP. However, a metamodel is necessary to draw direct conclusions.

We next created a regression metamodel to analyze the effects with more statis–tical rigor. A stepwise regression procedure yields factors that produce a significant relationship and pass the lack of fit test. A Box-Cox transformation analysis confirms that the error is minimized with a square root transformation. The resulting model includes significant first- and second-order terms and performs very well in terms of prediction ability with an adjusted R^2 of 0.9913. The residuals do not show signs of heteroscedasticity, and the residual by predicted plot does not raise concerns.

Table 7 summarizes the significant variables. We consider variables with an F-test p-value less than 0.05 significant. With this criterion, number of crews, probability of staying in a high threat condition, probability of staying in a low threat condition, and using MLSTD are all significant in both first- and second-order terms. Average demand is significant in second-order interaction. Using inner loop and outer loop iteration counts are not significant significant even when higher order effects are considered. This is due to the fast convergence of the ADP algorithm to the resulting ADP-generated policy.

According to our metamodel, the optimality gap is minimized with the probability of staying in a high threat map of 0.2, the probability of staying in a low threat map of 0.8, $5,000$ inner loops, 10 outer loops, 2 crews, 1 ton of average demand, and using MLSTD.

3.6 Conclusions

This paper examines the military inventory routing problem (MILIRP). The in–tent of this research is to examine the MILIRP's structural properties and develop

Table 4. Experimental Results

Run	V^{crew}	Ω	β	$\bar{d}$	K	N	Algorithm	$\bar{J}$	J^*	Optimality Gap
1	2	0.2	0.2	1	5000	10	LSTD	37.53	49.84	24.70%
2	2	0.2	0.2	1	5000	30	MLSTD	45.90	49.84	7.90%
3	2	0.2	0.2	1	15000	10	MLSTD	45.90	49.84	7.90%
4	2	0.2	0.2	1	15000	30	LSTD	37.53	49.84	24.70%
5	2	0.2	0.2	2	5000	10	MLSTD	91.46	98.91	7.53%
6	2	0.2	0.2	2	5000	30	LSTD	74.30	98.91	24.88%
7	2	0.2	0.2	2	15000	10	LSTD	74.30	98.91	24.88%
8	2	0.2	0.2	2	15000	30	MLSTD	91.46	98.91	7.53%
9	2	0.2	0.8	1	5000	10	MLSTD	42.58	49.08	13.25%
10	2	0.2	0.8	1	5000	30	LSTD	28.59	49.08	41.76%
11	2	0.2	0.8	1	15000	10	LSTD	28.59	49.08	41.76%
12	2	0.2	0.8	1	15000	30	MLSTD	42.58	49.08	13.25%
13	2	0.2	0.8	2	5000	10	LSTD	55.86	87.74	36.34%
14	2	0.2	0.8	2	5000	30	MLSTD	84.14	87.74	4.11%
15	2	0.2	0.8	2	15000	10	MLSTD	84.14	87.74	4.11%
16	2	0.2	0.8	2	15000	30	LSTD	55.86	87.74	36.34%
17	2	0.8	0.2	1	5000	10	MLSTD	48.42	49.94	3.05%
18	2	0.8	0.2	1	5000	30	LSTD	46.00	49.94	7.89%
19	2	0.8	0.2	1	15000	10	LSTD	46.00	49.94	7.89%
20	2	0.8	0.2	1	15000	30	MLSTD	48.42	49.94	3.05%
21	2	0.8	0.2	2	5000	10	LSTD	91.76	99.72	7.98%
22	2	0.8	0.2	2	5000	30	MLSTD	96.72	99.72	3.00%
23	2	0.8	0.2	2	15000	10	MLSTD	96.72	99.72	3.00%
24	2	0.8	0.2	2	15000	30	LSTD	91.76	99.72	7.98%
25	2	0.8	0.8	1	5000	10	LSTD	37.06	49.64	25.35%
26	2	0.8	0.8	1	5000	30	MLSTD	45.80	49.64	7.74%
27	2	0.8	0.8	1	15000	10	MLSTD	45.80	49.64	7.74%
28	2	0.8	0.8	1	15000	30	LSTD	37.06	49.64	25.35%
29	2	0.8	0.8	2	5000	10	MLSTD	91.04	96.08	5.25%
30	2	0.8	0.8	2	5000	30	LSTD	73.32	96.08	23.69%
31	2	0.8	0.8	2	15000	10	LSTD	73.32	96.08	23.69%
32	2	0.8	0.8	2	15000	30	MLSTD	91.04	96.08	5.25%
33	6	0.2	0.2	1	5000	10	MLSTD	41.15	49.84	17.43%

Table 5. Experimental Results Continued

Run	V^{crew}	Ω	β	$\bar{d}$	K	N	Algorithm	$\bar{J}$	J^*	Optimality Gap
34	6	0.2	0.2	1	5000	30	LSTD	17.38	49.84	65.13%
35	6	0.2	0.2	1	15000	10	LSTD	17.38	49.84	65.13%
36	6	0.2	0.2	1	15000	30	MLSTD	41.15	49.84	17.43%
37	6	0.2	0.2	2	5000	10	LSTD	32.77	99.23	66.97%
38	6	0.2	0.2	2	5000	30	MLSTD	81.60	99.23	17.76%
39	6	0.2	0.2	2	15000	10	MLSTD	81.60	99.23	17.76%
40	6	0.2	0.2	2	15000	30	LSTD	32.77	99.23	66.97%
41	6	0.2	0.8	1	5000	10	LSTD	6.49	49.20	86.80%
42	6	0.2	0.8	1	5000	30	MLSTD	40.38	49.20	17.93%
43	6	0.2	0.8	1	15000	10	MLSTD	40.38	49.20	17.93%
44	6	0.2	0.8	1	15000	30	LSTD	6.49	49.20	86.80%
45	6	0.2	0.8	2	5000	10	MLSTD	79.63	92.06	13.50%
46	6	0.2	0.8	2	5000	30	LSTD	10.34	92.06	88.77%
47	6	0.2	0.8	2	15000	10	LSTD	10.34	92.06	88.77%
48	6	0.2	0.8	2	15000	30	MLSTD	79.63	92.06	13.50%
49	6	0.8	0.2	1	5000	10	LSTD	32.39	49.94	35.14%
50	6	0.8	0.2	1	5000	30	MLSTD	42.82	49.94	14.27%
51	6	0.8	0.2	1	15000	10	MLSTD	42.82	49.94	14.27%
52	6	0.8	0.2	1	15000	30	LSTD	32.39	49.94	35.14%
53	6	0.8	0.2	2	5000	10	MLSTD	85.12	99.73	14.65%
54	6	0.8	0.2	2	5000	30	LSTD	63.71	99.73	36.12%
55	6	0.8	0.2	2	15000	10	LSTD	63.71	99.73	36.12%
56	6	0.8	0.2	2	15000	30	MLSTD	85.12	99.73	14.65%
57	6	0.8	0.8	1	5000	10	MLSTD	41.22	49.66	16.98%
58	6	0.8	0.8	1	5000	30	LSTD	16.60	49.66	66.57%
59	6	0.8	0.8	1	15000	10	LSTD	16.60	49.66	66.57%
60	6	0.8	0.8	1	15000	30	MLSTD	41.22	49.66	16.98%
61	6	0.8	0.8	2	5000	10	LSTD	31.17	96.82	67.81%
62	6	0.8	0.8	2	5000	30	MLSTD	81.54	96.82	15.78%
63	6	0.8	0.8	2	15000	10	MLSTD	81.54	96.82	15.78%
64	6	0.8	0.8	2	15000	30	LSTD	31.17	96.82	67.81%
65	4	0.4	0.4	1.5	10000	20	MLSTD	53.97	62.46	13.59%
66	4	0.4	0.4	1.5	10000	20	LSTD	53.97	62.46	13.59%

Table 6. Algorithm Performance Summary (Percent Optimal)

Algorithm	Min	Average	Max
MLSTD	82.07%	88.67%	97.00%
LSTD	11.23%	56.81%	92.11%
Difference	70.84%	31.86%	4.89%

Table 7. Factors Influencing CUAV Resupply Amount

Variable	Sum of Squares	F Test	% Contribution
V^{crew}	0.8817961	$< .0001$	29.42%
Ω	0.1773044	$< .0001$	5.92%
β	0.1493092	$< .0001$	4.98%
$\bar{d}$	0.0056066	0.087	0.19%
Algorithm	1.4872921	$< .0001$	49.63%
V^{crew}*Algorithm	0.107545	$< .0001$	3.59%
$\Omega * \beta$	0.0236994	0.0007	0.79%
Ω*Algorithm	0.0663471	$< .0001$	2.21%
$\beta * \bar{d}$	0.0075274	0.0484	0.25%
β*Algorithm	0.0903966	$< .0001$	3.02%

an algorithm that can help improve cargo unmanned aerial vehicles (CUAV) resupply performance. Development of a Markov decision process (MDP) model of the MILIRP enables examination of the disparate conditions the military faces in hostile environments.

Management of CUAV assets for resupply is an important issue to the United States military. Poorly developed transportation infrastructure, adverse weather conditions, terrain, enemy threat and actions, and the availability of distribution assets all inhibit successful distribution of supplies from the brigade support area (BSA) to the forward operating bases (FOBs). Moreover, insurgent use of improvised explosive devices (IEDs) greatly affects truck mobility throughout the operational environment and has been successful in disrupting replenishment procedures [25]. Since 2012 when the K-MAX successfully deployed to Afghanistan [15], CUAVs have been of increasing interest both to the United States and worldwide [14]. This paper provides unique insight into using CUAVs in combat environments for resupply. High casualty rates for convoy resupply missions have highlighted the importance of CUAV aerial resupply. CUAV benefits include: better performance in adverse weather conditions, higher flight ceilings, and no escort requirement restrictions. All these yield a lower probability of vehicle destruction via man-portable air-defense systems and small arms fire. The most important benefit of CUAVs is their ability to save lives by alleviating manned ground convoy resupply requirements. Although CUAVs do not yet have the ability to completely handle FOB supply requirements, each successful CUAV delivery means less men and women exposed to enemy threats to include IEDs.

We examined the MILIRP's structural properties and developed an algorithm that exploits this special structure. We have mathematically proven how the penalty affects value function monotonicity specific to the MILIRP. We formulated an MDP model of the MILIRP and determined the optimal policy on small instances in order to

compare two approximate dynamic programming algorithms. We tested our approach with an experimental design and empirically found that when a sufficient penalty is applied, monotone least squares temporal differences (MLSTD) performs statistically better and, ceteris paribus, performs no worse than least squares temporal differences (LSTD).

Although MLSTD performed statistically better than LSTD in all tested in–stances, the experimental region was explored in regions for which the post-processing local search was designed. If we lower the number of CUAVs dispatched, the local search heuristic would likely need to be tailored for the specific region of interest.

3.7 Follow-on Research

An important extension of our work involves expanding the number of forward operating bases allowing routing CUAVs to deliver to more than one customer on a delivery route. The integer program used within our algorithm will have to be modified because we will have to simultaneously solve both a vehicle routing problem and determine the optimal quantities to be delivered to each customer on a delivery route.

3.8 Acknowledgments

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

IV. Utilizing Heuristics Within a Least Squares Temporal Differences Algorithm to Solve a Large Instance Stochastic Inventory Routing Problem with Vehicle Loss

The remainder of this dissertation will refer to the brigade support battalion as the central planner and the brigade support area as the support area (SA). This chapter considers a military variant of the stochastic inventory routing problem with vehicle loss in which the central planner must decide how many fully loaded CUAVs to dispatch to fulfill the demand requirements of multiple FOBs. To fill this demand, multiple delivery across a finite set of feasible CUAV routes is allowed. A Markov decision process (MDP) model of the MILIRP that extends previous work on this military inventory routing problem (MILIRP) in Chapter III is developed. The unique aspect of the MILIRP includes the ability of enemy actions to permanently destroy the transportation vehicles. Although convoy resupply currently makes up the majority of tonnage of supplies delivered, insurgent use of improvised explosive devices make convoy resupply costly and dangerous [25]. The effective use of cargo unmanned aerial vehicles (CUAVs) can be instrumental in reducing loss of human capital in wartime environments.

This chapter improves upon current research by increasing the size of the problem instances previously considered and implementing and testing the monotone least squares temporal differences (MLSTD) algorithm as developed in Chapter III. Increasing the state, action, and outcome spaces greatly increases the computational complexity so that exact dynamic programming algorithms cannot be implemented. Moreover, this chapter develops two new heuristic algorithms (i.e., Index and Rollout) and embed the Index and Rollout algorithms within the approximate dynamic programming (ADP) techniques to solve these larger instances. To demonstrate the efficacy of the proposed solution methodology, a notional, representative planning

scenario based on an austere combat environment wherein convoy resupply may be difficult like that of Afghanistan is constructed. The solution quality of the developed Index and Rollout algorithms and two ADP techniques (i.e., least squares temporal differencing (LSTD) and monotone LSTD as developed in Chapter III), which embed these heuristics, are compared to a simple and easily implemented myopic strategy.

4.1 Markov Decision Process Model

This section describes the Markov decision process (MDP) model formulation of the military inventory routing problem (MILIRP) that extends previous work in Chapter III. The objective of the MILIRP is to determine the optimal resupply from a central planner with multiple forward operating bases (FOBs) via inventory routing decisions in order to maintain inventory. The reward function only rewards inventory delivered that does not exceed FOB holding capacity. We assume the central planner knows the inventory level at each FOB at the start of each period and that demand has a known historical average with some variability. We model this variability as an independent and identically distributed error term. We assume that no other external event (e.g., fire, theft) other than demand causes a loss of inventory.

The central planner distributes supplies to the FOBs utilizing V identical cargo unmanned aerial vehicles (CUAVs). Each CUAV has an identical load capacity of V^{cap} tons. We construct a finite set of routes $a \in \mathcal{A}$ that denotes the feasible routes for CUAV routing. When multiple FOBs are visited along the CUAV route, it is assumed that the load is evenly distributed among each FOB. V^a is the amount of supplies delivered to each FOB along route a. FOBs require $\hat{D}_t$ tons of supplies per time period t, a stochastic demand with a mean demand $\bar{d}$ and an independent and identically distributed exogenous error term $\hat{\epsilon}$. FOBs also have a finite maximum holding quantity R^{max}.

Given dangers inherent in military resupply in a combat environment, there is potential for delivery failure due to extrinsic uncontrollable factors (e.g., enemy action, mechanical failure, extreme weather conditions). The probability of a CUAV being destroyed depends on the threat conditions from these extrinsic factors. A set of M threat maps models the periodic changes in risk throughout the central planner's area of operations. For threat map $m = 1, 2, ..., M$, the parameter ψ_{ma} denotes the probability of a successful trip from (to) the support area (SA) to (from) the FOB on the specified route a. A CUAV may be destroyed on any segment of the given route to include it's flight to a FOB or after delivering supplies on the return route back to the depot at the SA.

We next describe the MDP model formulation of the MILIRP. With respect to a conventional inventory routing formulation, CUAVs are vehicles, FOBs are customers, and the centralized planner is the supplier. Tables 8 and 9 located at the end of this section provide a summary of the notation.

The MILIRP is formulated as an infinite horizon problem wherein at each decision epoch $t \in \mathcal{T} = \{1, 2, ...\}$ an inventory routing decision is made. During each time period a CUAV completes all tasks necessary to travel the specified route and return to the SA. In this research routes are limited to only visiting a maximum of three FOBs. Moreover, it is assumed that the route is within the fully loaded CUAV's range and that this route is serviceable in one time period. Current CUAV limitations validate this assumption [19].

