3-2021

Improving Air Battle Management Target Assignment Processes via Approximate Dynamic Programming

Joseph M. Liles IV

Improving Air Battle Management Target Assignment Processes via Approximate Dynamic Programming

THESIS

Joseph M. Liles IV, Lt Col, USAF

AFIT-ENS-MS-21-M-173

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

Wright-Patterson Air Force Base, Ohio

AFIT-ENS-MS-21-M-173

IMPROVING AIR BATTLE MANAGEMENT TARGET ASSIGNMENT PROCESSES VIA APPROXIMATE DYNAMIC PROGRAMMING

THESIS

Presented to the Faculty

Department of Operational Sciences

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 Master of Science in Operations Research

Joseph M. Liles IV, BS, MS

Lt Col, USAF

March 25, 2021

AFIT-ENS-MS-21-M-173

IMPROVING AIR BATTLE MANAGEMENT TARGET ASSIGNMENT PROCESSES VIA APPROXIMATE DYNAMIC PROGRAMMING

THESIS

Joseph M. Liles IV, BS, MS
Lt Col, USAF

Committee Membership:

Matthew J. Robbins, PhD
Chair

Brian J. Lunday, PhD
Member

AFIT-ENS-MS-21-M-173

Abstract

Military air battle managers face many challenges when directing operations in quickly evolving combat scenarios. These scenarios require rapid decisions to engage moving and unpredictable targets. In defensive operations, the success of a sequence of air battle management decisions is reflected by the friendly force's ability to maintain air superiority by defending friendly assets. We develop a Markov decision process (MDP) model of the air battle management (ABM) problem, wherein a set of unmanned combat aerial vehicles (UCAV) is tasked to defend a central asset from cruise missiles that arrive stochastically over time. The MDP model explains each component of this complex and dynamic problem with the understanding that an exact solution using traditional dynamic programming techniques is computationally intractable. We utilize an approximate dynamic programming (ADP) technique known as approximate policy iteration with least squares temporal differences (API-LSTD) to find high-quality solutions to the ABM problem. We create a generic yet representative combat scenario to illustrate how the ADP solution compares in quality to a reasonable benchmark policy. Our API-LSTD policy improves mean success rate by 6.8% compared to the benchmark policy and offers a 318% increase in the frequency at which the policy performs equivalently to an optimal policy. The improvements come at a cost of idle time, indicating a trade-off between defensive success rates and fuel cost. We show that the increase in performance of the API-LSTD policy is equivalent on average to increasing UCAV flight speed by 50% under the benchmark policy. These results inform force management and acquisition decisions, and aid in the development of more effective tactics, techniques, and procedures.

To Heidi and Isabelle — never stop asking questions.

Acknowledgements

First and foremost, to my wife and daughter: thank you for your support and patience. None of this would have been possible without you.

To my advisor, Dr. Matthew Robbins: thank you for your advice and talented teaching abilities helping me to find an unexpected passion in the modeling and analysis of stochastic systems.

To Dr. Brian Lunday: thank you for your valuable guidance both as a thesis reader and as a classroom instructor.

To my classmates with whom I shared this adventure over the last eighteen months: I could not have asked to be part of a better group of people. I wish you all the best in your future endeavors.

To my parents and family: thank you for your unquestioning support of my interests wherever I take them.

Finally, I would like to express my gratitude to the many staff members who work tirelessly behind the scenes at the Air Force Institute of Technology to enable these research efforts. Your important contributions to student success do not go unnoticed.

Joseph M. Liles IV

Table of Contents

Page

List of Figures

List of Tables

IMPROVING AIR BATTLE MANAGEMENT TARGET ASSIGNMENT PROCESSES VIA APPROXIMATE DYNAMIC PROGRAMMING

I. Introduction

The United States recognizes an increasingly complex and volatile global security environment marked by rapid technological advancements in areas such as artificial intelligence and autonomy (Department of Defense, 2018*c*). These technologies have decreasing barriers to entry, amplifying opportunities for both state- and non-state actors to develop and refine military capabilities in these sectors. Large-scale combat against these technologies is unprecedented, and given their widespread and increasing commercial availability, the traditional conventional overmatch of the United States is not guaranteed (Department of Defense, 2018*c*). Without this quantitative overmatch, offensive and defensive operations realize advantages in superlative decision-making processes.

The theater air control system (TACS) conducts air battle management (ABM) activities, providing counterair operations oversight as well as command and control functions for all friendly air assets within a theater of operations. Effective management of counterair operations is critical to the achievement of military objectives. Counterair operations comprise two categories: offensive counterair (OCA) and defensive counterair (DCA), with expected overlap and synchronization. OCA operations seek to destroy, disrupt, or neutralize enemy forces as close to their source as possible, whereas DCA operations seek to neutralize enemy forces attempting to penetrate an airspace (Department of Defense, 2018*b*). Together, these operations provide an area in which friendly forces can operate while protected from air and missile threats

(Department of Defense, 2018*b*). The TACS consists of regional ground and airborne elements wherein the ground-based control and reporting center (CRC), under the direction of the regional air defense commander (RADC), serves as the primary decision authority for target assignment during DCA missions (Department of the Air Force, 2011).

The RADC ensures friendly forces conduct effective DCA operations to obtain or maintain air superiority in a region within the theater of operations. Friendly forces are said to achieve air superiority in a region when they are able to operate without *prohibitive* enemy interference. Although air superiority itself provides little intrinsic benefit, it acts as an enabler for friendly forces to conduct missions subject to a minimized vulnerability of detection and attack (Department of the Air Force, 2011). To meet this objective of obtaining or maintaining air superiority, OCA and DCA forces work in concert. OCA forces aim to preempt enemy attacks, and DCA forces respond reactively as needed. DCA forces implement the same sequential process for any airborne threat: detect, identify, intercept, and destroy (Department of the Air Force, 2011). The RADC and a staff of air battle managers within the CRC must consider the speed, trajectory, location, and threat level of incoming hostile forces to appropriately prioritize assignments for intercept within this process, especially when considering multiple targets and multiple methods of intercept simultaneously.

DCA operations may be classified as *active air and missile defense* and include employment of several disparate asset types and systems, e.g., fighter aircraft, surface-to-air missiles, anti-aircraft artillery, electromagnetic warfare systems, and ballistic missile defense systems, to destroy the hostile forces or reduce the effectiveness of their attacks (Department of Defense, 2018*b*). Alternatively, DCA operations may be classified as *passive air and missile defense* and include all measures not considered active, e.g., detection and warning; mobility and dispersion; chemical, biological, ra-

diological, and nuclear (CBRN) defense; or low-observable technology (Department of Defense, 2018*b*). The TACS airborne elements, the Airborne Warning and Control System (AWACS) and the Joint Surveillance Target Attack Radar System (JSTARS), use long-range sensor platforms to relay real-time information supporting the decisions of how to engage enemy targets (Department of the Air Force, 2011). Fighter aircraft involved in these missions may be traditional manned aircraft, or they may also be unmanned aerial vehicles (UAV). Weaponized UAVs intended for use in combat are referred to as unmanned combat aerial vehicles (UCAV) and can be either remotely controlled or autonomous. The distinction between remotely controlled and autonomous is important because it highlights a key difference in the nature of their command and control.

Air battle managers face many challenges when conducting counterair operations. Air combat situations evolve quickly, requiring rapid decisions to engage moving and unpredictable targets. Moreover, each aircraft mission design series (MDS) is characterized by different flight dynamics and, even within the same MDS, each individual aircraft may have a different, mission-specific weapons configuration. The increased proliferation of UAV technology further complicates the DCA mission with regard both to the control of friendly UCAV forces and to the defense against hostile UCAV forces because UAVs are not subject to the same human factor limitations as traditional manned aircraft. The United States Department of Defense currently operates over 11,000 UAVs (Department of Defense, 2014). In the civilian and commercial sectors, over 1.6 million UAVs are currently registered with the Federal Aviation Administration, and the number is quickly increasing (Federal Aviation Administration, 2020). Worldwide, over 70 countries have advertised military UAV capabilities (Franke, 2014). Additionally, as Operation Inherent Resolve coalition forces witnessed recently during conflict with the Islamic State, basic UAV technology is so widely ac-

cessible that any non-state actor should reasonably be expected to employ UAVs in some fashion.

The 2018 National Defense Strategy of the United States highlights autonomous aircraft as a key capability (Department of Defense, 2018*c*). Operating autonomous UCAVs in counterair roles offers advantages to offensive and defensive strategies without human factor limitations. Of course, adversaries may realize these same benefits. Although the United States implements controls to guard against unethical employment of autonomous vehicles in combat, it is important to understand how these technologies can be employed independent of policy and ethical considerations to best defend against their employment by hostile forces. Understanding UCAV employment strategies can provide for improved ABM policies by considering the probabilistic outcomes of a large number of sequential decisions. These improvements are realized through development of enhanced training practices along with more effective tactics, techniques, and procedures. Furthermore, understanding desired capabilities informs the Joint Capabilities Integration and Development System (JCIDS) for a variety of systems, allowing for the creation of more accurate system requirements and overall more efficient acquisitions programs.

In this research, we formulate the ABM Problem, specifically considering the employment and management of autonomous UCAVs in a defensive role. During DCA operations, air battle managers must make sequential decisions regarding the assignment of friendly aircraft to the intercept of incoming enemy targets. Using a myopic (closest-available) task-assignment policy to intercept targets may risk leaving other parts of the airspace unnecessarily open for attack or risk leaving friendly forces poorly postured to respond to future threats. If improved task-assignment policies exist, they may not be immediately obvious, especially when confronted with an overwhelming enemy force requiring a complex series of decisions. The nature of ABM is both

dynamic and stochastic with enemy targets approaching over time at an uncertain arrival rate and moving towards their destination with some uncertain velocity. We assume prior intelligence-gathering activities allow for reasonable conclusions regarding the stochastic behavior of these enemy targets. Air battle managers must assign friendly forces to engage each enemy target before it poses a critical level of threat to friendly defended assets, suggesting there is a time window in which each target must be intercepted. To find high-quality solutions to the ABM Problem, we develop a Markov decision process (MDP), which is a stochastic dynamic programming technique used for sequential decision making when outcomes are uncertain.

Although traditional dynamic programming techniques are effective at finding exact solutions to stochastic and dynamic decision problems, many realistic problems such as the ABM Problem involve too many combinations of model states and possible decisions to be computationally tractable. This computational intractability is referred to as the *curse of dimensionality* or combinatorial explosion, wherein the dimension of a problem increases exponentially relative to the number of individual problem elements. In these cases, approximate dynamic programming (ADP) is an effective method for developing high-quality, approximate solutions to otherwise unsolvable problems. Our ADP solution employs an approximate policy iteration technique using least squares temporal differences (API-LSTD). To demonstrate the effectiveness of our ADP solution, we create a generic yet representative DCA scenario alongside a simulation model that characterizes how the DCA scenario evolves over time. We design and conduct a series of computational experiments to explore how the numeric algorithm and problem parameters affect API-LSTD solution quality.