The state space includes three components: the inventory level at each of the FOBs, the number of operational CUAVs, and the threat map index number. The inventory for all FOBs is defined as r_t, where $r_t = (r_{t1}, r_{t2}, ..., r_{tB})$ and $r_{tb} \in (0, R^{max})$ is the number of tons of supplies at FOB $b \in \{1, 2, ..., B\}$ at time t. Moreover, R^{max} is the maximum inventory capacity for the FOBs, and $R^{min} \in (0, R^{max})$ is the

minimum threshold inventory level that must be exceeded (i.e., the safety stock level). If $r_{tb} \leq R^{min}$ then resupply via convoy ground lines of communication (GLOC) is required for that FOB. The number of operational CUAVs able to perform resupply operations at time t is defined as v_t. The threat map index number at time t is defined as $m_t \in \{1, 2, ..., M\}$. The threat map impacts the flight risk associated with successfully completing sorties between the FOBs and the SA. The threat map information m_t is available at time t. The threat map information m_{t+1} available at time $t+1$ is conditioned on m_t and is unknown at time t. Utilizing these components, we define $s_t = \big(r_t, v_t, m_t\big) \in \mathcal{S}$ as the state of the system at time t, where $\mathcal{S}$ is the set of all possible states. Furthermore, each CUAV begins and ends each day at the SA.

We let $\mathcal{X}(s_t)$ be the set of all feasible actions when the system is in state s_t. Let $x_t = (x_t^d, x_t^{GLOC}) \in \mathcal{X}(s_t)$ denote an inventory routing decision wherein $x_t^d = (x_{t1}^d, x_{t2}^d, ..., x_{tA}^d)$, $x_t^{GLOC} = (x_{t1}^{GLOC}, x_{t2}^{GLOC}, ..., x_{tB}^{GLOC})$, $x_{ta}^d \in \mathbb{N}^0$ denotes the number of fully loaded CUAVs dispatched to resupply the FOBs along route $a \in \mathcal{A}$ and $x_{tb}^{GLOC} \in \{0, 1\}$ denotes whether a ground convoy is dispatched to resupply FOB b, which results in its inventory level increasing to capacity. Only CUAV resupply is available for a FOB if the inventory level is greater than the safety stock threshold (i.e., $r_{tb} > R^{min}$). Only GLOC resupply is available if the inventory level is less than or equal to the safety stock threshold (i.e., $r_{tb} \leq R^{min}$). Two constraints impact the CUAV routing decision: first, the number of CUAVs deployed cannot exceed the number of operational CUAVs (i.e., $x_t^d \leq v_t$); second, the number of CUAVs deployed cannot exceed the number of crews available (i.e., $x_t^d \leq V^{crew}$). We assume that each CUAV carries a maximum capacity load of V^{cap} and divides its load equally among all the FOBs visited along the route. The policy (i.e., decision function) $X^\pi(s_t)$ returns a decision $x_t \in \mathcal{X}(s_t)$ as a function of the system state $s_t \in \mathcal{S}$. After a routing decision is made, delivery is performed within one time period.

Transition probabilities are defined for each dimension of the state space to include the inventory level at each FOB, number of remaining CUAVs, and threat map. Inventory transitions are based on the routing decision x_t and the current state of the system s_t. When CUAVs are routed to the FOBs there are two possible outcomes: first, a CUAV may successfully travel along its route and return to the SA; second, a CUAV may be destroyed along its route before returning to the SA. Let ψ_{ma}^{SSSS} denote the probability of a successful round trip delivery along a three FOB route a, ψ_{ma}^{SSSF} denote the probability of a successful three leg FOB delivery along a three FOB route a, ψ_{ma}^{SSFF} denote the probability of a successful two leg FOB delivery along a three FOB route a, ψ_{ma}^{SFFF} denote the probability of a successful one leg FOB delivery along a three FOB route a, and ψ_{ma}^{FFFF} denote the probability of a failed delivery along a three FOB route a for a single CUAV routed to resupply FOBs during the threat conditions of map $m = 1, 2, ..., M$. Outcome probabilities for 1- and 2-FOB routes are similarly defined.

Since we are interested in a particular outcome of a routing decision, we proceed by defining the binomial marginal distributions for each outcome type (i.e., SSSS, SSSF, SSFF, SFFF, FFFF, SSS, SSF, SFF, FFF, SS, SF, FF). With the assumption that each outcome of a resupply mission to the FOBs are independent of other missions and recalling that x_t includes the decision to route x_t^d CUAVs to FOBs (each carrying a full supply load), we let $\hat{Z}_{t+1,b}^{SSSS}(\psi_{ma}^{SSSS}, x_{ta}^d)$ denote the binomial random variable with parameters ψ_{ma}^{SSSS} and x_{ta}^d that indicates the number of successful round trip CUAV deliveries to FOB b in route a during time interval $[t, t+1)$ on map m. Let all other outcomes (i.e., SSSS, SSSF, SSFF, SFFF, FFFF, SSS, SSF, SFF, FFF, SS, SF, FF) be similarly defined. For compactness, we refer to the set of random variables that indicate resupply mission outcomes for each FOB as follows:

$$\hat{Z}_{t+1,b} = \left\{ \hat{Z}_{t+1,b}^{SSSS}, \hat{Z}_{t+1,b}^{SSSF}, ..., \hat{Z}_{t+1,b}^{SS}, \hat{Z}_{t+1,b}^{SF}, \hat{Z}_{t+1,b}^{FF} \right\}. \tag{17}$$

Moreover, the outcome denoting the number of CUAVs that are destroyed in time interval $[t, t+1)$ is the summation of all failures as shown in Equation 20 where we drop the FOB distinction b, thereby denoting the total number of failures across all FOBs as demonstrated in Equation 18.

$$\hat{Z}_{t+1}^{SSSF} = \sum_{b=1}^{B} \hat{Z}_{t+1,b}^{SSSF} \tag{18}$$

The inventory levesl at the FOBs are limited by the maximum holding quantity R^{max}. Moreover, if the FOB supply level is less than or equal to a safety stock threshold, R^{min}, the FOB must be fully resupplied via ground convoy. Equation 19 is the inventory transition function for the FOB.

$$r_{t+1,b} = \begin{cases} R^{max} & \text{if } x_{tb}^{GLOC} = 1 \\ \min\left(r_{tb} + V^a(\hat{Z}_{t+1,b}) - \hat{D}_{t+1}, R^{max}\right) & \text{if } \sum\limits_{a \in \mathcal{A}} x_{tab}^{d} > 0 \\ r_{tb} - \hat{D}_{t+1} & \text{otherwise.} \end{cases} \tag{19}$$

In the first case, convoy resupply is selected, and the FOB is resupplied to capacity. In the second and third cases, the FOB inventory level changes according to supplies received and realized demand. The minimization in the second case enforces the FOB capacity constraint.

CUAV transition is contingent on the number of CUAVs that fail to return to the SA after attempting to travel their route. The number of CUAVs transition according to Equation 20.

$$v_{t+1} = v_t - (\hat{Z}_{t+1}^{SSSF} + \hat{Z}_{t+1}^{SSFF} + \hat{Z}_{t+1}^{SFFF} + \hat{Z}_{t+1}^{FFFF} + \hat{Z}_{t+1}^{SSF} + \hat{Z}_{t+1}^{SFF} + \hat{Z}_{t+1}^{FFF} + \hat{Z}_{t+1}^{SF} + \hat{Z}_{t+1}^{FF}) \tag{20}$$

The map transition function is a representation of the uncontrolled stochastic as– pect of the combat environment. The set of all maps captures the threat level of the operational environment. The map transitions are representative of the changing

environment. For relatively static combat conditions, the map transition probability would be relatively low. More dynamic combat environments yield a relatively higher map transition probability. The central supplier's intelligence teams gather information on threat conditions based on information such as enemy action, season, historical trends, and weather.

The contribution function rewards the system based on the amount of supplies delivered by CUAV to the FOBs. The amount of supplies delivered is bounded above by the maximum inventory quantity at each FOB, constraining any excess supplies delivered from affecting the system behavior. An immediate penalty, $\bar{\tau} > 0$, is applied if the FOB's inventory level is less than or equal to the safety stock threshold R^{min} due to the human risk associated with ground convoy resupply. The below-threshold penalty function for each FOB,

$$\tau(r_b) = \begin{cases} 0 & \text{if } r_b > R^{min} \\ \\ \bar{\tau} & \text{if } r_b \leq R^{min} \end{cases}, \tag{21}$$

allows the application of a penalty that can capture the difficulty of resupplying FOBs via ground convoy. We chose to set the penalty at the max inventory level at the FOB, R^{max}, so as to encourage CUAV resupply. Moreover, previous results in Chapter III indicate that on smaller instances monotonicity can be maintained if the penalty is set to this value. We present our contribution function in Equation 22.

$$C(s_t, x_t) = \mathbb{E}\Big\{ \sum_{b=1}^{B} \min \Big(R^{max} - r_{tb} + \hat{D}_{t+1}, V^a \cdot (\hat{Z}_{t+1,b}) \Big) - \tau(r_{tb}) \Big| s_t, x_t \Big\} \tag{22}$$

The single-period contribution (i.e., reward) is determined by the amount of supplies successfully delivered to the FOBs. However, the system is not rewarded for excess supplies (i.e., FOBs cannot take delivery of supplies in excess of their capacity).

The objective of this MDP is to maximize the expected total discounted reward over an infinite horizon. By definition, the transitions are Markovian. All decisions made at time t depend only on the current state of the system. To obtain the policy that maximizes the expected total discounted reward, Bellman's optimality equation, shown in Equation 23, is solved.

$$J(s_t) = \max_{x \in \mathcal{X}(s_t)} \Big(C(s_t, x) + \lambda \mathbb{E}\{J(s_{t+1}) | s_t, x\} \Big) \tag{23}$$

The value of being in state s_t results from choosing the action that maximizes the expected immediate contribution and the discounted expected future value of the system at time $t+1$. The parameter $\lambda \in [0, 1)$ denotes the discount factor. Using this MDP formulation, an approximate dynamic programming algorithm can be developed to obtain policies for resupplying the FOB via CUAVs.

4.2 Solution Methodology

This section introduces the Index and Rollout algorithms based on the quiz problem heuristic that balance risk with potential rewards to more effectively solve the military inventory routing problem (MILIRP), a variant of the stochastic inventory routing problem (IRP). These algorithms can be applied individually or within the construct of approximate dynamic programming. For comparison, we also introduce an adapted monotone least squares temporal differences (MLSTD) algorithm as developed in Chapter III and show how we can utilize either the Index or Rollout algorithmic technique to solve the required inner maximization problem. We first present the least squares temporal differences (LSTD) algorithm and note where and how the monotone variant, MLSTD, differs.

Table 8. Table of Notation

a	=	feasible route counter
b	=	forward operating base counter
c_i	=	monotone ordering of sampled observations
C	=	contribution function
d	=	daily FOB demand
e	=	observed value for constrained least squares optimization
E	=	matrix of sampled observations generated from LSTD sampling
F	=	failure to deliver supplies
g	=	route segment or leg of route a
h	=	constraint for constrained least square optimization
H	=	constraint matrix for constrained least square optimization
I	=	indicator variable
J	=	total expected reward (cost-to-go) function
k	=	policy evaluation loop counter
K	=	number of policy evaluation loops
M	=	number of threat maps
n	=	policy improvement loop counter
N	=	number of policy improvement loops
r	=	supplies on hand
R^{min}	=	supply threshold
R^{max}	=	FOB holding capacity
s	=	state of system
t	=	time epoch
v	=	current number of CUAVs
V	=	number of inital CUAVs

Table 9. Table of Notation Continued

V^a	=	the amount of supplies delivered to each FOB along route a
V^{cap}	=	CUAV holding capacity
V^{crew}	=	number of crews
w	=	exogenous information process
x	=	actions
Z	=	set of random variables corresponding to the number of possible SS, SF, and F events
$\mathcal{A}$	=	set of all feasible routes
$\mathcal{F}$	=	set of basis function features
$\mathcal{G}$	=	set of all feasible legs
$\mathcal{S}$	=	state space
$\mathcal{T}$	=	set of time epochs
$\mathcal{X}$	=	action space
α	=	stepsize
β	=	probability of remaining in the high threat map
θ	=	vector of weights
λ	=	discount factor
π	=	policy
τ	=	penalty cost
ψ_{ma}	=	one-way probability a CUAV successfully reaches its destination on route a
ϕ	=	basis function
Φ	=	matrix of fixed basis functions
Ω	=	probability of remaining in the low threat map

ADP Formulation.

Both of the considered approximate dynamic programming (ADP) algorithms utilize an approximate policy iteration (API) framework with a least squares tempo–ral differences (LSTD) value function approximation scheme. API mirrors the exact policy iteration algorithm closely. These API algorithms approximate and update the value function after simulating system trajectories instead of using the one-step transition matrix that is difficult to utilize when solving problems with high dimen–sionality. We choose to utilize the post-decision state, which is the state of the system immediately after a decision is made but before the exogenous information processes are realized. This allows the expectation to be moved outside of the maximization operator, altering our value function to the form

$$J^x(s_t^x) = \mathbb{E}\Big\{ \max_{x \in \mathcal{X}(s_{t+1})} \big(C(s_{t+1}, x) + \gamma J^x(s_{t+1}^x)\big) | s_t^x \Big\}.$$

LSTD utilizes a set of basis functions that capture relevant system information to reduce the dimensionality of the state space, providing an approximate solution [27]. Let $\phi_f(s)$, $f \in \mathcal{F}$, denote a basis function where $\mathcal{F}$ is a set of features. The value function approximation is given by

$$\bar{J}^x(s_t^x|\theta) = \sum_{f \in \mathcal{F}} \theta_f \phi_f(s_t^x)$$

wherein $\theta = (\theta_f)_{f \in \mathcal{F}}$ is a column vector of weights with one coefficient for each basis function. We select the number of features to be fewer than the size of the state space so that it is computationally efficient to estimate the value function using basis functions. Choosing an appropriate set of basis functions can be challenging; however, once selected, classical linear regression methods can be used to to generate the θ estimate.

LSTD iteratively updates the value function approximation for a fixed policy and projects it over an infinite horizon. LSTD subtracts the current value of being in a state from the updated value of being in a state at the following iteration to obtain a temporal difference. Powell [27] refers to this process as a batch algorithm that operates by collecting samples of temporal differences and then uses least squares regression to find the best linear fit. LSTD obtains a least squares regression fit so that the sum of the temporal differences over the simulation is as equal to zero as possible.

The algorithm collects a total of K temporal difference sample realizations in each policy evaluation loop where the kth temporal difference is denoted $C(s_{t,k}, X^{\pi}_{\theta}(s_{t,k}))+ \gamma\theta^{\top}\phi(s^{x}_{t,k}) - \theta^{\top}\phi(s^{x}_{t-1,k})$ where $\phi(\cdot)$ is a column vector of basis function evaluations and the policy (i.e., decision function) $X^{\pi}_{\theta}(s_{t,k})$ is defined as indicated below

$$X^{\pi}_{\theta}(s_{t,k}) = \underset{x\in\mathcal{X}(s_t)}{\arg\max}\ C(s_t, x) + \gamma\bar{J}(s^{x}_{t}|\theta).$$

To solve the approximate dynamic program, we need to solve the routing portion of the problem. To generate routing solutions, we utilize two separate algorithms of our own design for comparison. The first, which we call the Index algorithm, is a holistic routing scheme utilizing the quiz problem heuristic to generate routes while the second uses a segmented rollout algorithm approach as a value function estimator to make decisions. The Index algorithm selects the first available CUAV and examines all feasible routes utilizing the quiz problem heuristic to generate a value estimation for each route. The route with the highest value estimation is selected, the decision is recored, and inventory levels are updated according to the decision. The process is then repeated until all available CUAVs have been assigned a route. The decision can then be given to the dynamic programming algorithm. Our Index algorithm is summarized in Algorithm 2.

Algorithm 2 Index algorithm route construction utilizing the quiz problem heuristic

Step 1. **For** all available CUAVs, $v = 1, 2, ..., \min(v_t, V^{crew})$

Step 2a. **For** all possible routes $a = 1, 2, ..., A$

Utilizing the quiz problem heuristic $\frac{\psi_{ma} \min(\sum_{b=1}^{B} R^{max} - r_{tb} + \hat{D}_{t+1}, V^{cap})}{(1-\psi_{ma})}$, generate the value function estimation for each route

End

Step 2b. Assign CUAV route with max value

Step 2c. Record the decision for the vth CUAV and update inventory levels

End

Step 3. Provide routing decision to the appropriate dynamic programming algorithm

Modeling the success of the CUAV route holistically as in Algorithm 2 is ad–vantageous because it emphasizes selecting routes wherein the CUAV has greater probability of survival to deliver supplies in the future. This method will avoid routes that have low probability of success in favor of more easily accessible bases.

For our Rollout algorithm, the value for the gth leg is calculated in Equation 24 where the risk of each segment of the route is individually captured, and the value of the CUAV to deliver supplies in the future is explicitly captured as well. The segmented Rollout algorithm approach is summarized in Algorithm 3.

$$V_g = \frac{\psi_{ma_1} \min(R^{max} - r_{tb_1} + \hat{D}_{t+1}, \frac{V^{cap}}{g}) + ... + \psi_{ma_g} \min(R^{max} - r_{tb_g} + \hat{D}_{t+1}, \frac{V^{cap}}{g}) + \psi_{ma} V^{cap}}{(1 - \psi_{ma})} \tag{24}$$

After a decision is made, the ADP algorithm proceeds by collecting samples. Let Φ_{t-1} and Φ_t be $K \times |\mathcal{F}|$ matrices that consist of rows of basis function evaluations of the sampled post-decision states and C_t as the contribution vector for the sampled events as shown in Equation 25. The sample realization $\hat{\theta}$ is calculated using linear regression for each policy evaluation loop $n = 1, 2, ..., N$.

Algorithm 3 Rollout algorithm route construction utilizing the quiz problem heuristic

Step 1. **For** all available CUAVs, $v = 1, 2, ..., \min(v_t, V^{crew})$

Step 2a. **For** From current location select all $g \in \mathcal{G}$ such that the current location is an end point of g and g is not already part of the path.
Utilizing Equation 24, select the best gth leg and update each expected inventory level of all FOBs visited.
End

Step 2b. Record the decision for the vth CUAV and update inventory levels

End

Step 3. Provide routing decision to the appropriate dynamic programming algorithm

$$\Phi_{t-1} \triangleq \begin{bmatrix} \phi(s^x_{t-1,1})^\top \\ \vdots \\ \phi(s^x_{t-1,K})^\top \end{bmatrix}, \Phi_t \triangleq \begin{bmatrix} \phi(s^x_{t,1})^\top \\ \vdots \\ \phi(s^x_{t,K})^\top \end{bmatrix}, C_t \triangleq \begin{bmatrix} C(s_{t,1}, x_t) \\ \vdots \\ C(s_{t,K}, x_t) \end{bmatrix} \tag{25}$$

We now discuss the monotone adaption to LSTD (MLSTD) as done in Chapter III. Recall that the value function approximation is given by

$$\bar{J}(s|\theta) = \sum_{f \in \mathcal{F}} \theta^n_f \phi_f(s),$$

where the reliance on the parameter vector θ and the associated policy improvement iteration counter n is made more explicit here. The basis for understanding the MLSTD algorithm's ability to be projected to the infinite horizon is accomplished by examining the linear architecture where updates are given by

$$\theta^n \in \arg\min_{\theta} \{\|\theta - \theta^{n-1}\|_2 : \ \bar{J}(s^n_t|\theta) = z^n(s^n_t) \text{ and } \bar{J}(s^n_t|\theta) \text{ is monotone}\},$$

wherein $z^n(s_t^n)$ is the the approximated value smoothed with the value from the preceding iteration. In Chapter III we utilize constrained least-squares optimization as a monotone projection operator to enforce value function monotonicity within the construct of LSTD. For constrained linear least-squares problems, we solve a convex optimization problem of the following form:

$$\min_{\theta} \frac{1}{2}||E\theta - e||_2^2 \; s.t. \; G\theta \leq g. \tag{26}$$

The E matrix is the matrix of sampled observations generated from the LSTD sam–pling (i.e., $(\Phi_{t-1} - \gamma\Phi_t)$) with e being the observed value (i.e., C_t). Herein we explore two possible ways to generate the constraint matrix G with the associated constant g. The first involves generating $K-1$ constraints for the K observations by the following formula:

$$(c_i - c_{i+1})\theta \leq 0, \; i = 1, 2, ..., K-1$$

where c_i is a monotone ordering of sampled observations. Although this constraint matrix enforces monotonicity, it may result in memory issues due to the need to generate $K-1$ constraints for each dimension. Alternatively, Ahrens [2] shows the necessary and sufficient conditions that ensure global monotonicity in one or more dimensions based on the intuition derived from calculus (i.e., a function is monotone if $\frac{d}{dx}f(x) \geq 0$). With each dimension arranged in non-decreasing order and scaled to lie in the interval $[0, 1]$, the necessary and sufficient conditions for non-decreasing monotonicity of the quadratic response surface is given by Equation 27:

$$\theta_f^{linear} + 2\min(\theta_f^{quadratic}, 0) + \sum_{f' \in \mathcal{F}, f \neq f'} \min(\theta_{ff'}^{cross}, 0) \geq 0, \; \forall f \in \mathcal{F} \tag{27}$$

wherein $\theta^{linear}, \theta^{quadratic}$, and θ^{cross} represent all linear, quadratic, and cross product

features, respectively.

The monotone least squares temporal differences (MLSTD) algorithm takes the batch sampling from the LSTD methodology and generates the sample realization $\hat{\theta}$, enforcing monotonicity projection by solving Equation 26 and utilizing Equation 27 to generate monotonicity constraints. The MLSTD pseudo code is summarized in Algorithm 4.

Algorithm 4 Approximate Policy Iteration Using Monotone Least Square Temporal Differences (MLSTD)

Step 0. Initilize θ^0
Step 1. **For** n=1 to N **(Policy Improvement Loop)**
  Step 2. **For** k=1 to K **(Policy Evaluation Loop)**
    a. Generate a random post-decision state, $s^x_{t-1,k}$
    b. Record $\phi(s^x_{t-1,k})$
    c. Simulate transition to next event; obtain pre-decision state $s_{t,k}$
    d. Determine decision $x = X^{\pi}_{\theta^{n-1}}(s_{t,k})$ using the route selection algorithm
    e. Record contribution $C(s_{t,k}, x)$
    f. Record basis function evaluation $\phi(s^x_{t,k})$
    End
  Step 3. Compute θ^n using monotone projection operator Equation 26 and smoothing rule Equation 29
  End

We then smooth in the new observation $\hat{\theta}$ with the previous estimate θ by applying a harmonic stepsize rule to during implementation. The stepsize rule α_n is a function of the outer loop iteration count and is defined below.

$$\alpha_n = \frac{1}{n} \tag{28}$$

Utilizing the harmonic stepsize rule, we update our θ in the following way:

$$\theta^n \leftarrow \theta^{n-1}(1 - \alpha_n) + \hat{\theta}(\alpha_n). \tag{29}$$

Equation 29 shows that the updated θ^n is weighted most heavily by our current estimate, θ^{n-1} and then moved toward our new estimate, $\hat{\theta}$, by an incremental amount proportional to α_n. Initially, greater emphasis is placed on $\hat{\theta}$, but as the number of iterations increases the incremental effect of $\hat{\theta}$ is lessened. Moreover, as the number of iterations increases, any single $\hat{\theta}$ has less influence than the estimate based on information from the first $n-1$ iterations.

Upon obtaining an updated parameter vector θ, we have completed one policy improvement iteration of the algorithm. The parameters N and K are tunable, where N is the number of policy improvement iterations completed and K is the number of policy evaluation iterations completed. After both loops are completed the ADP algorithm has generated a policy and terminates.

4.3 Analysis

Utilizing the Markov decision process (MDP) formulation discussed in Section 4.1, we can find a policy for a five forward operating base (FOB) problem instance. We compare our two proposed route construction methods within the construct of our approximate dynamic programming (ADP) algorithm. Moreover, we run a designed experiment to find the algorithmic and model parameters that yield the best results for our ADP algorithm.

MDP Parameterization.

The military inventory routing problem (MILIRP) is formulated as an infinite horizon MDP wherein a single period represents a 6-hour interval. We assume that during each period the cargo unmanned aerial vehicle (CUAV) completes all mission preparation tasks and performs the assigned mission.

We consider a battalion made of subordinate units that each have a consumption

rate and storage capacity based on the number of personnel on site. Based on a General Dynamics report [13], the expected daily consumption requirements of a platoon is $7,482$ pounds. We round up as a conservative estimate to an 8,000 pound daily average consumption. With four periods in one day, about one ton of supplies per period is required for sustainment. For our testing, we model the stochastic demand using this known historical average $\bar{d}$ and a randomly generated error term, $\hat{\epsilon}$, uniformly distributed on the interval $[-0.5, 0.5]$. We also make the conservative assumption that the FOB has a maximum holding capacity of three times the daily average requirement, totaling 12 tons. We assume that there are no logistical failures limiting the amount of supplies available at the central planner. This assumption is reasonable since the central planner is supplied via fixed wing aircraft from outside the theater of operations.

In order to make the most conservative CUAV carrying capacity choice, we look at carrying capacity at maximum altitude under the assumption that this altitude will minimize the probability of CUAV destruction. We chose a conservative two ton carrying capacity for CUAV resupply because Lockheed Martin's K-MAX CUAV has delivered two tons at $15,000$ feet above ground level (AGL) with more tonnage delivered at lower altitudes [19]. We also chose the number of CUAVs and crews to be eight and four respectively, which mirrors operations for tactical unmanned aircraft system (TUAS) platoons [10]. As the requirements for CUAV resupply increase, we expect to see the number of CUAVs and crews the central planner utilizes to increase. As such, we parameterize the CUAVs and crews as multiples of TUAS platoon ratios. For example, if three TUAS platoons are deployed at the central planner, the number of CUAVs would be 12 and the number of crews 6.

Recall from Section 3.2 that ψ_{ma} denotes the probability of a successful route completion from (to) the support area (SA) to (from) the FOBs along route a on

map m. An intelligence team would assign risk values to each route in the area of operations (i.e., region). This number would account for threats to include but not limited to: weather, enemy action, and mechanical breakdown. Transition between maps can be created by leveraging observed trends specific to the region of interest. These problems influence threat levels, which might include time of the year. For our example, we chose to use $M = 2$ threat maps and parametrized penalizing longer flights with lower probability of success as shown below where the brigade support area is the first row. Feasible routes on the network are restricted by distance and routes are restricted to visit a maximum of 3 FOBs. The problem instance diagram is shown in Figure 2. The probability of successful one-way delivery across the network is shown for each threat map below where the first position in the matrix represents the SA.

$$m_1 = \begin{bmatrix} 1 & .99 & .99 & .95 & .99 & .99 \\ .99 & 1 & .99 & .90 & 0 & 0 \\ .99 & .99 & 1 & .95 & .95 & 0 \\ .95 & .90 & .95 & 1 & .95 & .90 \\ .99 & 0 & .95 & .95 & 1 & .99 \\ .99 & 0 & 0 & .90 & .99 & 1 \end{bmatrix} \quad m_2 = \begin{bmatrix} 1 & .95 & .95 & .90 & .95 & .95 \\ .95 & 1 & .95 & .85 & 0 & 0 \\ .95 & .95 & 1 & .90 & .90 & 0 \\ .90 & .85 & .90 & 1 & .90 & .85 \\ .95 & 0 & .90 & .90 & 1 & .95 \\ .95 & 0 & 0 & .85 & .95 & 1 \end{bmatrix}$$

We vary the transition probability in our experimental design to investigate the effects of region volatility on algorithmic outcome. We select an initial probability of 0.5 to remain in the current threat map to model the current instability of the region.

When a FOB's supply level falls below its predetermined minimum threshold, the FOB is immediately be resupplied via ground convoy to full capacity. When a convoy is sent, the penalty is immediately applied. The penalty represents the increased risk of supply stock out as well as risk to human life inherent in ground convoy operations.

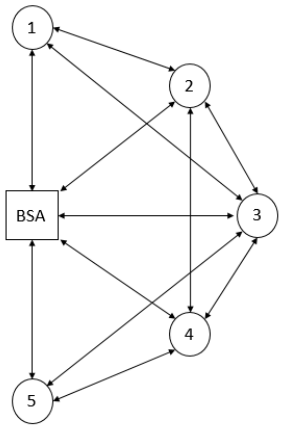

Figure 1. Problem Instance

This penalty would ideally be supplied by a subject matter expert who knows the dangers associated with the specific FOB. This penalty incentivizes CUAV resupply when possible. We set the penalty function $\bar{\tau}$ to be the maximum inventory level R^{max}, which is applied when the inventory level is less than or equal to the R^{min} threshold to be consistent with previous research in Chapter III.

We chose $\lambda = 0.98$ to be a discount factor that balances future needs with current needs. We utilized the above described MDP parameterization to create policies using the Index, and Rollout algorithms within the construct of the LSTD and MLSTD ADP solution techniques.

ADP Policies.

We obtain ADP policies from the monotone least squares temporal differences (MLSTD) algorithm and least squares temporal differences (LSTD) algorithm as explained in Section 3.2. We compare these ADP generated polices to those generated by the Index algorithm, Rollout algorithm, and a myopic strategy. The challenge with both these ADP algorithms is developing basis functions that accurately approximate the optimal value function. These algorithms are employed with the system initialized at full capacity for each FOB, and the resultant policy is compared over a one-month

planning horizon.

We develop ADP policies using our proposed route generation techniques in both LSTD and MLSTD algorithms. Our basis function includes first order effects for current inventory level, CUAVs not deployed, and number of CUAVs deployed. Moreover, we also chose to include the second-order inventory effect. Our proposed Index and Rollout algorithms quickly generate solutions, allowing us to perform a designed experiment with more breadth in a reasonable amount of time.

Baseline Instance.

We selected a representative baseline instance as a reference point for testing the routing algorithm's performance consistent with previous research. The baseline instance has 12 CUAVs, 2 crews, probability of staying in a low threat map of 0.8, probability of staying in a high threat map of 0.2, average period demand of 1 ton, with associated algorithmic features $K = 5000$ and $N = 30$. The algorithm's performance is compared to a myopic policy of direct delivery replenishment to the lowest inventory FOB. The baseline performance is shown in Table 10. The baseline scenario compares MLSTD and LSTD using Index and Rollout algorithms. For the baseline instance, all the proposed solution techniques performed better than the myopic policy. However, both LSTD algorithms performed substantially better than MLSTD algorithms. This illustrates the importance of the loss of the value functions monotonicity on using an approach based on monotone properties when those properties are violated. For this instance, the proposed Index and Rollout generated policies performed better alone than when implemented as part of a larger approximate dynamic programing method. It appears that our proposed Index and Rollout algorithms produce better results, so we expanded our experimental region of search with a designed experiment.

Table 10. Baseline Instance

Algorithm	Value	% Improvement
Myopic	92.87	-
MLSTD-Index	133.92	44.20%
MLSTD-Rollout	139.92	50.66%
LSTD-Index	320.90	245.54%
LSTD-Rollout	304.69	228.08%
Index	339.45	265.51%
Rollout	336.87	262.73%

Experimental design.

We created an experimental design to test the robustness of our ADP algorithm parameters and find the parameter settings that allow our proposed algorithmic ap–proach to achieve the best performance. We focused our response variable on the total value of the system when initialized at full capacity. Baseline results indicate that the value function does not maintain monotonicity, resulting in MLSTD's poor performance. We first run a screening design to confirm this hypothesis. In each ex–perimental run, we simultaneously assess five problem features and four algorithmic features. The five problem features of interest were chosen based on what we thought might have the most effect on the system performance. The problem features we chose to investigate are initial number of CUAVs (V), number of crews available (V^{crew}), probability of staying in a low threat map (Ω), probability of staying in a high threat map (β), and the average demand ($\bar{d}$). The four algorithmic features we chose to experiment on are inner loop iteration count (K), outer loop iteration count (N), a categorical variable indicating whether MLSTD or LSTD is applied, and a categorical variable indicating whether the Index or Rollout algorithm was used for the inner maximization problem within our chosen ADP algorithm. We simulate each resultant policy over one month and compare performance in both amount of supplies delivered by CUAV and number of ground convoys sent.