The remaining chapters discuss further details of this research. Chapter II reviews applicable previous literary works and discusses their relevance to the current research. Chapter III elaborates on the ABM Problem, and Chapter IV proposes an ADP

solution methodology. Chapter V describes a computational experiment designed to improve the performance of the ADP solution methodology along with an analysis of the results. Chapter VI concludes the research and provides suggestions for future research.

II. Literature Review

This paper focuses on finding high-quality policies for ABM target assignment during DCA operations. The study of UCAV routing to develop an appropriate target assignment mathematical model suggests the applicability of several classical combinatorial optimization routing problems: the traveling salesman problem (TSP), the vehicle routing problem (VRP), and the orienteering problem (OP). Additionally, Markov decision processes (MDP) are specifically suited for solving sequential decision-making problems under uncertainty. Routing considerations aside, the class of problems known as *assignment problems* (AP) is itself non-trivial and should be investigated. Finally, relevant ADP papers inform the development of our solution approach. The remainder of this chapter discusses the components of these related problems and examines a variety of applicable research.

2.1 Markov Decision Processes

An MDP formulates a stochastic sequential decision-making problem wherein a decision-maker selects actions based on the observed state of a system. The system then transitions to a new state based on the current state, the action selected, and a set of transition probabilities. Of profound importance is that the process exhibits the Markov property; if the present state of the system is known, the future of the system is independent of its past (Kulkarni, 2017). In an MDP, it follows that the decision maker is able to make decisions based only on the current state of the system and subsequently observe the system evolve independent of its previous history.

Puterman (2005) provides a thorough description of the components of an MDP model. First, there exists a set of decision epochs, $\mathcal{T}$, wherein each decision epoch $t \in \mathcal{T}$ represents a point in time when a decision is made, and the system is observed

to occupy a state $S_t \in \mathcal{S}$ during every decision epoch. Depending on the current state, $S_t \in \mathcal{S}$, the decision maker is able to choose from a set of allowable actions, $a \in \mathcal{A}_s$. The decision maker chooses action $a_t \in \mathcal{A}_s$ while in state $S_t \in \mathcal{S}$ at time $t \in \mathcal{T}$ and receives a contribution (or incurs a cost) given by $C(S_t, a_t)$. Finally, at time $t+1$, the decision maker observes the system in state $S_{t+1} \in \mathcal{S}$ with probability $p(S_{t+1}|S_t, a_t)$. Thus, an MDP is defined as the set of components $\{\mathcal{T}, \mathcal{S}, \mathcal{A}_s, C(S_t, a_t), p(S_{t+1}|S_t, a_t)\}$. These components serve as a common mathematical framework to represent problems in this area of research.

The decision maker chooses action $a_t \in \mathcal{A}_s$ while in state $S_t \in \mathcal{S}$ at time $t \in \mathcal{T}$ and receives a contribution (or incurs a cost) given by $C(S_t, a_t)$. Finally, at time $t+1$, the decision maker observes the system in state $S_{t+1} \in \mathcal{S}$ with probability $p(S_{t+1}|S_t, a_t)$.

2.2 Combinatorial Optimization Problems for Route Planning

The TSP (see, e.g., Dantzig et al. (1954)) is one of the most widely studied problems in combinatorial optimization. It involves a directed or undirected graph with a set of nodes and edges with associated weights (representing some manifestation of travel cost). The goal of the TSP is to determine a single entity's most efficient route to visit all nodes exactly once and return to the point of origin. The TSP and its variations are typically modeled using integer linear programming.

The VRP is a generalization of the TSP wherein the goal is also to visit all nodes as efficiently as possible, but it is formulated with a group of K identical entities available to traverse the graph (when $K = 1$, the VRP is equivalent to the TSP). Originally proposed by Dantzig and Ramser (1959), the VRP is experiencing increasing applicability to transportation logistics given the steadily increasing urbanization of the global population and the highly variable nature of urban travel times (United Nations, 2018). In applications of the VRP, a vehicle's ability to move directly be-

tween pairs of nodes may be additionally constrained, e.g., due to fuel capacity or an operator's fatigue level.

When there is a limited ability to visit every node in the graph, it necessary to formulate the problem such that the objective is modified to instead visit a subset of nodes that maximizes the sum of values associated with the individual nodes. This variation of the VRP (with $K = 1$) is known as the orienteering problem (OP) or (with $K > 1$) the team orienteering problem (TOP). Tsiligirides (1984) originally introduces the OP as distinct from the VRP. Golden et al. (1987) formalize the problem, describing the sport of orienteering wherein control points with associated reward values are located across a geographical region. Competitors attempt to visit any number of these points and then return to the starting point within the allotted time. The classical version of the OP is equivalent to the generalized TSP (Tsiligirides, 1984).

Chao et al. (1996) propose the TOP as an extension of the single-competitor OP wherein a team of competitors starting at the same point attempt to navigate to a number of control points and return to the starting location within a known time limit. The score for visiting a control point can only be received once. Each team member must determine a route that collectively maximizes the team's score and thus likely minimizes overlap between team members.

Kantor and Rosenwein (1992) introduce the OP with time windows (OPTW), which can be directly extended to the TOP with time windows (TOPTW). In the OPTW, each node i is associated with at least one time window $[e_i, d_i]$ in which that node may be visited. Visiting the node outside of the associated time window would lead to an infeasible solution or, in the case of an early arrival, a possible waiting timer.

A recent proposal by Vincent et al. (2019) presents the TOPTW with time-dependent scores (TOPTW-TDS) as a practical extension of the TOPTW, allowing for nodes to provide different scores depending on the time when they are visited. The authors explain the popularity of the TOPTW in the tourist trip design problem (TTDP) and suggest that time-dependent scores are a more realistic representation of the TTDP and other applications given that tourist landmarks are often more or less attractive at different times of day. They formulate an integer linear programming model and solve it with a hybrid artificial bee colony (HABC) algorithm, created by combining the simulated annealing (SA) technique with the artificial bee colony algorithm developed by Karaboga and Basturk (2007).

2.3 Assignment Problems and Approximate Dynamic Programming

Pentico (2007) provides a survey of AP variations dating back to the introduction of an effective linear programming solution methodology by Kuhn (1955). The classical AP seeks to find a one-to-one assignment of agents to tasks such that the total cost associated with the assignments is minimized. Similarly, in the ABM Problem, there exist n agents needing to be assigned to intercept m targets. However, m varies stochastically over time, and a high-quality assignment policy must consider the dynamic nature of the problem in order to minimize the expected total cost.

Rettke et al. (2016) examine a complex implementation of the AP wherein aerial medical evacuation (MEDEVAC) agents must be prioritized and dispatched to support service calls during high-intensity combat operations. Calls arrive stochastically over time with varying priority levels and must be serviced in accordance with timelines given in the Army Medical Evacuation Field Manual (Department of the Army, 2019). The problem state space is defined with information on each MEDEVAC agent's status (idle or busy), the list of queued service calls, and any presently ar-

riving service call. This formulation shares many similarities with the state space requirements of the ABM Problem, although the ABM Problem requires additional detail to define a precise location of each entity within a geographic region. The authors find that the MDP solution approach is computationally intractable and they implement an ADP approach using approximate policy iteration with least squares temporal differences (API-LSTD) to attain solutions. The API algorithm consists of a policy evaluation phase wherein the algorithm approximates the value function for a fixed policy using LSTD learning. The algorithm then enters a policy improvement phase and creates a new policy based on the updated value function approximation. The authors find that the task assignment policy produced by the API-LSTD algorithm is improved over the baseline closest-available assignment policy, and that the most significant problem feature is the MEDEVAC agents' speed.

Jenkins et al. (2021) describe a military MEDEVAC problem similar to Rettke et al. (2016) but compare API-LSTD with a proposed, improved ADP solution methodology using approximate policy iteration with a neural network (API-NN). The authors use a feed-forward neural network with one hidden layer and find that this approach produces significantly improved results over API-LSTD.

Summers et al. (2020) explore a specific type of AP known as the weapon target assignment problem (WTAP), wherein they investigate the defense of a number of assets from incoming theater ballistic missiles using a collocated air defense system. In a static representation of this problem, the air defense system considers a single salvo of incoming theater ballistic missiles and assigns the air defense system to intercept them appropriately. However, a dynamic representation with multiple salvos arriving stochastically over time is more realistic, but computationally intractable to solve exactly. Thus, the authors employ an API-LSTD implementation to find approximate, high-quality solutions to a dynamic WTAP. They perform extensive computational

experiments and find that the API-LSTD policy in all problem instances outperforms the baseline policies currently in use by the United States Army.

III. Problem Description

This chapter describes the ABM Problem wherein friendly airborne entities (e.g., UCAVs) are tasked to defend a central asset from incoming hostile airborne forces (e.g., cruise missiles). A decision authority, such as the RADC and staff of air battle managers, is responsible for sequentially determining how friendly forces are tasked. The effectiveness of the friendly forces is determined by their ability over time to maintain air superiority by successfully targeting and intercepting the hostile forces that encroach upon the defended airspace.

Targeting is the process of selecting and prioritizing targets and matching the appropriate response to them, considering operational requirements and capabilities (Department of Defense, 2018*a*). Targets that are both fleeting and critical constitute the most significant targeting challenge to the joint force (Joint Targeting School, 2017); in DCA operations, nearly all targets pose a critical threat with a narrow time window to intercept them. Because of the time-sensitive nature of defensive operations, effective DCA requires streamlined coordination and decision-making processes (Department of Defense, 2018*b*). Although the concept of UCAVs in warfighting can be traced back to the mid-nineteenth century (Buckley and Buckley, 1999), the use of autonomous UCAVs is more recent. Autonomous UCAVs developed by the United States must undergo rigorous verification, validation, test, and evaluation procedures before being considered for use in combat. Because of the relatively nascent use of UCAVs and the need for additional testing before the use of *completely* autonomous UCAVs is tenable, target selection during DCA operations with autonomous UCAVs must be approved by an authorized human operator (Department of Defense, 2012).

In this formulation of the ABM Problem, friendly forces consist of two or more UCAVs defending an airspace against incoming cruise missiles that arrive via a stationary Poisson process and proceed towards a defended asset. The defended asset

may be the static location of friendly ground forces, such as a forward military base, or it may be defined by a region encompassing more than one asset being protected. For the problem examined herein, the asset can be stationary or moving, so long as its velocity is constant. That is, we assume all entities move in an inertial reference frame around the asset, and reasonable constraints on speed with respect to the surrounding environment are not violated. As such, the realization of the defended asset could be extended to entities such as a convoy of ground forces or an aircraft carrier. Additionally, there exist sensor platforms in the vicinity of the defended asset capable of detecting incoming cruise missiles and relaying their location to the decision authority; we assume these sensors operate correctly and successfully.