For analysis, each of the five problem features are considered to be continuous. We chose the vehicle and crew level to be levels associated with deploying two, three, and four TUAS platoons at the SA. This was done under the assumption that as commanders increasingly value CUAV resupply, TUAS platoons will be sent in greater numbers to support brigade operations. The probability of remaining in a low or high threat map, Ω and β respectively, are parameterized to explore how regional volatility affects the value function. The lower value, 0.2, denotes a low chance of transitioning to a different threat map condition. The upper value, 0.8, denotes a high probability of transitioning to a different threat map condition. Demand was chosen to range from current consumption rates (i.e., one ton every six hours) to a considerably larger rate of consumption (i.e., two tons every six hours) to explore how demand schedule increases would affect the system.

The four algorithmic features were chosen to best explore the experimental space. The inner loop count was set to a low of $5,000$ and a high of $15,000$ based on initial testing. The center run is the midpoint of the upper and lower bounds and allows us to determine if our response variable demonstrates nonlinearity. The outer loop iteration counter was similarly chosen, allowing for a large upper bound to achieve the most accurate value function approximation for the basis functions we selected. Table 11 shows the problem and algorithmic settings for our experimental design.

We first implemented a 2^{9-4} resolution IV fractional factorial screening design with four center runs totaling 36 runs. After screening two variables, we imple–mented a 2^{7-1} resolution VI fractional factorial design with four center runs totaling 68 runs. In a resolution VI design, all first- and second-order effects are free from being aliased with other first- and second-order interactions. For each design run, a myopic, MLSTD, and LSTD policy is determined. Recall that an ADP policy utilizes θ-coefficients that correspond to selected basis functions. After the θ-coefficients are

Table 11. Factorial Screening Design Settings

Description	Factor	Low	Center	High
Initial number of CUAVs	V	4	8	12
Number of crews	V^{crew}	2	4	6
Probability of remaining low threat	Ω	0.2	0.5	0.8
Probability of remaining high threat	β	0.2	0.5	0.8
Average Demand	$\bar{d}$	1	1.5	2
Number of inner loops	K	5000	10000	15000
Number of outer loops	N	10	20	30
Algorithm	Strategy	MLSTD	-	LSTD
Routing strategy	Heuristic	Rollout	-	Index

determined, we compute the resulting MLSTD and LSTD policies and compare them to the myopic policy over a one-month planning horizon.

Results.

Screening led us to explore five problem features while reducing the algorithmic features to two: a categorical variable where 'Myopic' denotes myopic strategy and 'LSTD' denotes an LSTD solution strategy, and a categorical variable where 'Rollout' denotes using our proposed Rollout algorithm and 'Index' indicates using our pro–posed Index algorithm for the inner maximization problem within our chosen ADP algorithm. The experimental design settings are shown in Table 12. Interestingly, the policy value was invariant to outer and inner loop count. We found that LSTD would quickly converge to the estimated policy. Moreover, although subsequent iterations would update value function weights θ^n for iteration n, the generated policy would be invariant to the resulting change.

Tables 13 and 14 show the results from the experiment, with two additional columns, indicating the value of using our proposed Index and Rollout algorithms with the corresponding design settings. Although the values shown in the rightmost three columns are mostly negative, which reflect the large penalty function we se–lected, the goal is maximization. These values reflect the overall performance of the

Table 12. Factorial Parameter Design Settings

Description	Factor	Low	Center	High
Initial number of CUAVs	V	4	8	12
Number of crews	V^{crew}	2	4	6
Probability of remaining low threat	Ω	0.2	0.5	0.8
Probability of remaining high threat	β	0.2	0.5	0.8
Average Demand	$\bar{d}$	1	1.5	2
Algorithm	Strategy	Myopic	-	LSTD
Routing strategy	Heuristic	Rollout	-	Index

selected algorithm on the specific problem instance where more is better. The 'Value ADP' column reflects the value of either the myopic policy or ADP policy as shown in the 'LSTD' column. Although the 'Index' and 'Rollout' columns weren't used in the experimental design, they are included here for reference. Throughout the experimental region both Index and Rollout algorithms performed better than the ADP technique. Since both proposed heuristic search algorithms are far less com–putationally intensive, this is a great result. Although these other two techniques outperformed the ADP technique, the LSTD policy still outperformed the myopic policy, performing 22% better on average. The problem features for which the LSTD algorithm performed best include when probability of staying in a low threat map, initial number of CUAVs, and number of crews were at their high levels, and when demand and probability of staying in a high threat map were at their low levels. The algorithmic features that produced the best values are when LSTD and the in–dex heuristic is used. This intuitive result is caused by LSTD's ability to send more CUAVs in the relatively safer operating condition.

We next created a regression metamodel to analyze the parameter design effects with more statistical rigor. A stepwise regression procedure yields factors that pro–duce a significant relationship and pass the lack of fit test. Both the resulting models include significant first- and second-order terms and performs very well in terms of prediction ability with an adjusted R^2 over 0.99. The residuals do not show signs of

Table 13. Experimental Results

Run	V	K	Ω	β	$\bar{d}$	K	N	Heuristic	LSTD	Value ADP	Value Index	Value Rollout
1	4	2	0.2	0.2	1	5000	10	Rollout	LSTD	-360.88	-294.88	-300.89
2	4	2	0.2	0.2	1	5000	10	Index	Myopic	-361.47	-290.68	-279.92
3	4	2	0.2	0.2	2	5000	10	Rollout	Myopic	-909.25	-835.48	-836.84
4	4	2	0.2	0.2	2	5000	10	Index	LSTD	-900.69	-825.37	-840.32
5	4	2	0.2	0.8	1	5000	10	Rollout	Myopic	-398.07	-377.93	-369.56
6	4	2	0.2	0.8	1	5000	10	Index	LSTD	-402.15	-370.17	-359.81
7	4	2	0.2	0.8	2	5000	10	Rollout	LSTD	-945.63	-908.55	-916.15
8	4	2	0.2	0.8	2	5000	10	Index	Myopic	-945.57	-905.48	-921.54
9	4	2	0.8	0.2	1	5000	10	Rollout	Myopic	-309.95	-143.68	-111.14
10	4	2	0.8	0.2	1	5000	10	Index	LSTD	-259.88	-137.63	-117.37
11	4	2	0.8	0.2	2	5000	10	Rollout	LSTD	-864.94	-677.76	-672.92
12	4	2	0.8	0.2	2	5000	10	Index	Myopic	-766.78	-670.49	-684.23
13	4	2	0.8	0.8	1	5000	10	Rollout	LSTD	-355.57	-305.15	-297.67
14	4	2	0.8	0.8	1	5000	10	Index	Myopic	-372.84	-296.20	-295.16
15	4	2	0.8	0.8	2	5000	10	Rollout	Myopic	-907.89	-835.31	-842.03
16	4	2	0.8	0.8	2	5000	10	Index	LSTD	-871.50	-845.01	-847.43
17	4	6	0.2	0.2	1	5000	10	Rollout	Myopic	-373.17	-321.09	-322.61
18	4	6	0.2	0.2	1	5000	10	Index	LSTD	-384.29	-334.04	-324.45
19	4	6	0.2	0.2	2	5000	10	Rollout	LSTD	-901.99	-845.01	-845.45
20	4	6	0.2	0.2	2	5000	10	Index	Myopic	-919.79	-854.91	-836.60
21	4	6	0.2	0.8	1	5000	10	Rollout	LSTD	-413.65	-390.09	-382.77
22	4	6	0.2	0.8	1	5000	10	Index	Myopic	-402.62	-384.10	-379.05
23	4	6	0.2	0.8	2	5000	10	Rollout	Myopic	-967.82	-907.74	-907.39
24	4	6	0.2	0.8	2	5000	10	Index	LSTD	-939.87	-906.22	-920.12
25	4	6	0.8	0.2	1	5000	10	Rollout	LSTD	-304.37	-200.31	-197.95
26	4	6	0.8	0.2	1	5000	10	Index	Myopic	-290.23	-186.00	-201.95
27	4	6	0.8	0.2	2	5000	10	Rollout	Myopic	-838.33	-712.10	-701.36
28	4	6	0.8	0.2	2	5000	10	Index	LSTD	-780.57	-699.45	-697.93
29	4	6	0.8	0.8	1	5000	10	Rollout	Myopic	-378.40	-322.64	-345.77
30	4	6	0.8	0.8	1	5000	10	Index	LSTD	-368.85	-328.83	-342.83
31	4	6	0.8	0.8	2	5000	10	Rollout	LSTD	-901.54	-845.60	-850.93
32	4	6	0.8	0.8	2	5000	10	Index	Myopic	-914.59	-851.85	-846.60
33	12	2	0.2	0.2	1	5000	10	Rollout	Myopic	-30.35	209.84	177.90
34	12	2	0.2	0.2	1	5000	10	Index	LSTD	-61.07	199.05	187.68

Table 14. Experimental Results Continued

Run	V	K	Ω	β	$\bar{d}$	K	N	Heuristic	LSTD	Value ADP	Value Index	Value Rollout
35	12	2	0.2	0.2	2	5000	10	Rollout	LSTD	-608.02	-383.42	-394.37
36	12	2	0.2	0.2	2	5000	10	Index	Myopic	-576.61	-370.62	-387.07
37	12	2	0.2	0.8	1	5000	10	Rollout	LSTD	-186.41	17.29	3.68
38	12	2	0.2	0.8	1	5000	10	Index	Myopic	-169.46	19.62	17.08
39	12	2	0.2	0.8	2	5000	10	Rollout	Myopic	-678.91	-549.37	-575.04
40	12	2	0.2	0.8	2	5000	10	Index	LSTD	-708.09	-559.50	-562.55
41	12	2	0.8	0.2	1	5000	10	Rollout	LSTD	134.50	339.14	339.39
42	12	2	0.8	0.2	1	5000	10	Index	Myopic	133.69	334.21	342.77
43	12	2	0.8	0.2	2	5000	10	Rollout	Myopic	-474.00	-251.05	-248.63
44	12	2	0.8	0.2	2	5000	10	Index	LSTD	-416.06	-230.72	-246.45
45	12	2	0.8	0.8	1	5000	10	Rollout	Myopic	-72.08	189.05	178.08
46	12	2	0.8	0.8	1	5000	10	Index	LSTD	-69.35	186.24	198.37
47	12	2	0.8	0.8	2	5000	10	Rollout	LSTD	-593.42	-395.39	-395.53
48	12	2	0.8	0.8	2	5000	10	Index	Myopic	-574.48	-409.82	-411.85
49	12	6	0.2	0.2	1	5000	10	Rollout	LSTD	-75.97	83.42	62.41
50	12	6	0.2	0.2	1	5000	10	Index	Myopic	-84.50	93.13	49.15
51	12	6	0.2	0.2	2	5000	10	Rollout	Myopic	-518.60	-375.64	-459.75
52	12	6	0.2	0.2	2	5000	10	Index	LSTD	-504.13	-385.69	-441.10
53	12	6	0.2	0.8	1	5000	10	Rollout	Myopic	-188.56	-72.29	-91.85
54	12	6	0.2	0.8	1	5000	10	Index	LSTD	-180.91	-86.61	-100.11
55	12	6	0.2	0.8	2	5000	10	Rollout	LSTD	-638.60	-552.09	-603.76
56	12	6	0.2	0.8	2	5000	10	Index	Myopic	-635.12	-557.61	-600.36
57	12	6	0.8	0.2	1	5000	10	Rollout	Myopic	116.01	332.17	350.80
58	12	6	0.8	0.2	1	5000	10	Index	LSTD	133.58	330.99	340.10
59	12	6	0.8	0.2	2	5000	10	Rollout	LSTD	-309.74	-48.14	-155.51
60	12	6	0.8	0.2	2	5000	10	Index	Myopic	-331.65	-15.64	-179.54
61	12	6	0.8	0.8	1	5000	10	Rollout	LSTD	-90.79	50.09	56.93
62	12	6	0.8	0.8	1	5000	10	Index	Myopic	-57.97	51.53	32.61
63	12	6	0.8	0.8	2	5000	10	Rollout	Myopic	-538.43	-392.74	-450.78
64	12	6	0.8	0.8	2	5000	10	Index	LSTD	-532.20	-371.49	-455.68
65	8	4	0.5	0.5	1.5	5000	10	Rollout	Myopic	-441.89	-370.25	-390.17
66	8	4	0.5	0.5	1.5	5000	10	Rollout	LSTD	-447.24	-368.20	-361.02
67	8	4	0.5	0.5	1.5	5000	10	Index	Myopic	-468.18	-357.78	-362.80
68	8	4	0.5	0.5	1.5	5000	10	Index	LSTD	-467.43	-361.20	-362.97

heteroscedasticity, and the residual by predicted plot does not raise concerns.

We consider variables with an F-test p-value less than 0.05 significant. Significant variables are summarized in Table 15. With this criterion, the experimental variables initial number of CUAVs, probability of staying in a low threat condition, probability of staying in a high threat condition and demand are significant in both first- and second-order terms. Number of crews is significant in second-order interaction. Using inner loop and outer loop iteration count are not significant even when higher order effects are considered. This is due to the fast convergence of the ADP algorithm to the resulting ADP-generated policy.

According to our metamodel, the value function is maximized when the probability of staying in a high threat map of 0.2, the probability of staying in a low threat map of 0.8, $5,000$ inner loops, 10 outer loops, 12 CUAVs, 6 crews, and 1 ton of average demand. These results follow intuition and confirm that LSTD is significantly better than the myopic strategy.

4.4 Conclusions

The intent of this research is to demonstrate the efficacy of the proposed Index and Rollout algoritms on the military inventory routing problem (MILIRP) and compare algorithmic performance that can improve upon simple strategies. Development of an expanded Markov decision process (MDP) model of the MILIRP enables examination of the disparate conditions the military faces in hostile environments.

The efficient utilization of CUAV assets for resupply is an important issue in military applications. The varying threats that the military faces in combative environments and the availability of distribution assets all inhibit successful distribution of supplies from the support area (SA) to the forward operating bases (FOBs). Moreover, insurgent use of improvised explosive devices (IEDs) affects truck mobility and

Table 15. Factors Influencing CUAV Resupply

Variable	Sum of Squares	F Test	% Contribution
V	1764062.1	$< .0001$	27.41%
V^{crew}	2087.6	0.053	0.03%
Ω	174682.2	$< .0001$	2.71%
β	180820.6	$< .0001$	2.81%
$\bar{d}$	4217262.7	$< .0001$	65.54%
$V * V^{crew}$	6797.2	0.0008	0.11%
$V * \Omega$	24894.5	$< .0001$	0.39%
$V * \beta$	34312	$< .0001$	0.53%
$V * \bar{d}$	6641	0.0009	0.10%
$V^{crew} * \bar{d}$	9317.6	0.0001	0.14%
$\Omega * \beta$	13821.5	$< .0001$	0.21%

has been successful in disrupting replenishment procedures [25]. Since 2012 when the K-MAX successfully deployed to Afghanistan [15], CUAVs have been of increasing interest worldwide [14]. This chapter provides unique insight into efficient utilization of CUAV resupply in volatile environments. CUAV benefits include: better performance in adverse weather conditions, higher flight ceilings, and no escort requirement restrictions. These advantages mean a lower probability of vehicle destruction. The most important benefit of CUAVs is their ability to save lives by alleviating manned ground convoy resupply requirements. Although CUAVs do not yet have the ability to completely handle FOB supply requirements, each successful CUAV delivery means less men and women exposed to enemy threats to include IEDs.

We examine two ADP algorithms and conclude that monotonicity of the value function does not exist in the current region of operation and therefore ADP techniques based on this assumption perform poorly, although better than a myopic policy. We formulate an MDP model of the MILIRP and determine a policy on realistic instances in order to compare two approximate dynamic programming algorithms. We design a representative baseline instance and conclude that both the ADP algorithms and heuristic search techniques perform better than simple strategies. We test our

approach with an experimental design and empirically find the design settings that maximize the ADP algorithmic performance. Moreover, we conclude that there is no statistically significant difference between utilizing Index or Rollout techniques within the ADP algorithm despite the Rollout algorithm's more accurate representation of periodic risk. Additionally, we conclude both Index and Rollout algorithms perform better on this problem in terms of computational requirements and policy value than the proposed ADP techniques. This is likely due to the heuristic's strength in avoiding routes with low probability of success in favor of more easily assessable ones coupled with the inability of the linear model to capture the nuances of the complex problem. Further exploration of ADP techniques that exploit the special structure may yield improved results. However, direct applications of the Index and Rollout algorithms performed better for the MILIRP instances considered. Due to their demonstrated success, we conclude that any further ADP work should also endeavor to incorporate the Index and Rollout algorithms presented here.

4.5 Acknowledgments

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

V. Utilizing Heuristics Within a Least Squares Temporal Differences Algorithm to Solve a Multiclass Stochastic Inventory Routing Problem with Vehicle Loss

This chapter develops an extended Markov decision process (MDP) model to examine the performance of two approximate dynamic programming (ADP) algorithms for determining solutions to the military inventory routing problem (MILIRP). This formulation explicitly accounts for the possible loss of vehicles due to enemy activity and generates resupply polices to maximize the supplies delivered by cargo unmanned aerial vehicles (CUAVs) in order to reduce the need for casualty-prone convoy operations. A large baseline instance is examined and it is found that the proposed least squares temporal differences using our Index algorithm (LSTD-Index) and least squares temporal differences using our Rollout algorithm (LSTD-Rollout) algorithms presented herein perform 21% better than a myopic policy. Inventory routing is an inherently difficult logistical problem that requires vehicle routing and inventory management decisions. This chapter focuses on improving upon existing work by adding multiple supply classes to the MILIRP as researched in Chapter IV.

The MILIRP is a stochastic inventory routing problem that utilizes ground and aerial assets to deliver supplies to forward operating bases (FOBs) in an austere combat environment. Due to the difficult nature of FOB resupply in a combat environment, aerial assets can be destroyed thereby permanently impacting the ability of the central planner to resupply the FOBs. Military planners must consider the timing, routing, threat level, and supply configuration of distribution assets when executing resupply missions.

In this chapter, a central planner must decide how many fully loaded CUAVs to dispatch to fill demand across multiple FOBs with multiple supply classes. To fill this demand, multiple delivery across a finite set of feasible CUAV routes is allowed.

An MDP model to maximize the supplies delivered by aerial assets is developed and ADP solution techniques as developed in Chapter IV to solve instances of this larger problem class are implemented. These solution techniques include an LSTD algorithm with an embedded heuristic to solve this MILIRP, which exhibits high-dimensionality in the state, action, and outcome spaces. A notional, representative planning scenario based on an austere combat environment where convoy resupply may be difficult (e.g., as in Afghanistan) is constructed. The testing and application of these algorithms further strengthen the efficacy of the proposed solution methodology.

Replenishment during combat operations includes difficult, deliberate, and time-sen–sitive operations conducted to resupply forward companies with essential supplies to sustain the pace of operations [12]. This chapter's formulation includes multiple sup–ply classes to improve model realism. Thus, the Army's definition of supply classes deserves more attention. The U.S. Army defines ten different supply classes gener–ally defined in Table 16 [11]. The implementation of supply classes within our model greatly increases the complexity of the problem and will be addressed more throughly in Section 5.1.

The remainder of this chapter is organized as follows. Section 5.1 introduces the MDP formulation of the MILRIP. Section 5.2 presents the ADP solution approach.

Table 16. Definition of U.S. Army supply classes

U.S. Army Supply Classes	
I.	Subsistence
II.	Clothing/individual equipment
III.	Fuels/lubricants/fluids
IV.	Construction materials
V.	Ammunition
VI.	Personal demand items
VII.	Major end items(tanks, vehicles etc.)
VIII.	Medical supplies
IX.	Repair parts
X.	Non-military programs material

Section 5.3 presents the computational results and analysis. Section 5.4 provides conclusions for this research.

5.1 Markov Decision Process Model

This section describes the Markov decision process (MDP) model formulation of the military inventory routing problem (MILIRP). This MDP methodology is a multiclass extension of previous research done in Chapter IV. The objective of the multiclass MILIRP is to determine the optimal resupply policy for multiple forward operating bases (FOBs) to maintain inventory levels across multiple supply classes. The contribution function rewards supplies delivered to each FOB until it reaches the FOB's maximum holding capacity, after which additional supplies delivered yield no value. We assume demand has a known historical average with some variability, mod–eled as an independent and identically distributed error term. Moreover, we assume the inventory level at each FOB is known at the start of each period. Additionally, we assume that no other external event (e.g., enemy action, fire, expiration of supplies) other than demand causes inventory loss.

The central planner utilizes V identical cargo unmanned aerial vehicles (CUAVs) to resupply the FOBs. Each CUAV has an identical load capacity of V^{cap} tons. We denote the feasible paths for CUAV resupply by constructing a finite set of routes $\mathcal{A}$. When multiple FOBs are visited along the CUAV route, it is assumed that the load is evenly distributed among each FOB. We denote the set of supply classes for each FOB as $\mathcal{C}$. V^{ac} is the amount of supplies of class $c \in \mathcal{C}$ delivered to each FOB along route $a \in \mathcal{A}$. V^{ac} is assumed to be a fixed vehicle loadout proportioned for each class according to historic demand. FOBs require $\hat{D}_{tc}$ tons of supplies of class c per time period t, a stochastic demand with a mean demand $\bar{d}_c$, and an independent and identically distributed exogenous error term $\hat{\epsilon}$. FOBs also have a finite maximum

holding quantity for each class of supply R_c^{max}. A FOB's total maximum holding quantity is defined as $R^{max} \triangleq \sum_{c \in \mathcal{C}} R_c^{max}$. The minimum holding quantity for each FOB is defined for each class $c \in \mathcal{C}$ as R_c^{min}.

Given the dangers inherent in a combat environment, there is potential for delivery failure due to extrinsic uncontrollable factors (e.g., enemy action, mechanical failure, extreme weather conditions). These threat conditions determine the probability of a CUAV being destroyed. A set of M threat maps models the periodic changes in risk throughout the central planner's area of operations. Under threat map $m = 1, 2, ..., M$, the parameter ψ_{ma} denotes the probability of a successful trip from (to) the support area (SA) to (from) the FOB on the specified route $a \in \mathcal{A}$. A CUAV may be destroyed on its way to a FOB or after delivering supplies on the return route back to the depot at the SA.

We now continue by describing the MDP model formulation of the MILIRP. With respect to a conventional inventory routing formulation, CUAVs are vehicles, FOBs are customers, and the centralized planner is the supplier. Tables 17 and 18 located at the end of this section provide a summary of the notation.

The MILIRP is formulated as an infinite horizon Markov decision problem wherein an inventory routing decision is made at each decision epoch $t \in \mathcal{T} = \{1, 2, ...\}$. It is assumed a CUAV refuels, resupplies, receives maintenance, travels the specified route, unloads, and returns to the SA during each time period. In this research, routes are limited to only visiting a maximum of three FOBs. Moreover, the routes are constructed within range limitations and are serviceable in one time period. Current CUAV limitations validate this assumption [19].

The state space includes three components: the inventory level at the FOBs, the number of operational CUAVs, and the threat map index number. The inventory for all FOBs is defined as r_t, where $r_t = (r_{t11}, r_{t21}, ..., r_{t21}, r_{t22}, ...r_{tBC})$ and $r_{tbc} \in$

$(0, R_c^{max})$ is the number of tons of supplies at FOB $b \in \{1, 2, ..., B\}$ for class $c \in \mathcal{C}$ at time t. Moreover, R^{max} is the maximum inventory capacity for the FOBs, and $R_c^{min} \in (0, R^{max})$ is the minimum threshold inventory level that must be exceeded (i.e., the safety stock level). If $\exists\ c \in \mathcal{C} | r_{tbc} \leq R_c^{min}$ then resupply via convoy ground lines of communication (GLOC) is required. The number of operational CUAVs able to perform resupply operations at time t is defined as v_t. The threat map index number at time t is defined as $m_t \in \{1, 2, ..., M\}$. The threat map impacts the flight risk associated with successfully completing sorties between the FOBs and the SA. The threat map information m_t is available at time t. The threat map information m_{t+1} available at time $t+1$ is conditioned on m_t and is unknown at time t. Utilizing these components, we define $s_t = \left(r_t, v_t, m_t\right) \in \mathcal{S}$ as the state of the system at time t, where $\mathcal{S}$ is the set of all possible states. Moreover, due to route construction, each CUAV begins and ends each day at the SA.

The decision and transition spaces are similar to previous research in Chapter III and Chapter IV. The main difference in the transition space from previous efforts is denoted in the inventory transition functions. We let $\mathcal{X}(s_t)$ be the set of all feasible ac–tions when the system is in state s_t. Let $x_t = (x_t^d, x_t^{GLOC}) \in \mathcal{X}(s_t)$ denote an inventory routing decision wherein $x_t^d = (x_{t1}^d, x_{t2}^d, ..., x_{t|\mathcal{A}|}^d)$, $x_t^{GLOC} = (x_{t1}^{GLOC}, x_{t2}^{GLOC}, ..., x_{tB}^{GLOC})$, $x_{ta}^d \in \mathbb{N}^0$ denotes the number of fully loaded CUAVs dispatched to resupply the FOBs along route $a \in \mathcal{A}$ and $x_{tb}^{GLOC} \in \{0, 1\}$ denotes whether a ground convoy is dispatched to resupply FOB b, which results in its inventory level increasing to capacity for all supply classes. Only CUAV resupply is available for a FOB if the inventory level is greater than the safety stock threshold for every supply class (i.e., $\forall c \in \mathcal{C} | r_{tbc} > R_c^{min}$). Only GLOC resupply is available if the inventory level is less than or equal to the safety stock threshold for any supply class (i.e., $\exists c \in \mathcal{C} | r_{tbc} \leq R_c^{min}$). Two constraints impact the CUAV routing decision: first, the number of CUAVs deployed cannot ex–

ceed the number of operational CUAVs (i.e., $x_t^d \leq v_t$); second, the number of CUAVs deployed cannot exceed the number of crews available (i.e., $x_t^d \leq V^{crew}$). We assume that each CUAV carries a maximum capacity load of V^{cap} and divides its load equally among all the FOBs visited along the route. The policy (i.e., decision function) $X^\pi(s_t)$ returns a decision $x_t \in \mathcal{X}(s_t)$ as a function of the system state $s_t \in \mathcal{S}$. After a routing decision is made, delivery is performed within one time period.

Transition probabilities are defined for each dimension of the state space to include the inventory level at each FOB, number of remaining CUAVs, and threat map. Inventory transitions are based on the routing decision x_t and the current state of the system s_t. When CUAVs are routed to the FOBs there are two possible outcomes: first, a CUAV may successfully travel along its route and return to the SA; second, a CUAV may be destroyed along its route before returning to the SA. Let ψ_{ma}^{SSSS} denote the probability of a successful round trip delivery along a three FOB route a, ψ_{ma}^{SSSF} denote the probability of a successful three leg FOB delivery along a three FOB route a, ψ_{ma}^{SSFF} denote the probabilities of a successful two leg FOB delivery along a three FOB route a, ψ_{ma}^{SFFF} denote the probability of a successful one leg FOB delivery along a three FOB route a, and ψ_{ma}^{FFFF} denote the probability of a failed delivery along a three FOB route a for a single CUAV routed to resupply FOBs during the threat conditions of map $m = 1, 2, ..., M$. Outcome probabilities for one and two FOB routes are similarly defined. Since we are interested in a particular outcome of a routing decision, we proceed by defining the binomial marginal distributions for each outcome type (i.e., SSSS, SSSF, SSFF, SFFF, FFFF, SSS, SSF, SFF, FFF, SS, SF, FF). With the assumption that each outcome of a resupply mission to a FOB is independent of other missions and recalling that x_t includes the decision to route x_t^d CUAVs to the FOB (each carrying a full supply load), we let $\hat{Z}_{t+1,b}^{SSSS}(\psi_{ma}^{SSSS}, x_{ta}^d)$ denote the binomial random variable with parameters ψ_{ma}^{SSSS} and x_{ta}^d that indicates

the number of successful round trip CUAV deliveries to FOB b in route a during time interval $[t, t+1)$ on map m. Let all other outcomes (i.e., SSSS, SSSF, SSFF, SFFF, FFFF, SSS, SSF, SFF, FFF, SS, SF, FF) be similarly defined. For compactness, we refer to the set of random variables that indicate resupply mission outcomes for each FOB as follows:

$$\hat{Z}_{t+1,b} = \left\{ \hat{Z}^{SSSS}_{t+1,b}, \hat{Z}^{SSSF}_{t+1,b}, ..., \hat{Z}^{SS}_{t+1,b}, \hat{Z}^{SF}_{t+1,b}, \hat{Z}^{FF}_{t+1,b} \right\}. \tag{30}$$

Moreover, the outcome denoting the number of CUAVs that are destroyed in time interval $[t, t+1)$ is the summation of all failures as shown in Equation 33 where we drop the FOB distinction b, thereby denoting the total number of failures across all FOBs as demonstrated in Equation 31.

$$\hat{Z}^{SSSF}_{t+1} = \sum_{b=1}^{B} \hat{Z}^{SSSF}_{t+1,b}. \tag{31}$$

The addition of multiple supply classes requires an update on inventory transitions from previous efforts in Chapter IV. The inventory level at the FOBs are limited by the maximum holding quantity R^{max}. Moreover, if the FOB supply level is less than or equal to a safety stock threshold, R^{min}_c, the FOB must be fully resupplied via ground convoy. Equation 32 is the inventory transition function explicitly updated for each supply class in the FOB.

$$r_{t+1,b,c} = \begin{cases} R^{max}_c & \text{if } x^{GLOC}_{tb} = 1 \\ \min\left(r_{tbc} + V^{ac}(\hat{Z}_{t+1,b}) - \hat{D}_{t+1,c}, R^{max}_c \right) & \text{if } \sum\limits_{a \in \mathcal{A}} x^{d}_{tab} > 0 \\ r_{tbc} - \hat{D}_{t+1,c} & \text{otherwise.} \end{cases} \tag{32}$$

In the first case, convoy resupply is selected, and the FOB is resupplied to capacity. In the second and third cases, the FOB inventory level changes according to supplies received and realized demand. The minimization in the second case enforces the FOB

capacity constraint.

CUAV transition is contingent on the number of CUAVs that fail to return to the SA after attempting to travel their route. The number of CUAVs transition according to Equation 33.

$$v_{t+1} = v_t - (\hat{Z}^{SSSF}_{t+1} + \hat{Z}^{SSFF}_{t+1} + \hat{Z}^{SFFF}_{t+1} + \hat{Z}^{FFFF}_{t+1} + \hat{Z}^{SSF}_{t+1} + \hat{Z}^{SFF}_{t+1} + \hat{Z}^{FFF}_{t+1} + \hat{Z}^{SF}_{t+1} + \hat{Z}^{FF}_{t+1}) \tag{33}$$

The map transition function is a representation of the dynamic aspect of the combat environment in which the military resuppliers operate. The set of all maps captures the varying threat level of the operational environment. The map transi–tions are representative of the changing environment. For relatively stable combat conditions, the map transition probability would be relatively low. More volatile combat environments yield a relatively higher map transition probability. The cen–tral supplier's intelligence teams gather information on threat conditions based on information on enemy activities.