The decision authority must identify and implement the best ABM-related course of action at each of three key moments: first, whenever a cruise missile is detected by a sensor platform; second, whenever a UCAV completes an intercept action and is available for reassignment; and third, whenever a cruise missile impacts the defended asset. When no active threats exist, the UCAVs operate in accordance with the published airspace control plan (ACP) and orbit assigned DCA combat air patrol (CAP) locations. Upon cruise missile detection, the decision authority determines the location and velocity of the threat and characterizes the missile type to estimate the relative level of threat it poses. As this information pertains to the problem examined herein, we assume the cruise missiles have homogeneous offensive capabilities and move at a constant velocity. After threat detection and identification, the decision authority must consider the current location of all friendly UCAVs and determine a response for each, which may include relaying to each UCAV a target intercept location or waypoint for defensive posturing. Upon detection of multiple cruise missiles, the decision authority must consider the location and velocity of each missile and prioritize this response accordingly. After all threats have been neutralized, the

decision authority must determine where to position each UCAV in accordance with the ACP.

A defensive ABM system endeavors to minimize the expected damage to the defended asset posed by adversary threat capabilities. The ABM system must assign UCAVs to incoming targets sequentially over time to achieve this objective, maximizing expected total discounted reward. The ABM Problem is both dynamic and stochastic in that cruise missile arrival times and locations are not known *a priori*. The decision authority must characterize the stochastic arrival behavior of the enemy based on past intelligence-gathering activities and determine how to task UCAVs to maintain the most effective defensive posture over time.

IV. Methodology

4.1 MDP Formulation

This section describes the MDP formulation of the ABM Problem. To minimize expected damage to an asset defended by a fleet of friendly UCAVs, any hostile forces detected in the area of operations must be engaged to prevent them the opportunity to attack. A solution to this sequential decision-making problem is represented by a policy that provides a decision-maker with decision rules to employ, given any possible state of the system. The survivability of the defended asset depends on establishing a high-quality policy to govern UCAV actions during DCA operations.

Consider the following situation. Cruise missiles arrive in the area of operations defined by a Cartesian coordinate system over a circular region encompassing a central sensor platform collocated with the defended asset. The radius of the circle represents the sensor's detection horizon, where missiles are detected with certainty upon arrival. Missiles arrive via a stationary Poisson process with rate λ at random radial positions along the horizon governed by a statistical distribution intended to estimate the enemy's expected attack strategy based on intelligence reports. The missiles proceed with a constant velocity towards their target.

When assigned to intercept a target, a UCAV proceeds via a minimum-time intercept trajectory. To maintain a focus on finding high-quality target assignment policies, we assume the UCAVs are uniformly armed with a weapon system that will not be quickly depleted of its offensive means to destroy a target. Additionally, given a reasonable time horizon for our model, we assume that the UCAVs are able to remain airborne without encountering limitations on flight time otherwise imposed by range or maintenance requirements. Upon closing distance with a target with respect

to the UCAV's weapon engagement zone (WEZ), the UCAV destroys the target with a specified probability.

To model the ABM Problem as an MDP, we leverage the modeling framework set forth by Jenkins et al. (2021) for a different application having related asset management features. The problem formulation uses the notation shown in Table 1.

Table 1. Markov Decision Process Model Notation

$\mathcal{T}$	the set of decision epochs, indexed by t
t	a decision epoch; the index element of $\mathcal{T}$
$\mathcal{S}$	the set of all system states, referenced at a specific decision epoch by S_t
S_t	the system state in $\mathcal{S}$ at decision epoch $t \in \mathcal{T}$
τ_t	the system time at decision epoch $t \in \mathcal{T}$
U_t	the UCAV status tuple in S_t
M_t	the missile status tuple in S_t
$\hat{R}_t$	the stochastic information arriving at decision epoch $t \in \mathcal{T}$, realized in S_t
$\mathcal{U}$	the set of UCAVs in the system, indexed by u
u	a UCAV; the index element of $\mathcal{U}$
U_{tu}	the status tuple representing UCAV $u \in \mathcal{U}$ at decision epoch $t \in \mathcal{T}$
$\boldsymbol{v}_{tu}$	the Cartesian position vector in $\mathbb{R}^2$ of UCAV $u \in \mathcal{U}$ at decision epoch $t \in \mathcal{T}$
$\dot{\boldsymbol{v}}_{tu}$	the Cartesian velocity vector in $\mathbb{R}^2$ of UCAV $u \in \mathcal{U}$ at decision epoch $t \in \mathcal{T}$
d_{tu}	the rotational direction indicator of the velocity vector of UCAV $u \in \mathcal{U}$ at decision epoch $t \in \mathcal{T}$
$\mathcal{M}_t$	the set of missiles in the system at decision epoch $t \in \mathcal{T}$, indexed by m
m^{max}	the maximum allowed cardinality of $\mathcal{M}_t$
m	a missile; the index element of $\mathcal{M}_t$
M_{tm}	the status tuple representing missile $m \in \mathcal{M}_t$ at decision epoch $t \in \mathcal{T}$
$\boldsymbol{\rho}_{tm}$	the Cartesian position vector in $\mathbb{R}^2$ of missile $m \in \mathcal{M}_t$ at decision epoch $t \in \mathcal{T}$
$\dot{\boldsymbol{\rho}}_{tm}$	the Cartesian velocity vector in $\mathbb{R}^2$ of missile $m \in \mathcal{M}_t$ at decision epoch $t \in \mathcal{T}$
$\mathcal{K}$	the non-empty set of static CAP locations in the system, indexed by k
k	a CAP location; the index element of $\mathcal{K}$
K_k	the status tuple representing CAP location $k \in \mathcal{K}$
$\boldsymbol{\kappa}_k$	the Cartesian position vector in $\mathbb{R}^2$ of CAP location $k \in \mathcal{K}$

4.1.1 Decision Epochs

The set of decision epochs $\mathcal{T} = \{1, 2, ...\}$ represents the points in time that require a decision regarding the tasking of a UCAV. These decision epochs occur alongside three system events: first, whenever a cruise missile is detected by the sensor platform; second, whenever a UCAV completes an intercept action and is available for reassignment; and third, whenever a cruise missile impacts the defended asset.

4.1.2 State Space

At decision epoch $t \in \mathcal{T}$, the system state $S_t \in \mathcal{S}$ is given by the tuple

$$S_t = (\tau_t, U_t, M_t, \hat{R}_t),$$

wherein τ_t represents the system time, U_t represents the UCAV status tuple, M_t represents the missile status tuple, and $\hat{R}_t$ represents the stochastic information arriving at decision epoch $t \in \mathcal{T}$. It is important to note that, although decision epochs occur at discrete times $(\tau_1, \tau_2, ...)$, the system evolves in continuous time and may occupy any number of states during the time between decision epochs.

The UCAV status tuple, U_t, contains information describing each UCAV in the system at decision epoch $t \in \mathcal{T}$. Specifically, we define

$$U_t = (U_{tu})_{u \in \mathcal{U}} \equiv (U_{t1}, U_{t2}, ..., U_{t|\mathcal{U}|}),$$

wherein $\mathcal{U} = (1, 2, ..., |\mathcal{U}|)$ denotes the set of UCAVs in the system and the tuple U_{tu} contains all necessary information regarding UCAV $u \in \mathcal{U}$ at decision epoch $t \in \mathcal{T}$. We further specify the tuple

$$U_{tu} = (\boldsymbol{v}_{tu}, \dot{\boldsymbol{v}}_{tu}, d_{tu}),$$

wherein at decision epoch $t \in \mathcal{T}$, $\boldsymbol{v}_{tu} \in \mathbb{R}^2$ denotes the Cartesian position vector of UCAV $u \in \mathcal{U}$; $\dot{\boldsymbol{v}}_{tu} \in \mathbb{R}^2$ denotes the Cartesian velocity vector of UCAV $u \in \mathcal{U}$; and d_{tu} indicates the rotational direction of the velocity vector of UCAV $u \in \mathcal{U}$. The UCAV status tuples maintain information necessary to represent movement according to a two-degree-of-freedom, point-mass aircraft model. We let $d_{tu} \in \{-1, 0, 1\}$ wherein -1 denotes that the UCAV is currently performing a right turn, 0 denotes no turn, and 1 denotes a left turn. Because the UCAV is not subject to human endurance limits, we assume it will always perform turns at the maximum allowable rate until aligning to intercept a target. We apply realistic bounds to $\dot{\boldsymbol{v}}_{tu}$ based on the flight characteristics and the operating flight strength of a particular UCAV MDS. Moreover, an important element of the ABM Problem is the notion of task preemption, wherein a UCAV currently en route to intercept a particular target may be reassigned to intercept a different target as needed. We do not allow for the destruction of UCAVs, nor are UCAVs removed from the system temporarily or permanently for other reasons, so $|\mathcal{U}| > 0, \forall t \in \mathcal{T}$.

In the same manner, we define the missile status tuple M_t, which describes the status of each missile in the system at decision epoch $t \in \mathcal{T}$. Let

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

wherein $\mathcal{M}_t = \{1, 2, ..., |\mathcal{M}_t|\}$ denotes the set of missiles in the system at decision epoch $t \in \mathcal{T}$ with the tuple M_{tm} containing all necessary information to describe each missile. We further specify the tuple

$$M_{tm} = (\boldsymbol{\rho}_{tm}, \dot{\boldsymbol{\rho}}_{tm}),$$

wherein at decision epoch $t \in \mathcal{T}$, $\boldsymbol{\rho_{tm}} \in \mathbb{R}^{\mathbf{2}}$ denotes the Cartesian position vector of missile $m \in \mathcal{M}_t$, and $\dot{\boldsymbol{\rho}}_{tm} \in \mathbb{R}^2$ denotes the Cartesian velocity vector of missile $m \in \mathcal{M}_t$. Because the incoming missiles arrive randomly, the dimension of the state space is a random variable. To maintain the assurance of a finite-dimensional state space, we establish the parameter $m^{max} \in \mathbb{N}$ wherein $|\mathcal{M}_t| \leq m^{max}, \forall t \in \mathcal{T}$. If no missiles exist in the system at decision epoch $t \in \mathcal{T}$, $\mathcal{M}_t = \emptyset$.

We represent the stochastic information arriving at decision epoch $t \in \mathcal{T}$ by the tuple $\hat{R}_t$. Specifically, if a new missile is detected in the system at decision epoch $t \in \mathcal{T}$, $\hat{R}_t$ contains all information necessary to update the system state upon realization of the random variables. If no new missile is detected at decision epoch $t \in \mathcal{T}$, $\hat{R}_t = \emptyset$. Let

$$\hat{R}_t = (\hat{\boldsymbol{\rho}}_t, \hat{\dot{\boldsymbol{\rho}}}_t),$$

wherein the components of $\hat{R}_t$ represent a random realization of the previously defined components of M_{tm}.