The contribution function rewards the system based on the amount of supplies CUAVs deliver to the FOBs. The amount of supplies delivered is bounded above by the maximum inventory quantity at each FOB, constraining any excess supplies delivered from affecting the system behavior. An immediate penalty, $\bar{\tau}$, is applied if the FOB's inventory level is less than or equal to a safety stock threshold R_c^{min} due to the human risk associated with ground convoy resupply. The below-threshold penalty function for each FOB

$$\tau(r_{bc}) = \begin{cases} 0 & \text{if } r_{bc} > R_c^{min} \; \forall \; c \in \mathcal{C}, \\ \bar{\tau} & \text{otherwise} \end{cases} \tag{34}$$

allows the application of a penalty that can capture the difficulty of resupplying FOBs

via ground convoy. We chose to set the penalty at the max inventory level at the FOB, R^{max}, so as to encourage CUAV resupply based on findings from previous research in Chapter III. We present our contribution function in Equation 35.

$$C(s_t, x_t) = \mathbb{E}\Big\{ \sum_{b=1}^{B} \sum_{c \in \mathcal{C}} \min \Big(R_c^{max} - r_{tbc} + \hat{D}_{t+1,c}, V^{ac} \cdot (\hat{Z}_{t+1,b}) \Big) - \tau(r_{tbc}) \Big| s_t, x_t \Big\} \quad (35)$$

The amount of supplies successfully delivered to the FOBs determines the single-pe–riod contribution (i.e., reward). However, the system is not rewarded for excess supplies (i.e., FOBs cannot take delivery of supplies in excess of their capacity).

The objective of this MDP is to maximize the expected total discounted reward over an infinite horizon. By definition, the transitions are Markovian. All decisions made at time t depend only on the current state of the system. To obtain the policy that maximizes the expected total discounted reward, Bellman's optimality equation shown in Equation 36 is solved.

$$J(s_t) = \max_{x \in \mathcal{X}(s_t)} \Big(C(s_t, x) + \lambda \mathbb{E}\{ J(s_{t+1}) | s_t, x \} \Big) \quad (36)$$

The value of being in state s_t results from choosing the action that maximizes the expected immediate reward and the discounted expected future value of the system at time $t + 1$. The parameter $\lambda \in [0, 1)$ denotes the discount factor. Using this MDP formulation, we can apply our extended approximate dynamic programming algorithm to obtain policies for resupplying the FOB via CUAVs.

5.2 Solution Methodology

This section introduces the extention of the Index and Rollout algorithms based on the quiz problem heuristic that balance risk with potential rewards to more effectively solve the military inventory routing problem (MILIRP), a variant of the stochastic

Table 17. Table of Notation

a	=	feasible route counter
b	=	forward operating base counter
c	=	supply class counter
C	=	contribution function
C_t	=	contribution vector for the sampled events
d	=	daily FOB demand
E	=	matrix of sampled observations generated from LSTD sampling
F	=	failure to deliver supplies
g	=	route segment or leg of route a
G	=	the total number of route segments or legs of route a
I	=	indicator variable
J	=	total expected reward (cost-to-go) function
k	=	policy evaluation loop counter
K	=	number of policy evaluation loops
M	=	number of threat maps
n	=	policy improvement loop counter
N	=	number of policy improvement loops
r	=	supplies on hand
R^{min}	=	supply threshold
R^{max}	=	FOB holding capacity
s	=	state of system
t	=	time epoch
v	=	current number of CUAVs
V	=	number of inital CUAVs

Table 18. Table of Notation Continued

V^{ac}	=	the amount of supplies for each supply class c delivered to each FOB along route a
V^{cap}	=	CUAV holding capacity
V^{crew}	=	number of crews
w	=	exogenous information process
x	=	actions
Z	=	set of random variables corresponding to the number of possible SS, SF, and F events
$\mathcal{A}$	=	set of all feasible routes
$\mathcal{C}$	=	set of all supply classes
$\mathcal{F}$	=	set of basis function features
$\mathcal{G}$	=	set of all feasible legs
$\mathcal{S}$	=	state space
$\mathcal{T}$	=	set of time epochs
$\mathcal{X}$	=	action space
α	=	stepsize
β	=	probability of remaining in the high threat map
θ	=	vector of weights
λ	=	discount factor
π	=	policy
τ	=	penalty cost
ψ_{ma}	=	one-way probability a CUAV successfully reaches its destination on route a
ϕ	=	basis function
Φ	=	matrix of fixed basis functions
Ω	=	probability of remaining in the low threat map

inventory routing problem (IRP). These algorithms can be applied individually or within the construct of approximate dynamic programming. We extend this approach, as utilized in Chapter IV, to the multiclass formulation. Moreover, we also present the least squares temporal differences (LSTD) algorithm that will utilize the Index and Rollout algorithm.

ADP Formulation.

Our LSTD-Index and LSTD-Rollout approximate dynamic programming (ADP) algorithms utilize an approximate policy iteration (API) framework with an LSTD value function approximation scheme. API follows the exact policy iteration algo–rithm framework. Our API algorithm approximates and updates the value function after simulating system trajectories because using the one-step transition matrix is difficult to utilize when solving problems with high dimensionality. We utilize the post-decision state, which is the state of the system immediately after a decision is made but before the exogenous information processes are realized. This allows the expectation to be moved outside of the maximization operator, altering our value function to the form

$$J^x(s_t^x) = \mathbb{E}\Big\{ \max_{x \in \mathcal{X}(s_{t+1})} \big(C(s_{t+1}, x) + \gamma J^x(s_{t+1}^x)\big) | s_t^x \Big\}.$$

LSTD reduces the dimensionality of the state space by utilizing a set of basis func–tions that capture relevant information in the system thus providing an approximate solution [27]. Let $\phi_f(s)$, $f \in \mathcal{F}$, denote a basis function where $\mathcal{F}$ is a set of features. The value function approximation is given by

$$\bar{J}^x(s_t^x|\theta) = \sum_{f \in \mathcal{F}} \theta_f \phi_f(s_t^x)$$

wherein $\theta = (\theta_f)_{f \in \mathcal{F}}$ is a column vector of weights with one coefficient for each basis function. It is computationally efficient to estimate the value function using basis functions because we choose the number of features to be fewer than the size of the state space. Classical linear regression methods can then be used to estimate θ. However, choosing an appropriate set of basis functions to properly capture the complexity of the modeled system can be challenging. LSTD updates θ iteratively during execution of the API algorithm.

LSTD updates the value function approximation for a fixed policy during each iteration and projects it over an infinite horizon. LSTD subtracts the current value of being in a state from the updated value of being in a state at the following iteration. LSTD can be viewed as a batch algorithm that operates by collecting samples of temporal differences and then using least squares regression to find the best linear fit [27]. LSTD obtains a least squares regression fit so that the sum of the temporal differences over the simulation is as close to zero as possible.

Within the construct of our LSTD algorithm, a total of K temporal difference sample realizations are collected in each policy evaluation loop where the kth temporal difference is denoted $C(s_{t,k}, X^{\pi}_{\theta}(s_{t,k})) + \gamma \theta^{\top} \phi(s^{x}_{t,k}) - \theta^{\top} \phi(s^{x}_{t-1,k})$ where $\phi(\cdot)$ is a column vector of basis function evaluations and the policy (i.e., decision function) $X^{\pi}_{\theta}(s_{t,k})$ is defined below

$$X^{\pi}_{\theta}(s_{t,k}) = \underset{x \in \mathcal{X}(s_t)}{\arg\max}\ C(s_t, x) + \gamma \bar{J}(s^{x}_{t} | \theta).$$

To solve the approximate dynamic program, we need to solve the routing portion of the problem. The size and complexity of this inner maximization problem makes exact solution methods intractable. To generate routing solutions, we adapt two separate algorithms as previously developed in Chapter IV. The first is a holistic routing scheme utilizing the quiz problem heuristic to generate routes while the second

uses a segmented rollout algorithm approach as a value function estimator to make decisions. The Index algorithm selects the first available CUAV and examines all feasible routes utilizing the quiz problem heuristic to generate a value estimation for each route. The route with the highest value estimation is selected, the decision is recorded, and inventory levels are updated according to the decision. The process is then repeated until all available CUAVs have been assigned a route. The decision can then be given to the dynamic programming algorithm. Our Index heuristic, which explicitly account for our multiclass formulation, is summarized in Algorithm 5.

Algorithm 5 Index algorithm route construction utilizing the quiz problem heuristic

Step 1. **For** all available CUAVs, $v = 1, 2, ..., \min(v_t, V^{crew})$

Step 2a. **For** all possible routes $a = 1, 2, ..., A$
Utilizing the quiz problem heuristic
$\frac{\psi_{ma} \min(\sum_{b=1}^{B} \sum_{c=1}^{C} R^{max} - r_{tbc} + \hat{D}_{t+1,c}, V^{cap})}{(1-\psi_{ma})}$, generate the value function estimation for each route
End

Step 2b. Assign CUAV route with max value

Step 2c. Record the decision for the vth CUAV and update inventory levels

End

Step 3. Provide routing decision to the dynamic programming algorithm

Modeling the success of the CUAV route holistically as in Algorithm 5 is ad–vantageous because it emphasizes selecting routes wherein the CUAV has greater probability of survival to deliver supplies in the future. This method will avoid routes that have low probability of survival in favor of more easily accessible bases. For the Rollout algorithm, the value for the gth leg is calculated using Equation 37 wherein the risk of each segment of the route is individually captured and the value of the CUAV to deliver supplies in the future is explicitly captured as well. The segmented Rollout algorithm approach, which explicitly accounts for our multiclass formulation, is summarized in Algorithm 6.

$$V_g = \frac{\psi_{ma_1} \min(R^{max} - \sum_{c=1}^{C} r_{tbc_1} + \hat{D}_{t+1,c}, \frac{V^{cap}}{g}) + ... + \psi_{ma_g} \min(R^{max} - \sum_{c=1}^{C} r_{tb_g} + \hat{D}_{t+1,c}, \frac{V^{cap}}{g}) + \psi_{ma} V^{cap}}{(1 - \psi_{ma})} \quad (37)$$

Algorithm 6 Rollout algorithm route construction utilizing the quiz problem heuristic

Step 1. **For** all available CUAVs, $v = 1, 2, ..., \min(v_t, V^{crew})$
  Step 2a. **For** From current location select all $g \in \mathcal{G}$ such that the current location is an end point of g and g is not already part of the path.
  Utilizing Equation 37, select the best gth leg and update each expected inventory level of all FOBs visited.
  End
  Step 2b. Record the decision for the vth CUAV and update inventory levels
  End
Step 3. Provide routing decision to the dynamic programming algorithm

After a decision is made, temporal difference samples must be taken to continue the ADP algorithm. Let Φ_{t-1} and Φ_t consist of rows of basis function evaluations of the sampled post-decision states and C_t as the contribution vector for the sampled events as shown in Equation 38. The sample realization $\hat{\theta}$ is calculated using linear regression for each policy evaluation loop $n = 1, 2, ..., N$. A harmonic step-size rule is applied to smooth θ during implementation.

$$\Phi_{t-1} \triangleq \begin{bmatrix} \phi(s^x_{t-1,1})^\top \\ \vdots \\ \phi(s^x_{t-1,K})^\top \end{bmatrix}, \Phi_t \triangleq \begin{bmatrix} \phi(s^x_{t,1})^\top \\ \vdots \\ \phi(s^x_{t,K})^\top \end{bmatrix}, C_t \triangleq \begin{bmatrix} C(s_{t,1}, x_t) \\ \vdots \\ C(s_{t,K}, x_t) \end{bmatrix} \quad (38)$$

We then apply a harmonic stepsize rule to smooth in the new observation $\hat{\theta}$ with the previous estimate θ during implementation. The stepsize rule α_n is a function of

the outer loop iteration count and is defined below.

$$\alpha_n = \frac{1}{n} \tag{39}$$

The stepsize rule α_n greatly influences the rate at which the API algorithm con–verges, thus impacting the attendant solutions. Utilizing the harmonic stepsize rule, we update our θ in the following way:

$$\theta^n \leftarrow \theta^{n-1}(1 - \alpha_n) + \hat{\theta}(\alpha_n). \tag{40}$$

Equation 40 shows that the updated θ^n is weighted most heavily by our current estimate, θ^{n-1}, and then moved toward our new estimate, $\hat{\theta}$, by an incremental amount proportional to α_n. Initially, greater emphasis is placed on $\hat{\theta}$, but as the number of iterations increases the incremental effect of $\hat{\theta}$ is lessened. Moreover, as the number of iterations increases, any single $\hat{\theta}$ has less influence than the estimate based on information from the first $n - 1$ iterations.

Upon obtaining an updated parameter vector θ, we have completed one policy improvement iteration of the algorithm. The parameters N and K are tunable, where N is the number of policy improvement iterations completed and K is the number of policy evaluation iterations completed. After both loops are completed our algorithm has generated a policy and terminates.

5.3 Analysis

We can find a policy for a large, multiclass forward operating base (FOB) problem instance of the military inventory routing problem (MILIRP) utilizing the Markov decision process (MDP) formulation discussed in Section 5.1. We compare our two proposed route construction methods within the construct of our approximate dy–

namic programming (ADP) algorithm (i.e., LSTD-Index, and LSTD-Rollout). Moreover, we run a designed experiment to find the algorithmic and model parameters that yield the best results for our ADP algorithm.

MDP Parameterization.

The research effort maintains constancy in terms of MDP model parameterization. The MILIRP is formulated as an infinite-horizon MDP wherein a single period represents a 6-hour interval. We assume that during each period the cargo unmanned aerial vehicle (CUAV) can complete all mission preparation tasks and perform the assigned mission. We also assume the FOB has stochastic demand as defined in Section 5.1.

As an extension of previous research, we model a battalion comprised of subordinate platoons that each have a consumption rate and storage capacity based on the number of personnel on site. Each platoon is located in different forward operating bases (FOBs) at dispersed locations. Based on a General Dynamics report [13], the expected daily consumption requirements of a platoon is $7,482$ pounds. We round up as a conservative estimate to an 8,000 pound daily average consumption. With four periods in one day, about one ton of supplies per period is required for sustainment. For our testing, we model the stochastic demand using this known historical average, $\bar{d}_c$, and a randomly generated error term, $\hat{\epsilon}$, uniformly distributed on the interval $[-0.5, 0.5]$. We choose to investigate two supply classes, fixing the demand to be equally proportioned between the two classes. We also make the conservative assumption that a FOB has a maximum holding capacity of three times the daily average requirement, totaling 12 tons equally divided between the two classes as well. We assume that there are no logistical failures limiting the amount of supplies available at the centralized planner. This assumption is reasonable since the central

planner is supplied via fixed wing aircraft from outside the theater of operations.

We chose a conservative two-ton carrying capacity for CUAV resupply based on Lockheed Martin's K-MAX CUAV [19]. As the requirements for CUAV resupply increase, we expect to see the number of CUAVs and crews the central planner utilizes to increase. As such, we parameterize the CUAVs and crews as multiples of Tactical Unmanned Aerial System (TUAS) platoon ratios [10]. For example, if three TUAS platoons are deployed at the central planner, the number of CUAVs would be 12 and the number of crews 6.

Recall from Section 5.1 that ψ_{ma} denotes the probability of a successful route completion from (to) the support area (SA) to (from) the FOBs along route a on map m. For our example, we assigned lower probability of success to longer flight paths and chose to use $M = 2$ threat maps. We restrict feasible routes on the network by distance and only allow routes to visit a maximum of 3 FOBs. The probability of successful one-way delivery across the network is shown for each threat map below where the first position in the matrix represents the central planner. The problem instance diagram is shown in Figure 2.

$$m_1 = \begin{bmatrix} 1 & .99 & .99 & .95 & .99 & .99 \\ .99 & 1 & .99 & .90 & 0 & 0 \\ .99 & .99 & 1 & .95 & .95 & 0 \\ .95 & .90 & .95 & 1 & .95 & .90 \\ .99 & 0 & .95 & .95 & 1 & .99 \\ .99 & 0 & 0 & .90 & .99 & 1 \end{bmatrix} \quad m_2 = \begin{bmatrix} 1 & .95 & .95 & .90 & .95 & .95 \\ .95 & 1 & .95 & .85 & 0 & 0 \\ .95 & .95 & 1 & .90 & .90 & 0 \\ .90 & .85 & .90 & 1 & .90 & .85 \\ .95 & 0 & .90 & .90 & 1 & .95 \\ .95 & 0 & 0 & .85 & .95 & 1 \end{bmatrix}$$

To investigate the effects of region volatility on algorithmic outcome, we vary the map transition probability in our experimental design . We select an initial probability of 0.5 to remain in the current threat map to model the current instability of the

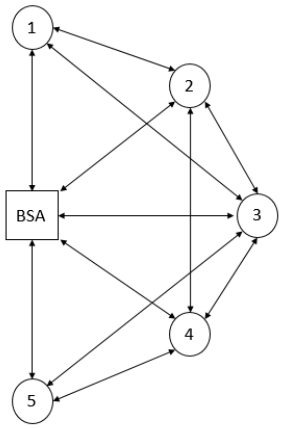

Figure 2. Problem Instance

modeled region.

The FOB must be resupplied via ground convoy to regain full capacity when the FOB's supply level for any class falls below a predetermined minimum threshold. Each time a convoy is deployed for resupply an immediate penalty is applied. This penalty represents both the risk of a FOB stock out and the increased human capital risk inherent in ground convoy operations. A subject matter expert who knows the terrain and enemy activity levels associated with the FOB would ideally supply their penalty for each FOB. This penalty creates a strong incentive to ensure that CUAVs resupply all classes within the FOB when possible. We set the penalty function $\bar{\tau}$ to be the maximum inventory level R^{max}, which is applied when the inventory level is at or below the R^{min} threshold.

We chose $\lambda = 0.98$ to be a discount factor that balances future needs with current needs. We utilized the above described MDP parameterization to create policies using our LSTD-Index and LSTD-Rollout ADP algorithms.

ADP Policies.

We obtain the ADP policy from the least squares temporal differences (LSTD) algorithm as explained in Section 5.1. We compare LSTD-Index, LSTD-Rollout, In–

dex, and Rollout policies to a myopic policy of direct delivery to the FOB with the lowest inventory level over a simulated one month planning horizon. The challenge with both these ADP algorithms is developing basis functions that accurately ap–proximate the optimal value function. All algorithms are employed with the system initialized at full capacity for each FOB.

We develop ADP policies using our proposed route generation techniques in both LSTD-Index and LSTD-Rollout algorithms. Our basis function includes first order effects for current inventory level for every class and number of CUAVs deployed. Moreover, we also chose to include the second order inventory effect for each class. Our proposed routing algorithms quickly generate solutions, allowing us to perform a designed experiment with more breadth in a reasonable amount of time.

Baseline Instance.

To stay consistent with previous research in Chapter IV, we selected the same representative baseline instance for testing the routing algorithm's performance. The baseline instance has 12 CUAVs, 2 crews, probability of staying in a low threat map of 0.8, probability of staying in a high threat map of 0.2, average period demand of 1 ton, with associated algorithmic features $K = 5000$ and $N = 30$. The algo–rithm's performance is compared to a myopic policy of direct delivery replenishment to the lowest inventory FOB. The baseline performance is shown in Table 19. The baseline scenario compares policies determined via LSTD using Index and Rollout algorithms, Index and Rollout algorithms by themselves, and the myopic policy. For the baseline instance, all the proposed solution techniques performed better than the myopic policy. However, both LSTD algorithms performed substantially better than the Index and Rollout algorithms by themselves. For this instance, the proposed Index and Rollout generated policies performed better when implemented as part of

a larger approximate dynamic programing method than when implemented alone. This result is interesting because on less complicated instances, Index and Rollout Algorithm generated polices outperformed their ADP counterparts. It appears that our proposed LSTD-Index and LSTD-Rollout algorithms are better suited to solving this more complicated instance, so we expanded our experimental region of search with a designed experiment.

Experimental design.

The goal of our experimental design is to test the robustness of our ADP algorithmic parameters and find the parameter settings that allow our proposed algorithmic approach to achieve the best performance. Our response variable for these experiments is the total value of the system when initialized at full capacity. In each experimental run, we simultaneously assess five problem features and three algorithmic features. The five problem features of interest were chosen based on what we thought might have the most effect on the system performance and what was significant in past research efforts in Chapter IV. The selected problem features are initial number of CUAVs (V), number of crews available (V^{crew}), probability of staying in a low threat map (Ω), probability of staying in a high threat map (β) and the average demand ($\bar{d}$). The three algorithmic features we chose to experiment on are inner loop iteration count (K), outer loop iteration count (N), and a categorical variable indicating whether we used the LSTD-Index or LSTD-Rollout algorithm. We simulate

Table 19. Baseline Instance

Algorithm	Value	% Improvement
Myopic	-1574.30	-
LSTD-Index	-1240.07	21.23%
LSTD-Rollout	-1241.88	21.12%
Index	-1555.15	1.22%
Rollout	-1556.74	1.12%

each resultant policy over a one month planning horizon and compare performance in terms of value function evaluation.

We consider five problem features, each of which are considered to be continuous for our analysis. We chose the vehicle and crew level to be levels associated with deploying two, three, and four TUAS platoons at the support area (SA). We do this under the assumption that TUAS platoons will be sent in greater numbers to support resupply operations as the central planners increasingly value CUAV resupply. We parameterize the probability of remaining in low or high threat map, Ω and β respectively, to explore how regional volatility affects the value function. The lower and uppper values, 0.2 and 0.8, denote a low and high chance of transitioning to a different threat map condition, respectively. We chose demand for each class so that their sum would be equal to the total FOB consumption rate (e.g., if $\bar{d} = 1$ then $\bar{d}_1 = 0.5$, and $\bar{d}_2 = 0.5$). Each class maintains their own exogenous error term ϵ. Average demand was chosen to range from current consumption rates (i.e., one ton every six hours) to a considerably larger rate of consumption (i.e., two tons every six hours) to explore how demand schedule increases would affect the system.

We chose the four algorithmic features to better explore the experimental space. Based on our initial testing, we set the inner loop count to a low of $5,000$ and a high of $15,000$. We chose a wider upper bound for the outer loop iteration counter to allow for the most accurate value function approximation for the selected basis functions. Table 20 shows the problem and algorithmic settings for our experimental design.

We implemented a 2^{8-2} resolution V fractional factorial design with two center runs totaling 66 runs. In a resolution V design, all first- and second-order effects are free from being aliased with other first- and second-order interactions. For each design run, a myopic, LSTD-Index, LSTD-Rollout, Index, and Rollout policy is determined. After the ADP algorithms generate the θ-coefficients, we compute the resulting and

Table 20. Factorial Design Settings

Description	Factor	Low	Center	High
Initial number of CUAVs	V	4	8	12
Number of crews	V^{crew}	2	4	6
Probability of remaining low threat	Ω	0.2	0.5	0.8
Probability of remaining high threat	β	0.2	0.5	0.8
Average Demand	$\bar{d}$	1	1.5	2
Number of inner loops	K	5000	10000	15000
Number of outer loops	N	10	20	30
Routing strategy	Strategy	Rollout	-	Index

LSTD-Index and LSTD-Rollout policies and compare them to the myopic, Index, and Rollout policies over a one-month planning horizon.

Results.

Tables 21 and 22 show the results from the experiment with columns indicating the value of the using an LSTD-Index or LSTD-Rollout policy, myopic policy, and our proposed Index and Rollout algorithms with the corresponding design settings. The large negative numbers of the value function reflect both the difficulty of CUAV resupply with current technology and the large penalty for convoy resupply. These values indicate the overall performance of the selected algorithm on the specific problem instance where more is better. The 'Value ADP' column reflects the value of either the LSTD-Index or LSTD-Rollout policy as shown in the 'Strategy' column. Although we did not use the 'Myopic,' 'Index,' and 'Rollout' columns in the experimental design, they are included here for reference. Throughout the experimental region, the ADP techniques performed better than all others in all instances. The ADP generated policies outperformed the myopic policy by 20% on average. The problem features for which the LSTD algorithm performed best include when probability of staying in a low threat map, initial number of CUAVs, and number of crews were at their high levels, and when demand and probability of staying in a high threat

map were at their low levels. The algorithmic features that produced the best values are when number of inner and outer loops were at their high values and LSTD-Index is used.

To best analyze our gathered data, we created a regression metamodel to evaluate the parameter design effects with more statistical rigor. A stepwise regression procedure yields factors that produce a significant relationship and pass the lack of fit test. Both the resulting models include significant first- and second-order terms and perform very well in terms of prediction ability with an adjusted R^2 over 0.99. The residuals do not show signs of heteroscedasticity, and the residual by predicted plot does not raise concerns.

We consider variables significant if they pass the F-test with a p-value less than 0.05. Significant variables are summarized in Table 23. With this criterion, initial number of CUAVs, number of crews, probability of staying in a low threat condition, probability of staying in a high threat condition, and demand are significant in both first- and second-order terms. The ADP algorithm quickly converged to its generated policy, which lead to the inner loop and outer loop iteration count being not significant, even when higher order effects are considered.

According to our metamodel, the value function is maximized for the following factor levels: the probability of staying in a high threat map is 0.2, the probability of staying in a low threat map is 0.8, $5,000$ inner loops, 10 outer loops, 12 CUAVs, 6 crews, and 1 ton of average demand. These results follow intuition and suggest that both LSTD algorithms are significantly better than the myopic strategy. We also performed a paired t-test with a hypothesized mean difference of zero yielding a p-value of < 0.0001 that confirms the LSTD generated policies are statistically better than the myopic policy.

Table 21. Experimental Results

Run	V	V^{crew}	Ω	β	$\bar{d}$	K	N	Strategy	Value ADP	Myopic	Index	Rollout
1	4	2	0.2	0.2	1	5000	10	Index	-1548.90	-1715.969794	-1681.61	-1679.95
2	4	2	0.2	0.2	1	15000	30	Rollout	-1555.22	-1719.260785	-1684.97	-1686.36
3	4	2	0.2	0.2	2	5000	30	Rollout	-3586.03	-3943.281288	-3910.99	-3915.72
4	4	2	0.2	0.2	2	15000	10	Index	-3542.43	-3938.943588	-3902.88	-3911.42
5	4	2	0.2	0.8	1	5000	30	Rollout	-1571.82	-1719.13815	-1692.31	-1682.48
6	4	2	0.2	0.8	1	15000	10	Index	-1566.44	-1720.799681	-1692.69	-1674.93
7	4	2	0.2	0.8	2	5000	10	Index	-3664.27	-3946.322863	-3914.71	-3916.49
8	4	2	0.2	0.8	2	15000	30	Rollout	-3652.90	-3945.536463	-3916.34	-3910.94
9	4	2	0.8	0.2	1	5000	30	Index	-1504.40	-1711.784882	-1676.65	-1676.40
10	4	2	0.8	0.2	1	15000	10	Rollout	-1511.53	-1711.661491	-1676.89	-1677.99
11	4	2	0.8	0.2	2	5000	10	Rollout	-3271.06	-3940.215795	-3904.77	-3918.76
12	4	2	0.8	0.2	2	15000	30	Index	-3310.59	-3943.403798	-3911.43	-3907.44
13	4	2	0.8	0.8	1	5000	10	Rollout	-1550.31	-1713.567305	-1677.01	-1682.63
14	4	2	0.8	0.8	1	15000	30	Index	-1548.78	-1714.407914	-1677.44	-1678.80
15	4	2	0.8	0.8	2	5000	30	Index	-3503.55	-3929.211004	-3913.81	-3913.95
16	4	2	0.8	0.8	2	15000	10	Rollout	-3521.04	-3928.994617	-3914.82	-3914.84
17	4	6	0.2	0.2	1	5000	30	Index	-1562.50	-1705.587971	-1682.97	-1636.41
18	4	6	0.2	0.2	1	15000	10	Rollout	-1566.92	-1708.111843	-1687.04	-1647.94
19	4	6	0.2	0.2	2	5000	10	Rollout	-3673.18	-3942.857511	-3922.75	-3873.08
20	4	6	0.2	0.2	2	15000	30	Index	-3663.97	-3948.048595	-3928.37	-3874.82
21	4	6	0.2	0.8	1	5000	10	Rollout	-1586.53	-1717.413149	-1692.92	-1649.50
22	4	6	0.2	0.8	1	15000	30	Index	-1575.65	-1708.445735	-1692.11	-1652.46
23	4	6	0.2	0.8	2	5000	30	Index	-3721.41	-3946.543992	-3925.47	-3883.33
24	4	6	0.2	0.8	2	15000	10	Rollout	-3722.17	-3948.66439	-3922.71	-3884.01
25	4	6	0.8	0.2	1	5000	10	Index	-1516.04	-1696.770594	-1695.33	-1639.89
26	4	6	0.8	0.2	1	15000	30	Rollout	-1502.33	-1698.815012	-1689.86	-1640.60
27	4	6	0.8	0.2	2	5000	30	Rollout	-3556.65	-3962.82243	-3918.77	-3877.74
28	4	6	0.8	0.2	2	15000	10	Index	-3557.27	-3956.320609	-3923.45	-3873.12
29	4	6	0.8	0.8	1	5000	30	Rollout	-1550.45	-1710.556398	-1697.03	-1643.96
30	4	6	0.8	0.8	1	15000	10	Index	-1546.78	-1700.724099	-1697.25	-1647.11
31	4	6	0.8	0.8	2	5000	10	Index	-3653.72	-3944.965783	-3922.88	-3867.24
32	4	6	0.8	0.8	2	15000	30	Rollout	-3655.06	-3945.801073	-3927.19	-3875.37
33	12	2	0.2	0.2	1	5000	10	Rollout	-1276.81	-1583.836409	-1565.32	-1560.62

Table 22. Experimental Results Continued

Run	V	V^{crew}	Ω	β	$\bar{d}$	K	N	Strategy	Value ADP	Myopic	Index	Rollout
34	12	2	0.2	0.2	1	15000	30	Index	-1279.54	-1583.500102	-1564.21	-1560.55
35	12	2	0.2	0.2	2	5000	30	Index	-2499.37	-3669.332462	-3713.90	-3706.52
36	12	2	0.2	0.2	2	15000	10	Rollout	-2440.28	-3665.02532	-3711.78	-3709.51
37	12	2	0.2	0.8	1	5000	30	Index	-1321.59	-1588.618442	-1559.62	-1555.34
38	12	2	0.2	0.8	1	15000	10	Rollout	-1309.08	-1589.062276	-1559.90	-1549.03
39	12	2	0.2	0.8	2	5000	10	Rollout	-2868.65	-3685.589355	-3710.99	-3710.23
40	12	2	0.2	0.8	2	15000	30	Index	-2908.70	-3687.660727	-3713.12	-3708.05
41	12	2	0.8	0.2	1	5000	30	Rollout	-1236.09	-1580.803106	-1557.04	-1548.60
42	12	2	0.8	0.2	1	15000	10	Index	-1232.74	-1577.395522	-1551.69	-1547.09
43	12	2	0.8	0.2	2	5000	10	Index	-2011.37	-3668.522817	-3702.51	-3692.25
44	12	2	0.8	0.2	2	15000	30	Rollout	-2029.47	-3674.833836	-3706.11	-3696.85
45	12	2	0.8	0.8	1	5000	10	Index	-1294.69	-1574.973118	-1560.62	-1562.80
46	12	2	0.8	0.8	1	15000	30	Rollout	-1289.24	-1582.542705	-1561.19	-1561.61
47	12	2	0.8	0.8	2	5000	30	Rollout	-2491.67	-3677.072133	-3707.40	-3704.87
48	12	2	0.8	0.8	2	15000	10	Index	-2493.45	-3674.682342	-3709.28	-3702.81
49	12	6	0.2	0.2	1	5000	30	Rollout	-797.61	-1611.275541	-1566.69	-1318.89
50	12	6	0.2	0.2	1	15000	10	Index	-775.18	-1619.719101	-1575.16	-1309.59
51	12	6	0.2	0.2	2	5000	10	Index	-3032.26	-3793.053258	-3646.01	-3309.71
52	12	6	0.2	0.2	2	15000	30	Rollout	-3038.86	-3777.582757	-3637.85	-3333.65
53	12	6	0.2	0.8	1	5000	10	Index	-835.64	-1625.494082	-1566.95	-1332.59
54	12	6	0.2	0.8	1	15000	30	Rollout	-917.61	-1624.65203	-1567.15	-1339.58
55	12	6	0.2	0.8	2	5000	30	Rollout	-3152.58	-3781.178338	-3636.95	-3347.00
56	12	6	0.2	0.8	2	15000	10	Index	-3113.25	-3799.63979	-3627.46	-3325.41
57	12	6	0.8	0.2	1	5000	10	Rollout	-757.38	-1619.718581	-1565.64	-1323.13
58	12	6	0.8	0.2	1	15000	30	Index	-676.57	-1625.772357	-1569.38	-1326.78
59	12	6	0.8	0.2	2	5000	30	Index	-2860.35	-3774.010358	-3640.37	-3334.68
60	12	6	0.8	0.2	2	15000	10	Rollout	-2830.60	-3784.484823	-3632.67	-3303.06
61	12	6	0.8	0.8	1	5000	30	Index	-731.34	-1604.656758	-1565.58	-1327.78
62	12	6	0.8	0.8	1	15000	10	Rollout	-824.70	-1619.332354	-1561.62	-1319.42
63	12	6	0.8	0.8	2	5000	10	Rollout	-3038.20	-3772.708612	-3639.67	-3322.43
64	12	6	0.8	0.8	2	15000	30	Index	-3027.44	-3772.385515	-3643.42	-3316.16
65	8	4	0.5	0.5	1.5	10000	20	Rollout	-2352.30	-2778.31403	-2730.57	-2542.59
66	8	4	0.5	0.5	1.5	10000	20	Index	-2354.24	-2782.664696	-2734.94	-2560.27