Although not formally a component of the state variable because of its static nature, the following CAP information represents an important aspect of the ABM Problem and merits development. The CAP location tuple K contains information regarding static CAP locations orbited by UCAVs not actively intercepting a target. We assume that any number of UCAVs can be assigned to a single CAP location. The set K is written

$$K = (K_k)_{k \in \mathcal{K}} = (K_1, K_2, ..., K_{|\mathcal{K}|}),$$

wherein $\mathcal{K} = \{1, 2, ..., |\mathcal{K}|\} : |\mathcal{K}| > 0$, denotes the non-empty set of static CAP locations in the system, and K_k contains all necessary information to define each CAP location. The set K is necessarily non-empty because we define K_1 in all cases

to be the static location of the defended asset. We define

$$K_k = (\boldsymbol{\kappa}_k),$$

wherein $\boldsymbol{\kappa}_k \in \mathbb{R}^2$ denotes the Cartesian position vector of CAP $k \in \mathcal{K}$.

4.1.3 Action Space

A decision-maker must consider the overall system state to decide how to best assign UCAVs to intercept targets at each decision epoch. The set of all possible decisions while in state S_t is represented by the set

$$\mathcal{X}_{S_t} = \left\{ \left((x_{tum})_{m \in \mathcal{M}_t}, (x_{tuk})_{k \in \mathcal{K}} \right) : \sum_{m \in \mathcal{M}_t} x_{tum} + \sum_{k \in \mathcal{K}} x_{tuk} = 1 \ \forall u \in \mathcal{U} \right\},$$

wherein the constraint

$$\sum_{m \in \mathcal{M}_t} x_{tum} + \sum_{k \in \mathcal{K}} x_{tuk} = 1 \ \forall u \in \mathcal{U}$$

prevents each UCAV $u \in \mathcal{U}$ from being assigned to perform multiple tasks simultaneously. The decision associated with each UCAV represents a set of individual decisions to assign UCAV $u \in \mathcal{U}$ to one of two tasks. First, let $x_{tum} = 1$ if UCAV $u \in \mathcal{U}$ is assigned to intercept missile $m \in \mathcal{M}_t$ at decision epoch $t \in \mathcal{T}$, and 0 otherwise. Second, let $x_{tuk} = 1$ if UCAV $u \in \mathcal{U}$ is assigned to move to CAP $k \in \mathcal{K}$ at decision epoch $t \in \mathcal{T}$, and 0 otherwise. Based on the decision, each UCAV either navigates via the most direct route to a CAP or navigates to the calculated missile intercept location and destroys the target with probability p^{kill} once within the range established by the WEZ.

4.1.4 Transition Probabilities

The ABM system state at each decision epoch $t \in \mathcal{T}$ is determined by the state transition function $S_{t+1} = S^M(S_t, x_t, W_{t+1})$, although the system may transition through any number of states between decision epochs. This transition function indicates that the system state at decision epoch $t+1 \in \mathcal{T}$ is fully determined by the state at decision epoch t, the decision made at decision epoch t, and the information that arrives at decision epoch $t+1$, represented by W_{t+1}.

A central aspect of a system state transition is the method by which each UCAV $u \in \mathcal{U}$ navigates from location to location. We model the UCAV kinematics in a two-degree-of-freedom, point-mass aircraft model. The equations of motion for this model are

$$\upsilon_x' = \upsilon_x + \zeta \cos\Theta,$$

$$\upsilon_y' = \upsilon_y + \zeta \sin\Theta,$$

wherein ζ is the UCAV's speed and Θ is the UCAV's directional heading.

Upon UCAV assignment to intercept a missile, we calculate the most direct intercept locations by first establishing the location of missile $m \in \mathcal{M}_t$ as a function of time elapsed since τ_t. We denote this time difference as δ. Let

$$\boldsymbol{\rho}_{tm}^{future} = \boldsymbol{\rho}_{tm} + \delta \dot{\boldsymbol{\rho}}_{tm}. \tag{1}$$

Recall that each missile $m \in \mathcal{M}_t$ moves with a constant velocity. In the time interval $[\tau_t, \tau_t + \delta]$, the range of UCAV $u \in \mathcal{U}$ is defined by radius r, and we would like to determine the time δ when r is equal to the distance between the UCAV and the

projected location of the missile. This relationship is expressed as

$$r^2 = \left(\delta \frac{\dot{\boldsymbol{v}}_{tu}}{\|\dot{\boldsymbol{v}}_{tu}\|}\right)^2 = (\boldsymbol{\rho}_{tm}^{future} - \boldsymbol{v}_{tu})^2. \tag{2}$$

The variable δ indicates the minimum time until intercept of missile $m \in \mathcal{M}_t$ by UCAV $u \in \mathcal{U}$ and is given by the smallest real solution to the quadratic equation

$$\left(\|\dot{\boldsymbol{\rho}}_{tm}\|^2 - \left(\frac{\dot{\boldsymbol{v}}_{tu}}{\|\dot{\boldsymbol{v}}_{tu}\|}\right)^2\right)\delta^2 + 2\Big((\boldsymbol{\rho}_{tm} - \boldsymbol{v}_{tu}) \cdot \dot{\boldsymbol{\rho}}_{tm}\Big)\delta + \|\boldsymbol{\rho}_{tm} - \boldsymbol{v}_{tu}\|^2 = 0, \tag{3}$$

such that $\delta < \delta^{impact}$, wherein δ^{impact} is the time until missile $m \in \mathcal{M}_t$ impacts the target. Additionally, we define the function

$$Y : (u, m) \mapsto [0, \infty)$$

as the smallest real solution to Equation (3) with respect to UCAV $u \in \mathcal{U}$ and missile $m \in \mathcal{M}_t$. If Equation (3) has no real solutions, UCAV $u \in \mathcal{U}$ is not able to intercept missile $m \in \mathcal{M}_t$ prior to it reaching the defended asset. If the UCAV's heading is closely aligned at decision epoch $t \in \mathcal{T}$ with the required direction to intercept the target, we assume the UCAV can make the directional correction immediately, and the intercept location for tasking is determined directly from Equation (1). However, if the UCAV requires a directional adjustment greater than its maximum turning rate per unit of time, we solve Equation (3) iteratively by projecting the UCAV and missile positions forward in time while the UCAV performs full left and right turns, as appropriate. The smallest real solution to Equation (3) using these projected positions, wherein the UCAV's projected heading aligns closely with the projected intercept location, determines the appropriate intercept location for UCAV tasking via Equation (1).

4.1.5 Rewards and Costs

The system incurs a cost whenever a missile successfully reaches the defended asset. In the case of a missile impact, it is difficult to determine which action or actions taken by the decision-maker ultimately resulted in this event, and at times the decision-maker may make many decisions prior to observing a non-zero cost. This is referred to as a delayed cost, and the difficulty of assigning a cost value to a specific state-action pair is referred to as the credit assignment problem (Sutton and Barto, 2018). We define this cost as

$$C(S_t, x_t) = -\sum_{m \in \mathcal{M}_t} \mathbb{1}(\boldsymbol{\rho}_{tm}, K_1), \tag{4}$$

wherein the indicator function $\mathbb{1} : (\boldsymbol{\rho}_{tm}, K_1) \mapsto \{0, 1\}$ is defined as

$$\mathbb{1}(\boldsymbol{\rho}_{tm}, K_1) = \begin{cases} 1 & \text{if } \|\boldsymbol{\rho}_{tm} - K_1\| \leq \rho_{tm}^{impact}, \\ 0 & \text{otherwise.} \end{cases} \tag{5}$$

The scalar ρ_{tm}^{impact} indicates the distance at which missile $m \in \mathcal{M}_t$ may impact the defended asset at position K_1 with a significant probability of damage. Given a cruise missile impact on the defended asset, damage expectancy calculations are outside the scope of this research; we assume that a missile $m \in \mathcal{M}_t$ within its impact range $\boldsymbol{\rho}_{tm}^{impact}$ poses a homogeneous level of threat to the defended asset regardless of the precise impact location. The reward function calculates the total cost of all missiles impacting the defended asset at decision epoch $t \in \mathcal{T}$, or 0 if no missiles are within their impact range.

4.1.6 Objective Function

This MDP model aims to determine an optimal policy $\pi^* \in \Pi$, which is the optimal sequence of decision rules mapping system states to actions. The optimal policy guides the decision maker by determining the action for any possible system state that maximizes the expected total discounted reward given by the objective function

$$\max_{\pi \in \Pi} \mathbb{E}^{\pi} \left(\sum_{t=1}^{\infty} \gamma^{\tau_t} C(S_t, \pi(S_t)) \right), \tag{6}$$

wherein $\gamma \in [0, 1)$ is the discount factor. As originally established by Bellman (1957), an optimal policy has the property that, whatever the initial state and initial decision are, the remaining decisions must constitute an optimal policy with regard to the state resulting from the first decision. Thus, we determine the optimal policy from the solution to the Bellman equation

$$V(S_t) = \max_{x_t \in X(S_t)} \Big(C(S_t, x_t) + \gamma^{(\hat{\tau}(S_{t+1}) - \tau_t)} \mathbb{E}[V(S_{t+1}) \mid S_t, x_t] \Big), \tag{7}$$

wherein $\hat{\tau}(S_{t+1})$ indicates the time when the system visits state S_{t+1}.

The state space of the ABM Problem is naturally continuous, as the locations and velocities of the system entities must be described over time at a resolution that offers meaningful insight into the behavior of the system. Although the action space of the ABM Problem is discrete, attaining an exact solution to the Bellman equation is computational intractable given the continuous state space. Moreover, given the high dimensionality of the state space, the Bellman equation would be computationally intractable even if the state space were to be discretized. Thus, we propose an approximate dynamic programming (ADP) technique for finding an approximation of $V(S_t)$ to produce a high-quality task-assignment policy.

4.2 ADP Formulation

This section describes an ADP approach for the ABM Problem to formulate a high-quality approximation of the Bellman equations for optimality. The high dimensionality and continuous nature of the state space along with the sparse reward structure require the use of an approximation technique that can uncover complex, nonlinear relationships between system states, actions, and rewards. We employ basis functions in this approximation and we seek to determine problem-specific basis functions that can be used in a linear combination to approximate the Bellman equations.