Table 23. Factors Influencing CUAV Resupply

Variable	Sum of Squares	F Test	% Contribution
V	7308646	$< .0001$	11.16%
V^{crew}	41422	0.1678	0.06%
Ω	281232	0.0006	0.43%
β	250655	0.0011	0.38%
$\bar{d}$	55822140	$< .0001$	85.21%
$V * \bar{d}$	424127	$< .0001$	0.65%
$V^{crew} * \bar{d}$	1373597	$< .0001$	2.10%
$\Omega * \beta$	7804	0.5466	0.01%
$\Omega * \bar{d}$	113499	0.0245	0.17%
$\beta * \bar{d}$	102674	0.032	0.16%

5.4 Conclusions

The intent of this research is to implement established algorithms on more complex problem instances of a military inventory routing problem (MILIRP) to compare algorithmic performance and improve upon simple strategies. Developing a Markov decision process (MDP) model of the multiclass MILIRP enables examination of the disparate conditions the military faces in hostile environments.

Utilizing CUAV assets for resupply efforts is an important issue in military ap–plications. A plethora of obstacles, to include poorly developed transportation in–frastructure, adverse weather conditions, terrain, enemy threat and actions, and the availability of distribution assets, all inhibit successful distribution of supplies from the support area (SA) to the forward operating bases (FOBs). Moreover, insurgent use of improvised explosive devices (IEDs) present a clear and present danger to convoy resupply operations throughout the operational environment and has been successful in disrupting replenishment procedures [25]. Because of this danger, the effective implementation of CUAVs has been of increasing interest worldwide [15, 14]. This paper provides unique insight into using CUAVs in combat environments for resupply. The effective removal of the human element via CUAV resupply could help reduce the need for convoy resupply missions with high casualty rates. CUAV bene–fits include: better performance in adverse weather conditions, higher flight ceilings, and no escort requirement restrictions. All these benefits yield a lower probability of vehicle destruction via enemy actions (e.g., via man-portable air-defense systems and small arms fire). The most important benefit of CUAVs is their ability to save lives by alleviating manned ground convoy resupply requirements. Although CUAVs do not yet have the ability to completely handle FOB supply requirements as seen through our large negative value function, each successful CUAV delivery means less men and women exposed to enemy threats to include IEDs.

We formulate an MDP model of the extended MILIRP and determine a policy for realistic instances in order to compare two approximate dynamic programming algorithms. We develop a representative baseline instance and conclude that both the ADP algorithms and heuristic search techniques perform better than simple strate–gies. We test our approach with an experimental design and empirically find the design settings that maximize the ADP algorithmic performance. Moreover, we con–clude that there is no statistically significant difference between utilizing LSTD-Index or LSTD-Rollout techniques despite the Rollout algorithm's more accurate represen–tation of periodic risk. Additionally, although they all perform better than the myopic policy, we conclude both LSTD-Index and LSTD-Rollout algorithms perform better on this problem in terms of policy value than the standalone Index and Rollout al–gorithms. This is due to the ADP's ability to evaluate and update the value function estimation via the selection of proper basis functions. It is possible that there exists a better set of basis functions that will allow the linear model to more effectively capture the nuances of this complex problem. Further exploration of basis functions may yield improved results. Due to their demonstrated success, we conclude that the LSTD-Index and LSTD-Rollout algorithms presented here are an effective way to develop resupply policies for the MILIRP.

5.5 Acknowledgments

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

VI. Contributions

The military inventory routing problem (MILIRP) is a stochastic inventory problem (IRP) with distinct complicating features that distinguishes itself from other stochastic IRPs currently in the literature. Decision makers must develop resupply polices to ensure the safety and security of their troops throughout their area of operations. The goal of this research is to develop models and solution procedures for the MILIRP and to inform the development of tactics, techniques, and procedures for utilizing cargo unmanned aerial vehicles (CUAVs) for resupply.

This dissertation provides high quality and innovative research that constitute a significant contribution to the Operations Research field. This research analyzes the structural properties of a small instance MILIRP and generates a mathematical proof that shows the conditions that must be met to ensure global monotonicity of the value function. Moreover, this work develops a Markov decision process (MDP) model for this instance, creates a novel algorithm that exploits this monotonic structure, and demonstrates this algorithm's efficacy. This work expands the model to a larger scale, allowing for multiple routing, and develops two additional routing algorithms to solve these more complex instances. This dissertation tests these algorithms and shows that each perform better than a myopic strategy and have merit when used individually or in conjunction with approximate dynamic programming methods. Moreover, this dissertation formulates a multi-class MDP model and demonstrates the created algorithms' efficacy on these more vastly complex instances. Results indicate that the LSTD-Index, and LSTD-Rollout algorithms developed herein scale well as complexity increases, and which indicate future work on the MILIRP should endeavor to incorporate these algorithms.

Appendix A. Acronyms

ADP - approximate dynamic programming

BCT - brigade combat team

BSA - brigade supply area

BSB - brigade supply battalion

CUAV - cargo unmanned aerial vehicle

F - CUAV does not successfully deliver supplies

FOB - forward operating base

IED - improvised explosive device

IRP - inventory routing problem

LSTD - least squares temporal differencing

MANPADS - man-portable air-defense system

MDP - Markov decision process

MILIRP - military inventory routing problem

MLSTD - monotone least squares temporal differencing

SA - support area

SF - CUAV delivers supplies to FOB, but does not successfully return

SIRP - stochastic inventory routing problem

SS - CUAV completes both legs of the journey

UAV - unmanned aerial vehicles

VMI - vendor managed inventory

VRP - vehicle routing problem

Bibliography

1. Adelman, Daniel. 2004. A price-directed approach to stochastic inventory routing. *Operations Research*, **52**(4), 499–514.

2. Ahrens, Frederick A. 2015. Monotonic response surface estimation by constrained coefficients. *Pages 689–700 of: Winter Simulation Conference (WSC), 2015*. IEEE.

3. Bertsekas, Dimitri P, & Castanon, David A. 1999. Rollout algorithms for stochastic scheduling problems. *Journal of Heuristics*, **5**(1), 89–108.

4. Bertsekas, Dimitri P, & Tsitsiklis, John N. 1996. *Neuro-dynamic programming*. Vol. 7. Belmont, MA: Athena Scientific.

5. Bradtke, Steven J, & Barto, Andrew G. 1996. Linear least-squares algorithms for temporal difference learning. *Machine Learning*, **22**(1-3), 33–57.

6. Campbell, Ann, Clarke, Lloyd, Kleywegt, Anton, & Savelsbergh, Martin. 1998. The inventory routing problem. *Pages 95–113 of: Fleet management and logistics*. Springer.

7. Coelho, Leandro C., Cordeau, Jean-Franois, & Laporte, Gilbert. 2014. Thirty Years of Inventory Routing. *Transportation Science*, **48**(1), 1–19.

8. Davis, Michael T, Robbins, Matthew J, & Lunday, Brian J. 2017. Approximate dynamic programming for missile defense interceptor fire control. *European Journal of Operational Research*, **259**(3), 873–886.

9. Department of the Army. 2009. Army Unmanned Aircraft System (UAS) Operations No. 3-04. *Army Field Manual*.

10. Department of the Army. 2010. Brigade Combat Team No. 3-90.6. *Army Field Manual*.

11. Department of the Army. 2014a. Brigade Support Battalion No. 4-90. *Army Field Manual*.

12. Department of the Army. 2014b. General Supply and Field Services Operations No. 4-42. *Army Techniques Publication*.

13. General Dynamics Information Technology. 2010. *Future Modular Force Resupply Mission for Unmanned Aircraft Systems (UAS)*. General Dynamics Information Technology.

14. Haider, André. 2014. Remotely Piloted Aircraft Systems in Contested Environments. *Joint Air Power Competence Centre JAPCC*.

15. Hoffman, Michael. 2012. *K-Max cargo UAS exceeds expectations in Afghanistan test.* http://www.defensetech.org/2012/07/26/k-max-cargo-uas-exceeds-expectations-in-afghanistan-test/. Accessed: 2016-04-3.

16. Jiang, Daniel R, & Powell, Warren B. 2015. An approximate dynamic pro–gramming algorithm for monotone value functions. *Operations Research*, **63**(6), 1489–1511.

17. Kleywegt, Anton J., Nori, Vijay S., & Savelsbergh, Martin W. P. 2002. The Stochastic Inventory Routing Problem with Direct Deliveries. *Transportation Science*, **36**(1), 94.

18. Kleywegt, Anton J., Nori, Vijay S., & Savelsbergh, Martin W. P. 2004. Dy–namic Programming Approximations for a Stochastic Inventory Routing Problem. *Transportation Science*, **38**(1), 42 – 70.

19. Lockheed Martin. 2010. *K-MAX Unmanned Aircraft System.* http://www.lockheedmartin.com/content/dam/lockheed/data/ms2/documents/K-MAX-brochure.pdf. Accessed: 2016-3-1.

20. McCormack, Ian. 2014. *The Military Inventory Routing Problem with Direct Delivery.* M.Sci., Air Force Institute of Technology.

21. McKenna, R. S., Robbins, M. J., Lunday, B. J., & McCormack, I. M. 2016. *Approximate Dynamic Programming for the The Military Inventory Routing Problem with Direct Delivery.* Tech. rept. Air Force Institute of Technology.

22. McKenna, Rebekah. 2015. *Using Approximate Dynamic Programming to Solve the The Military Inventory Routing Problem with Direct Delivery.* M.Sci., Air Force Institute of Technology.

23. Michaels, Jim. 2017. *The next big thing: Drones supplying U.S. troops.* http://www.usatoday.com/story/news/world/2017/02/22/drones-supplying-united-states-troops/98155244/. Accessed: 2017-03-09.

24. Minkoff, Alan S. 1993. A Markov decision model and decomposition heuristic for dynamic vehicle dispatching. *Operations Research*, **41**(1), 77–90.

25. Peterson, Troy M., & Staley, Jason R. 2011. Business Case Analysis of Cargo Unmanned Aircraft Systems (UAS) Capability in Support of Forward Deployed Logistics in Operation Enduring Freedom (OEF). 1–15.

26. Powell, Warren B. 2009. What you should know about approximate dynamic programming. *Naval Research Logistics (NRL)*, **56**(3), 239–249.

27. Powell, Warren B. 2011. *Approximate Dynamic Programming: Solving the Curses of Dimensionality.* 2 edn. Hoboken, NJ: John Wiley & Sons, Inc.

28. Powell, Warren B. 2016. Perspectives of approximate dynamic programming. *Annals of Operations Research*, **241**(1-2), 319–356.

29. Puterman, Martin L. 1994. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. Hoboken, NJ: John Wiley & Sons, Inc.

30. Rettke, Aaron J, Robbins, Matthew J, & Lunday, Brian J. 2016. Approximate dynamic programming for the dispatch of military medical evacuation assets. *European Journal of Operational Research*, **254**(3), 824–839.

31. Salgado, Ethan. 2016. *Using Approximate Dynamic Programming to Solve the Stochastic Demand Military Inventory Routing Problem with Direct Delivery*. M.Sci., Air Force Institute of Technology.

32. Sutton, Richard S, & Barto, Andrew G. 1998. *Reinforcement learning: An introduction*. Cambridge, MA: MIT press.

33. Toth, Paolo, & Vigo, Daniele. 2001. *The Vehicle Routing Problem*. Vol. 18. SIAM.

34. Van Roy, Benjamin, Bertsekas, Dimitri P, Lee, Yuchun, & Tsitsiklis, John N. 1997. A neuro-dynamic programming approach to retailer inventory management. *Pages 4052–4057 of: Decision and Control, 1997., Proceedings of the 36th IEEE Conference on*, vol. 4. IEEE.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704–0188

1. REPORT DATE *(DD–MM–YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From — To)*
16–09–2018	Doctoral Dissertation	SEP 2016 — SEP 2018

4. TITLE AND SUBTITLE

The Military Inventory Routing Problem: Utilizing Heuristics Within a Least Squares Temporal Differences Algorithm to Solve a Multiclass Stochastic Inventory Routing Problem with Vehicle Loss

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)

Salgado, Ethan L., Captain, 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/ENS)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER

AFIT-ENS-PhD-18-DS-042

9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES)

TRADOC Capability Manager for Unmanned Aircraft Systems
Deputy, TCM-UAS Mr. Glenn A. Rizzi
453 Novosel Street
Fort Rucker, AL 36362
glenn.a.rizzi.civ@mail.mil

10. SPONSOR/MONITOR'S ACRONYM(S)

TCM-UAS

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

Military commanders currently resupply forward operating bases (FOBs) from a central location within an area of operations mainly via convoy operations in a way that closely resembles vendor managed inventory practices. Commanders must decide when and how much inventory to distribute throughout their area of operations while minimizing soldier risk. Technology currently exists that makes utilizing unmanned cargo aerial vehicles (CUAVs) for resupply an attractive alternative due to the dangers of utilizing convoy operations. Enemy actions in wartime environments pose a significant risk to a CUAV's ability to safely deliver supplies to a FOB. We develop a Markov decision process (MDP) model to examine this military inventory routing problem (MILIRP).
In our first paper we examine the structure of the MILIRP by considering a small problem instance and prove value function monotonicity when a sufficient penalty is applied. Moreover, we develop a monotone least squares temporal differences (MLSTD) algorithm that exploits this structure and demonstrate its efficacy for approximately solving this problem class. We compare MLSTD to least squares temporal differences (LSTD), a similar ADP algorithm that does not exploit monotonicity. MLSTD attains a 3.05% optimality gap for a baseline scenario and outperforms LSTD by 31.86% on average in our computational experiments. Our second paper expands the problem complexity with additional FOBs. We generate two new algorithms, Index and Rollout, for the routing portion and implement an LSTD algorithm that utilized these to produce solutions 22% better than myopic generated solutions on average. Our third paper greatly increases problem complexity with the addition of supply classes. We formulate an MDP model to handle the increased complexity and implement our LSTD-Index and LSTD-Rollout algorithms to solve this larger problem instance and perform 21% better on average than a myopic policy.

15. SUBJECT TERMS

stochastic inventory routing approximate dynamic programming Markov decision process vendor managed inventory military least squares temporal differences

16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT	18. NUMBER OF PAGES	19a. NAME OF RESPONSIBLE PERSON
a. REPORT	b. ABSTRACT	c. THIS PAGE			Dr. Matthew J. Robbins, AFIT/ENS
U	U	U	U	122	19b. TELEPHONE NUMBER *(include area code)* (937) 255-3636, x4539 matthew.robbins@afit.edu