The creation of problem-specific basis functions to support an ADP algorithm is both an art and a science. It is necessary to leverage empirical knowledge of air combat scenarios to develop basis functions that describe important patterns in the state space. Let $\phi_{f\in\mathcal{F}}(S_t^x)$ represent a basis function wherein $f \in \mathcal{F}$ is a feature of state S_t and $\mathcal{F}$ is the set of features. The first set of basis functions describes the distances of each UCAV $u \in \mathcal{U}$ from its assigned target $m \in \mathcal{M}_t$:

$$\phi_f(S_t) = \|\boldsymbol{\upsilon}_{tu} - \boldsymbol{\rho}_{tm}\|. \tag{8}$$

The second set of basis functions describes the distances between the defended asset (located at the origin of the coordinate system) and the assigned target $m \in \mathcal{M}_t$ of UCAV $u \in \mathcal{U}$:

$$\phi_f(S_t) = \|\boldsymbol{\rho}_{tm}\|. \tag{9}$$

The third set of basis functions describes the average distance of a UCAV's assigned target, missile $m_j \in \mathcal{M}_t$, from all other missiles in $\mathcal{M}_t$:

$$\phi_f(S_t) = \begin{cases} \frac{1}{|\mathcal{M}_t|-1} \sum_{i=1, i\neq j}^{|\mathcal{M}_t|} \|\boldsymbol{\rho}_{tm_i} - \boldsymbol{\rho}_{tm_j}\|, \text{for } i, j \in \{1, 2, ..., |\mathcal{M}_t|\} & \text{if } |\mathcal{M}_t| \geq 2, \\ 0 & \text{otherwise.} \end{cases} \tag{10}$$

The fourth set of basis functions describes the distance between each UCAV $u \in \mathcal{U}$ and its assigned target's intercept location, calculated in Equation (3):

$$\phi_f(S_t) = \|\boldsymbol{\upsilon}_{tu} - Y(u, m)\|. \tag{11}$$

The fifth and final set of basis functions describes the distance between the origin and the calculated intercept location of the assigned target $m \in \mathcal{M}_t$ of UCAV $u \in \mathcal{U}$:

$$\phi_f(S_t) = \|Y(u, m)\|. \tag{12}$$

We employ an approximate policy iteration technique using least squares temporal differences (API-LSTD) using a implementation similar to Rettke et al. (2016), Davis et al. (2017), Jenkins et al. (2021), McKenna et al. (2020), and Summers et al. (2020). The architecture of the API-LSTD algorithm is defined in part by several tunable hyperparameters. These hyperparameters differ from other system parameters in that the ideal settings cannot be determined directly from the data and must be discovered by experimentation.

First, we define a learning rate (or smoothing rate) given by the polynomial step-size rule

$$\alpha_g = \frac{1}{g^\alpha}, \tag{13}$$

wherein $\alpha \in (0, 1]$. This hyperparameter determines how new estimates of θ are incorporated into the existing estimate of θ. In all cases, α_g decreases over time as the policy improvement counter g increases, indicating that the model incorporates new information more quickly during the initial simulations, but relies more heavily on the existing model in later simulations. However, the rate at which α_g decreases depends on the setting for α.

Next, we define an exploration-exploitation parameter, ε, given by the polynomial step-size rule

$$\varepsilon_g = \frac{1}{g^{\varepsilon}}, \tag{14}$$

wherein $\varepsilon \in (0, 1]$. A key aspect of reinforcement learning is balancing the trade-off between exploration and exploitation. Exploration refers to the process of discovering new system behavior by taking actions that are potentially different from what the current model may recommend. Conversely, exploitation refers to the incremental refinement of the existing model by following recommended actions. Lower values of ε indicate a higher tendency towards exploration.

Finally, we define a regression regularization parameter, $\eta > 0$. Because our API-LSTD implementation uses linear regression analysis, we need to ensure that we are working with non-singular matrices, and it is often very difficult to avoid multicolinearity in high-dimensional data matrices. In linear regression, the covariance matrix $X^\intercal X$ is naturally positive semidefinite, so there is no guarantee that $(X^\intercal X)^{-1}$ exists. However, by adding a regularization component, we ensure that $X^\intercal X + \eta I$ is positive definite and thus $(X^\intercal X + \eta I)^{-1}$ will always exist. This characteristic of the perturbed covariance matrix is a consequence of the fact that b is an eigenvalue of $X^\intercal X$ if and only if $b + \eta$ is an eigenvalue of $X^\intercal X + \eta I$. Because $b \geq 0$ and $\eta > 0$, all eigenvalues of $X^\intercal X + \eta I$ are positive, thus the matrix is non-singular. However, regularization comes at the cost of introducing bias into the regression estimates.

V. Testing, Analysis, and Results

In this chapter, we develop a generic yet representative scenario to demonstrate our ADP approach using an implementation of the API-LSTD algorithm. We design computational experiments to conduct sensitivity analyses on how algorithm hyperparameter settings and certain numeric aspects of the ABM Problem, such as the cruise missile arrival rate, affect the API-LSTD algorithm's ability to find high-quality solutions. Recall that ideal hyperparameter settings cannot be determined outside of problem-specific experimentation. The processing system for this experiment uses an Intel Core i7-9700k with 8 cores at 5.2GHz and 32GB RAM. We implement the API-LSTD algorithm in MATLAB R2020b and use MATLAB's Parallel Computing Toolbox alongside built-in functionality for solving large systems of linear equations via matrix inverse operations.

5.1 Representative Scenario

To effectively implement the ABM Problem in a realistic environment, we develop a generic yet representative scenario wherein the United States (US) military is conducting combat operations. We examine the case of defending a forward operating base (FOB) wherein intelligence reports indicate that enemy forces intend to attack the base by employing a large arsenal of ground-launched cruise missiles. The system incurs a cost when cruise missiles impact the defended asset. The quality of a policy is represented by the expected total discounted reward generated by the system under that policy compared to the same metric under a myopic or benchmark policy.

To model cruise missile attacks, we use a stationary Poisson process (PP) with rate parameter $\lambda = \frac{1}{30}$, indicating exponentially distributed interarrival times and a mean interarrival time of 30 seconds. Observations from trial runs inform this parameter

level selection, as less frequent arrivals often result in a sequence of trivial, single-entity assignment problems, whereas a mean interarrival time of 30 seconds creates a scenario that requires some amount of intelligent and proactive decision-making for success. Additionally, more rapid arrivals allow for an efficient evaluation of defensive performance without necessitating a simulation horizon that would introduce constraints such as fuel, weapons load, or periodic maintenance requirements.

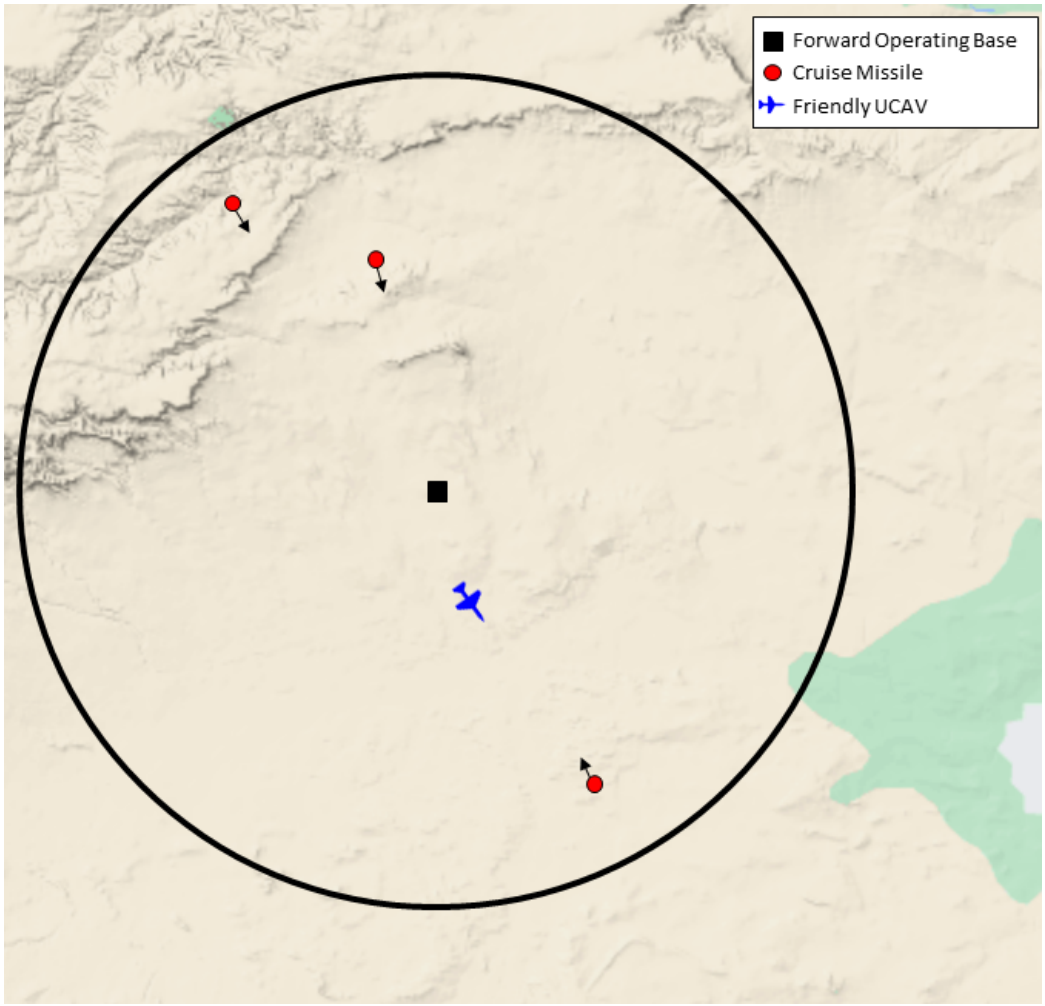

Figure 1. Scenario Representation

A notional geographic representation of the scenario environment is shown in Figure 1. We establish a circular region with a radius of 15 miles in which incoming cruise missiles are detected with certainty when they arrive according to the PP(λ)

at the region boundary. The missiles arrive over time from any direction with equal probability and proceed with a constant velocity of 500 miles per hour towards the FOB, impacting the FOB in approximately 100 seconds if not intercepted. The UCAVs defend the FOB, also moving with a constant velocity of 500 miles per hour, and maneuver according to a two-degree-of-freedom model with a turning rate of 11.25 degrees per second. If a UCAV is assigned to intercept a particular cruise missile and is able to reach a position within one-half of a mile of the missile regardless of relative velocity, the UCAV intercepts the cruise missile with probability $p^{kill} = 1$.

5.2 Simulation Environment

To accurately characterize how this dynamic system evolves over time, we develop and implement a modular simulation system used by the API-LSTD algorithm to find high-quality solutions. As shown in Figure 2, the simulation process consists of four primary objects: the ABM environment object, ABM entity objects, the simulation process, and the API-LSTD process.

The ABM environment object is the primary data interface between all objects and is central to the simulation process. It establishes the overall parameters for the environment and handles the creation and destruction of all ABM entity objects. The ABM environment object maintains event timing for the exponentially distributed interarrival times of the cruise missiles. As the simulation progresses, it records positional updates for each entity, determines entity ranges from their targets, triggers intercept or missile impact events, and handles the target assignment process.

Within the ABM environment exists a collection of ABM entity objects that represent all UCAVs and cruise missiles present in the simulation. The ABM entities maintain entity-unique parameters such as position and velocity. The ABM entities also carry out all calculations for determining target intercept trajectories.

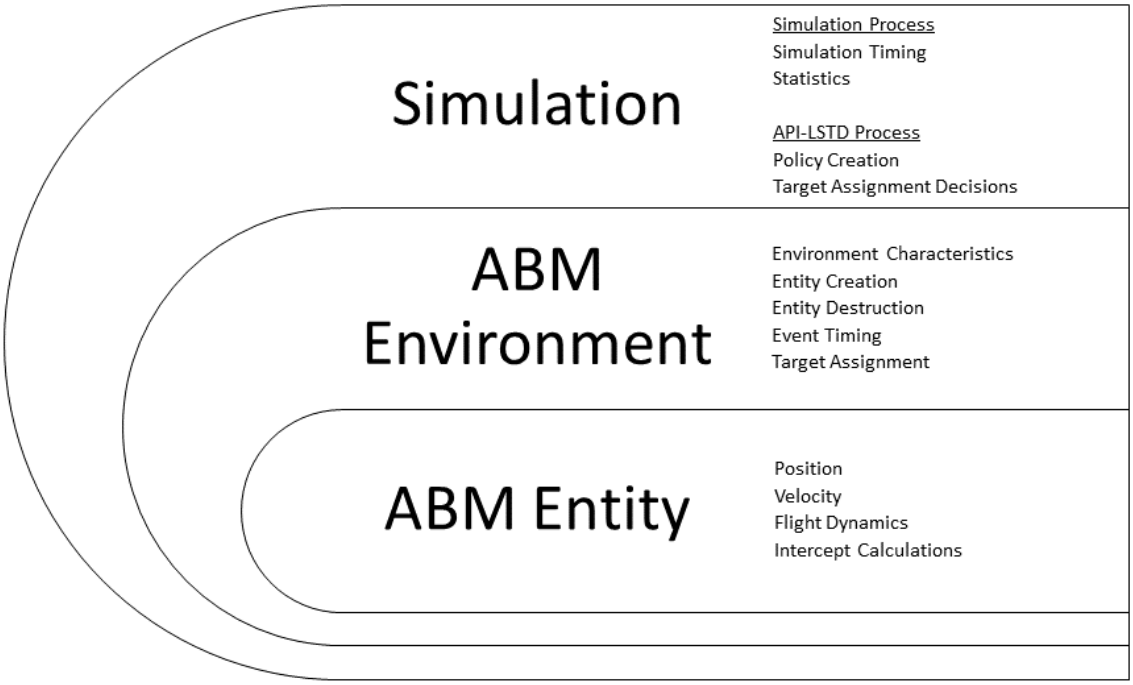

Figure 2. Simulation Class Hierarchy Diagram

External to the ABM environment, the simulation process advances the simulation clock, determines when decisions are necessary, invokes the API-LSTD process to determine appropriate decisions, and passes those decisions to the ABM environment object. The simulation process flow is shown in Figure 3.

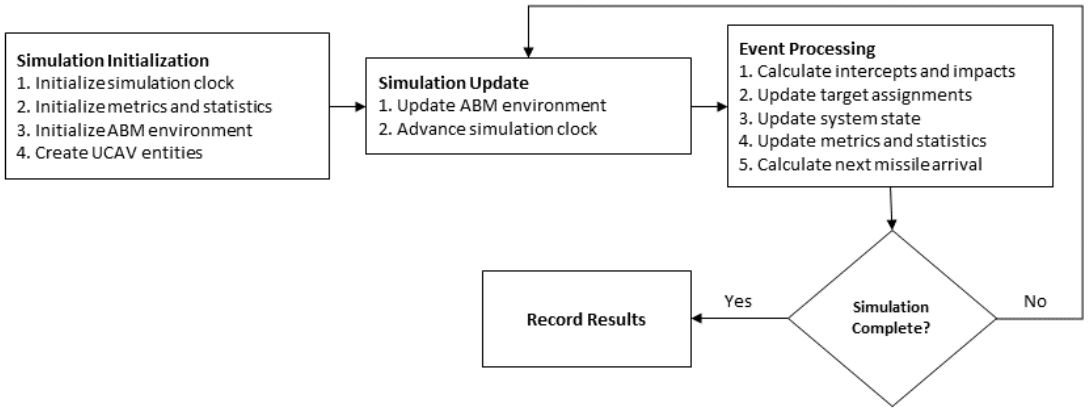

Figure 3. Simulation Flow Diagram

5.3 Computational Experiments - Hyperparameters

In this section, we design a sequential experiment to screen and refine API-LSTD hyperparameter settings. In the first phase of the experiment, we investigate a wide range of settings to confirm which hyperparameters appear to be significant predictors of API-LSTD algorithm performance when applied to the ABM Problem. Table 2 presents the factors and factor levels for this experiment.

Table 2. Initial Experiment Factors

Factor Name	Factor	Levels	Type
Learning Parameter	α	0.1, 0.5, 0.9	Continuous
Exploration Parameter	ε	0.1, 0.5, 0.9	Continuous
Regularization Parameter	η	0.0001, 0.5, 1	Continuous
Basis Function Level	F	1a, 1b, 2, 3	Categorical

The four levels of Factor F represent the inclusion in the model of first-order basis functions only; first order basis functions with two-factor interactions; second-order basis functions; and third-order basis functions, respectively.

Although the ADP algorithm seeks to find a policy that maximizes expected total discounted reward, this value difficult to interpret. Thus, we develop a proxy dependent variable that has a strong correlation to observed total discounted reward, but that is much easier to interpret and explain. The dependent variable is defined as the mean percentage of UCAV successful intercept actions over $G = 400$ simulation runs. During simulation run g, let Ω_g denote the number of successful intercept actions and ω_g denote the number of missile impacts. We define the UCAV success rate as

$$J = \frac{1}{G} \sum_{g=1}^{G} \frac{100\Omega_g}{\left(\Omega_g + \omega_g\right)}. \tag{15}$$

The API-LSTD algorithm performs 400 policy improvement iterations with a 1000-second, trajectory-following simulation for policy evaluation. Once the algorithm terminates, we evaluate the resulting policy again using 400 repetitions of a

1000-second, trajectory-following simulation. Our computational experiment implements a full factorial experimental design with five overall replications resulting in 108 experimental runs and requiring approximately 30 hours of computation time. Table 3 reports the results, sorted in order of decreasing mean(J).

Table 3. Initial Hyperparameter Experiment Results

Run	α	ε	η	F	max(J)	mean(J)	var(J)
70	0.5	0.9	1	1b	96.08	95.73	0.05
30	0.5	0.9	1	1a	95.74	95.59	0.01
72	0.5	0.9	0.5	3	95.86	95.49	0.13
69	0.5	0.9	0.5	1b	95.60	95.32	0.07
67	0.5	0.9	1	3	95.72	95.32	0.07
71	0.5	0.9	1	2	95.51	95.28	0.04
66	0.5	0.9	0.5	1a	95.87	95.27	0.25
106	0.1	0.9	1	1b	95.95	95.27	0.69
107	0.5	0.9	0.5	2	95.86	95.26	0.15
105	0.9	0.9	0.5	2	95.39	95.25	0.01
⋮				⋮			⋮
15	0.1	0.5	0.0001	2	71.91	54.82	137.90
2	0.1	0.1	0.0001	1a	63.23	53.42	47.06
3	0.1	0.1	0.0001	2	61.56	52.06	127.29
40	0.5	0.1	0.0001	3	57.17	50.79	55.39
38	0.5	0.1	0.0001	1b	52.81	43.87	38.49
4	0.1	0.1	0.0001	3	61.30	43.51	333.33
76	0.9	0.1	0.0001	3	44.97	42.92	2.54
74	0.9	0.1	0.0001	1b	41.39	38.73	5.59
75	0.9	0.1	0.0001	2	39.53	35.86	9.94
39	0.5	0.1	0.0001	2	53.37	33.98	161.24

The results of the initial experiment provide insight into effective API-LSTD hyperparameter settings for the ABM Problem. The most successful policies appear robust, exhibiting low variance, whereas the least successful policies show variance several orders of magnitude higher, indicating policies that are not robust to a variety of stochastic realizations of the problem. A multiple linear regression metamodel of the experimental results shows that all factors with the exception of F are statistically significant predictors of API-LSTD algorithm performance based on this data set. Similarly, we observe the significance of the various basis functions by examin-

ing the magnitude of their coefficients. A graphical depiction of the basis function coefficients is shown in Figure 4.

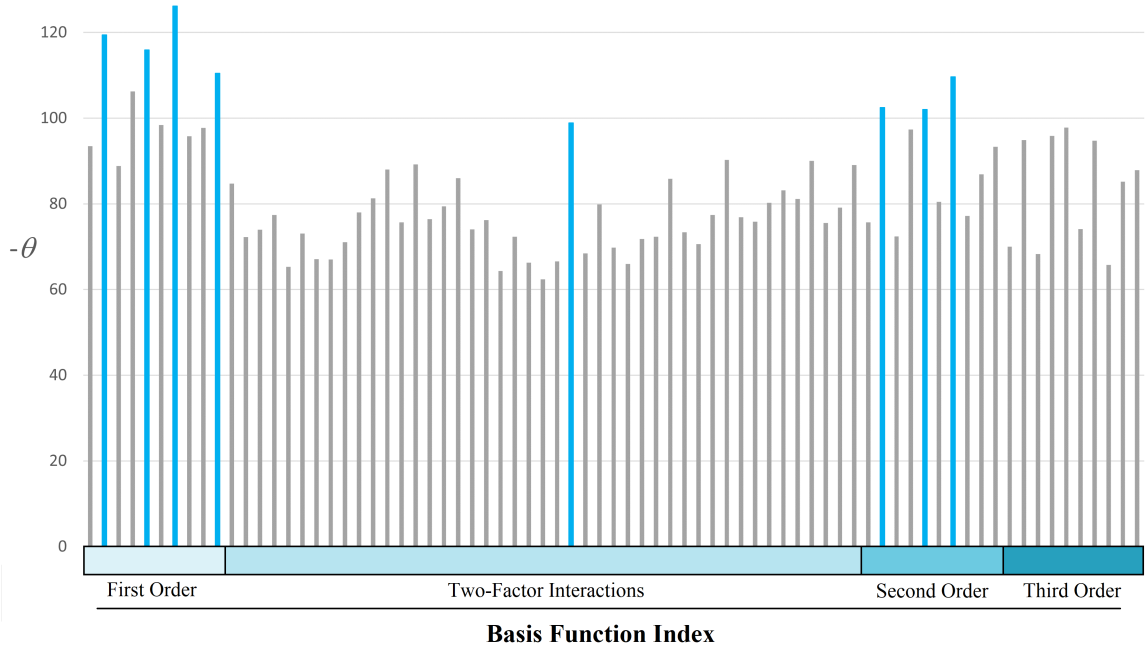

Figure 4. Basis Function Coefficients

Larger magnitudes of basis function coefficients indicate that the particular basis function or interaction is more significant in predicting the value of a given system state. Highlighted in Figure 4 are four first-order basis functions that are of noticeably higher magnitude than the others within the same category. These results correspond to the second and fifth basis functions for each of the two UCAVs. Recall that the second set of basis functions describes the distance of a UCAV's assigned target to the defended asset, and the fifth set of basis functions describes the distance of the calculated intercept location of a UCAV's target to the defended asset. Three of the four of these coefficients also appear significant among the second-order coefficients. A single two-factor interaction has a coefficient that is noticeably larger in magnitude than all other interactions, which is the interaction between the first UCAV's third basis function and the second UCAV's fourth basis function. Recall that the

third set of basis functions describes the average distance of a UCAV's target to all other targets, and the fourth set of basis functions describes the distance between each UCAV and its assigned target's calculated intercept location. Interestingly, the reverse interaction is not of notable magnitude. This analysis of basis function coefficient magnitudes assists the future development of new basis functions as well as the refinement of existing basis functions to better predict the value of any particular system state. Further investigation into the significance of individual basis functions and basis function interactions with the intent to remove less significant terms is outside of the scope of this research but is acknowledged to be potentially valuable in improving computational efficiency by reducing the size of the basis function covariance matrix.

For this initial experiment, with the factor levels chosen to span nearly the entire allowable range of the factors, it would be unusual for the experiment to predict the optimal ADP algorithm parameter settings. Thus, we use the results of this initial experiment to inform selection of the factor levels for a follow-on optimization experiment. A linear metamodel predicts optimal settings of $\alpha = 0.66, \varepsilon = 0.9, \eta = 0.47$, and $F = 3$. We design a more focused experiment wherein we vary the factor levels over a smaller range to achieve a more accurate metamodel. Because F is shown to be a statistically insignificant predictor at the factor levels chosen, we fix $F = 3$. Table 4 presents the factors and factor levels for the optimization experiment.

Table 4. Hyperparameter Optimization Experiment Factors

Factor Name	Factor	Levels	Type
Learning Parameter	α	0.4, 0.6, 0.8	Continuous
Exploration Parameter	ε	0.75, 0.85, 0.95	Continuous
Regularization Parameter	η	1, 0.1, 0.01, 0.001	Continuous

Applying the same policy improvement and policy evaluation construct, we perform five overall replications for a total of 36 experiment runs requiring approximately 10 hours of computation time. Table 5 reports the results of the optimization exper-

iment, sorted in order of decreasing mean(J). We observe minor improvements in mean success rate from the initial experiment shown in Table 3. In particular, the superlative parameter settings offers both an increase in mean success rate and a decrease in variance when compared to the initial experiment.

Table 5. Hyperparameter Optimization Experiment Results

Run	α	ε	η	max(J)	mean(J)	var(J)
21	0.6	0.95	1	95.91	95.81	0.03
9	0.4	0.95	1	95.91	95.69	0.10
33	0.8	0.95	1	95.54	95.42	0.03
34	0.8	0.95	0.1	95.49	95.37	0.01
17	0.6	0.85	1	95.56	95.34	0.08
23	0.6	0.95	0.01	95.51	95.33	0.02
22	0.6	0.95	0.1	95.58	95.31	0.13
36	0.8	0.95	0.001	95.44	95.30	0.05
19	0.6	0.85	0.01	95.42	95.30	0.03
10	0.4	0.95	0.1	95.50	95.23	0.11
⋮				⋮		⋮
20	0.6	0.85	0.001	94.36	94.15	0.03
4	0.4	0.75	0.001	95.42	94.06	2.86
8	0.4	0.85	0.001	94.95	94.00	0.74
7	0.4	0.85	0.01	95.24	93.74	3.09
3	0.4	0.75	0.01	92.85	92.78	0.01

Analyzing a multiple linear regression metamodel created using the data from the hyperparameter optimization experiment indicates statistical significance for main factors ϵ and η along with the $\alpha \cdot \eta$ interaction and second-order terms for α and η. Despite the apparent statistical insignificance of the α term, we retain it in the metamodel for the principle hierarchy. Otherwise, we remove statistically insignificant terms for a 95% confidence level. Table 6 reports the regression metamodel coefficients. The metamodel predicts the optimal hyperparameter settings of $\alpha = 0.62, \varepsilon = 0.95$, and $\eta = 0.57$. We use these hyperparameter settings to formally compare the performance of the benchmark policy with the API-LSTD-generated policy.

Table 6. Multiple Linear Regression for Hyperparameter Optimization

Term	Parameter Estimate	Standard Error	t Ratio	$\mathbb{P} > \|t\|$
Intercept	96.23	0.35	275.55	< 0.01
α	0.08	0.07	1.16	0.25
ε	0.39	0.06	6.11	< 0.01
η	0.36	0.07	5.59	< 0.01
$\alpha \cdot \eta$	-0.24	0.08	-3.21	< 0.01
α^2	-0.27	0.11	-2.39	0.02
η^2	-1.18	0.36	-3.25	< 0.01

5.4 ADP and Benchmark Policy Comparison

The API-LSTD policies in the previous section appear effective in general, but we have not yet established the effectiveness of a competing benchmark policy. We develop an intuitive and easily implemented benchmark policy wherein each UCAV will be assigned to intercept the missile with the closest calculated intercept location. Moreover, multiple UCAVs will not be assigned to intercept the same missile. The benchmark policy allows for task preemption, wherein a UCAV may be reassigned to intercept a newly detected missile with a calculated intercept location that is closer than its current destination. To compare policies, we perform five overall replications of four hundred 1000-second simulations using the benchmark policy and the same five replications using the policy generated by the API-LSTD algorithm, with hyperparameters set in accordance with the multiple linear regression metamodel predictions from the previous section. Table 7 reports the results of this comparison.

Table 7. ADP and Benchmark Policy Performance Comparison

Policy	Success Rate	Success Rate $> 90\%$	Success Rate $> 95\%$	Success Rate $= 100\%$	UCAV Idle Time
Benchmark	$89.7\% \pm 0.2\%$	$47.8\% \pm 1.5\%$	$19.8\% \pm 1.0\%$	$6.4\% \pm 0.7\%$	$22.2\% \pm 0.4\%$
API-LSTD	$95.8\% \pm 0.2\%$	$89.4\% \pm 1.3\%$	$59.6\% \pm 3.0\%$	$26.8\% \pm 1.3\%$	$11.9\% \pm 0.3\%$
ADP Improvement	6.82%	87.03%	201.52%	317.97%	-46.51%

The API-LSTD-generated policy offers statistically significant improvements in several key metrics. First, the ADP solution significantly improves mean success rate, ensuring that more targets are successfully intercepted during any stochastic realization of the ABM Problem. Second, the ADP solution immensely improves the frequency of achieving a greater-than-90% and higher success rate. Of specific note is the 318% increase in frequency that the policy will perform equivalently to an optimal policy, defending the FOB with 100% successful intercepts and no cruise missile impacts. Finally, the increased success rates of the API-LSTD policy come at a cost of decreased UCAV idle time, which roughly translates to increased fuel costs. A more detailed examination of UCAV idle time is presented in Section 5.9.

5.5 ADP and Benchmark Policy Behavior Analysis

The statistically significant improvement in the API-LSTD policy's success rate is a salient finding, but it is important to address what specific behaviors the API-LSTD policy exhibits that are responsible for this improvement. This section illustrates the behavioral differences between the API-LSTD and benchmark policies in two scenarios. Specifically, we subject a pair of UCAVs to a selection of cruise missile arrival patterns and observe how the policies prioritize intercept actions.

The first scenario is shown in Figures 5 and 6, wherein three cruise missiles approach from the southwest while one approaches from the northeast. It is feasible for the two UCAVs to intercept all four cruise missiles, but the order in which the UCAVs intercept the missiles is paramount. Under both policies, as shown in Figures 5(a) and 6(a), we observe that one UCAV is tasked to intercept the northeast cruise missile and the other is tasked to intercept the closest missile to the southwest. After the southwest intercept action is complete, the benchmark policy in Figure 5(b) directs the UCAV to intercept the next-closest target, which is the northernmost of

the two remaining missiles. However, this decision leaves the UCAV out of position to intercept the last missile as shown in Figure 5(c), and the defended asset suffers a missile impact shortly afterwards. Alternatively, in Figure 6(b), we observe that the API-LSTD policy recognizes this danger and instead intercepts the farthest of the two remaining missiles. The UCAV is then positioned to intercept the remaining missile in Figure 6(c), thus successfully defending the asset from all present threats. In more complex environments, controlling the order in which missiles are intercepted to optimize UCAV positioning for subsequent intercept actions becomes increasingly important.

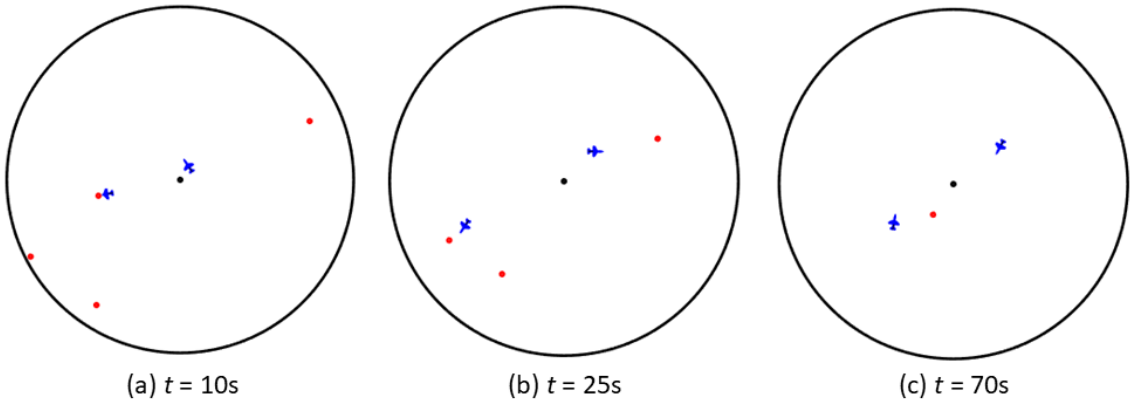

Figure 5. Scenario 1 Benchmark Policy Behavior

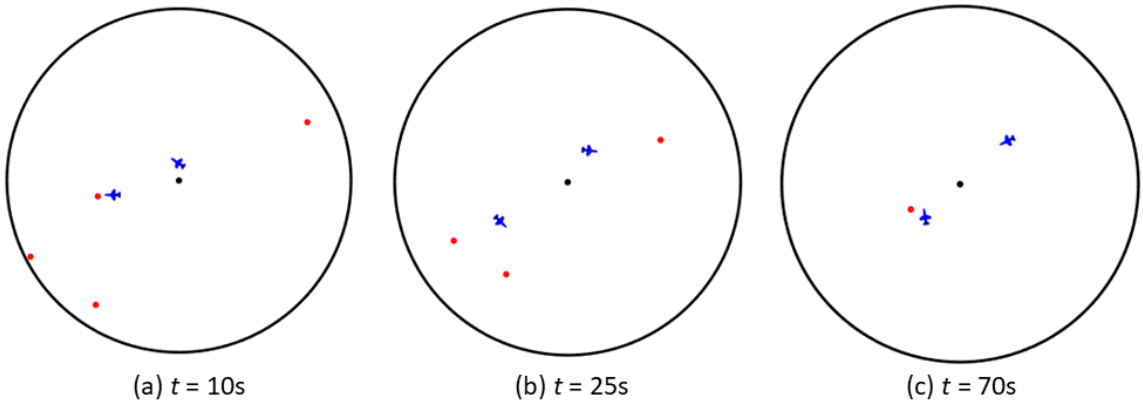

Figure 6. Scenario 1 API-LSTD Policy Behavior

The second scenario is shown in Figures 7 and 8, wherein four cruise missiles approach from each of the cardinal directions, followed by a total of three additional missiles from the east and southeast. It is again feasible for the two UCAVs to intercept all incoming missiles, but intercept positioning is critical. In Figure 7(a), we observe the benchmark policy directing the UCAVs to intercept the cruise missiles approaching from the north and south. However, upon completion of these intercept actions, the UCAVs shown in Figure 7(b) are out of position to intercept the missiles approaching from the east and west, and the defended asset suffers two missile impacts. In Figure 7(c), we observe the UCAVs are able to intercept the remaining three missiles. In contrast, the API-LSTD policy initially provides the same direction for the UCAVs to intercept the missiles approaching from the north and south, but as more missiles are detected, the policy directs the UCAVs to remain close to the defended asset. The cruise missile approaching from the west shown in Figure 8(a) poses the most immediate threat, so the UCAVs intercept that missile first while positioning themselves in Figure 8(b) to intercept the missiles approaching from the north and south. Finally, in Figure 8(c), the UCAVs are positioned to intercept all remaining missiles, and thus the defended asset is protected from all present threats. This scenario illustrates the importance of trajectory optimization, wherein the UCAVs position themselves not only to intercept their current target, but to intercept all follow-on targets as well.

To successfully intercept all missiles in the second scenario, the UCAVs must intercept them much closer to the defended asset to maintain proper positioning for subsequent intercepts. There are benefits in waiting to intercept missiles related to decreasing the size of the region that is actively being defended, but it is reasonable to assume decision-makers may prefer engagements farther away. We investigate this behavior in particular in Section 5.8.

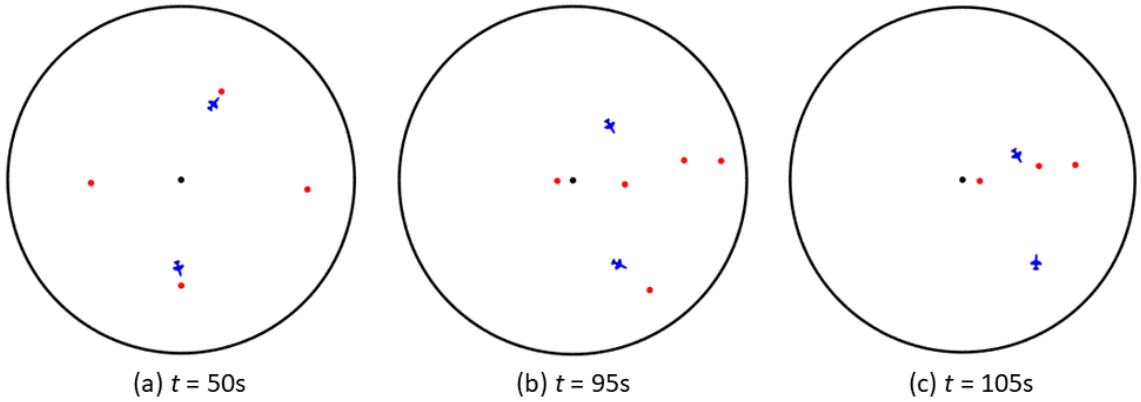

Figure 7. Scenario 2 Benchmark Policy Behavior

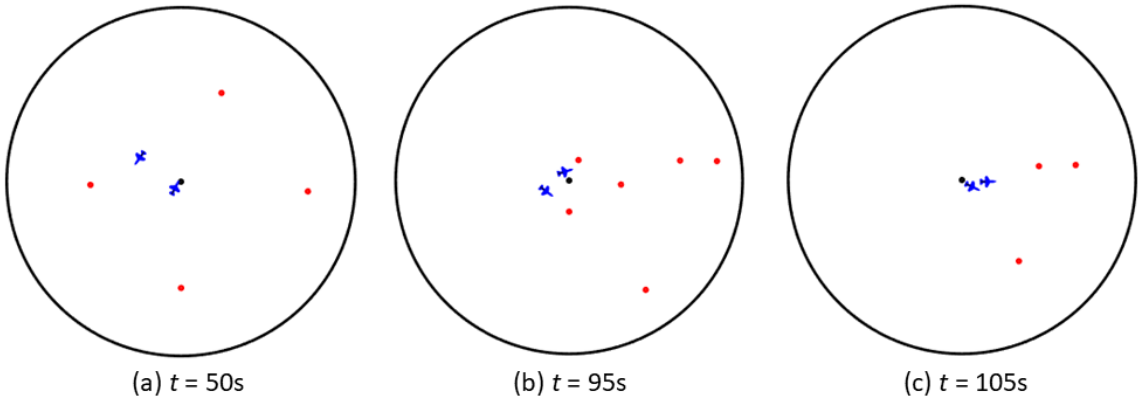

Figure 8. Scenario 2 API-LSTD Policy Behavior

5.6 Problem Environment Sensitivity Analysis

In this section, we design an additional computational experiment to determine how changes to the ABM Problem environment, such as enemy arrival rate, affect the performance of both the benchmark policy and the API-LSTD-generated policy. In this experiment, we vary the cruise missile arrival rate λ by $\pm 50\%$ of the baseline scenario value of $\frac{1}{30}$. Moreover, we modify the relative speed of the UCAV and cruise missile entities by varying the UCAV speed by $\pm 20\%$ of the baseline scenario value of 500 miles per hour. Finally, we introduce a cruise missile arrival asymmetry factor, ψ, where $\psi = 0$ indicates that cruise missiles arrive with equal probability from any direction as in the baseline scenario, and $\psi = 1$ indicates an asymmetric

arrival pattern wherein 70% of the arriving cruise missiles are detected in the northern hemisphere and the remaining 30% are detected in the southern hemisphere. Table 8 details the factor levels for this full-factorial experiment. For each of the 36 policy and factor combinations, we perform five replications of four hundred 1000-second simulations.

Table 8. Sensitivity Analysis Experiment Factors

Factor Name	Factor	Levels	Type
Mean Interarrival Time	$\frac{1}{\lambda}$	20, 30, 60	Continuous
UCAV Speed	v	400, 500, 600	Continuous
Asymmetry Indicator	ψ	0, 1	Categorical

Table 9 displays the results of the sensitivity analysis. The API-LSTD-generated policy shows statistically significant improvement over the benchmark policy in every instance. In particular, the API-LSTD policy shows the largest improvements in scenarios with the fastest cruise missile arrival rate and slowest UCAV speed. In these instances, the UCAVs are at a considerable disadvantage, and it would be expected that success requires more thoughtful decision-making. We also observe that the benchmark policy performs better in the asymmetric cruise missile arrival scenarios in all cases, whereas the API-LSTD policy shows mixed results, sometimes performing better in the symmetric scenarios.

A multiple linear regression metamodel depicts the statistical significance of factors λ, v, and ψ in predicting API-LSTD policy success rate (J). Table 10 reports the parameter estimates for this metamodel. We develop a second-order metamodel including all factor interactions and observe that all main effects are statistically significant predictors of the dependent variable, the policy success rate (J). We remove statistically insignificant terms for a 95% confidence level. Based on the repeated observations in the experiment, we compare the sum of squares due to lack of fit with the sum of squares due to pure error and find no strong evidence of a lack of

Table 9. Sensitivity Analysis Experiment Results for Success Rate (J)

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Success Rate (J)	Benchmark Success Rate (J)	API-LSTD Improvement
1	20	400	1	$87.19\% \pm 0.10\%$	$79.32\% \pm 0.22\%$	9.92%
2	20	400	0	$88.06\% \pm 0.17\%$	$75.21\% \pm 0.22\%$	17.08%
3	20	500	1	$92.07\% \pm 0.10\%$	$83.98\% \pm 0.07\%$	9.64%
4	20	500	0	$91.38\% \pm 0.24\%$	$80.58\% \pm 0.20\%$	13.40%
5	20	600	1	$95.03\% \pm 0.04\%$	$88.51\% \pm 0.12\%$	7.36%
6	20	600	0	$94.01\% \pm 0.09\%$	$85.61\% \pm 0.18\%$	9.81%
7	30	400	1	$92.58\% \pm 0.22\%$	$88.42\% \pm 0.16\%$	4.70%
8	30	400	0	$92.93\% \pm 0.25\%$	$85.00\% \pm 0.29\%$	9.33%
9	30	500	1	$95.76\% \pm 0.13\%$	$91.61\% \pm 0.10\%$	4.53%
10	30	500	0	$95.76\% \pm 0.26\%$	$89.14\% \pm 0.17\%$	7.42%
11	30	600	1	$97.98\% \pm 0.11\%$	$94.45\% \pm 0.11\%$	3.73%
12	30	600	0	$97.48\% \pm 0.15\%$	$92.71\% \pm 0.16\%$	5.15%
13	60	400	1	$95.24\% \pm 0.14\%$	$92.74\% \pm 0.19\%$	2.69%
14	60	400	0	$95.22\% \pm 0.18\%$	$89.81\% \pm 0.31\%$	6.02%
15	60	500	1	$97.29\% \pm 0.17\%$	$94.83\% \pm 0.13\%$	2.59%
16	60	500	0	$97.08\% \pm 0.19\%$	$92.96\% \pm 0.21\%$	4.43%
17	60	600	1	$98.78\% \pm 0.14\%$	$96.84\% \pm 0.09\%$	2.00%
18	60	600	0	$98.71\% \pm 0.07\%$	$95.73\% \pm 0.21\%$	3.12%

fit for this linear metamodel. We observe the metamodel's root mean square error of 0.21 and adjusted R^2 of 0.999, indicating the metamodel predicts the relationship between ABM Problem environmental parameters and API-LSTD policy success rate extremely accurately for this data sample.

Table 10. Multiple Linear Regression for Sensitivity Analysis

Term	Parameter Estimate	Standard Error	t Ratio	$\mathbb{P} > \lvert t \rvert$
Intercept	46.01	0.04	1186.80	< 0.01
λ	5.74	0.03	209.19	< 0.01
v	3.56	0.03	129.76	< 0.01
ψ_0	-1.31	0.02	-58.37	< 0.01
$\lambda \cdot v$	-0.94	0.03	-28.15	< 0.01
$\lambda \cdot \psi_0$	0.28	0.03	10.27	< 0.01
$v \cdot \psi_0$	0.34	0.03	12.34	< 0.01
$\lambda \cdot v \cdot \psi_0$	0.16	0.03	4.02	< 0.01
λ^2	-2.02	0.05	-42.55	< 0.01

To further examine the performance differences between the benchmark policy and the API-LSTD policy, we investigate the frequency at which each policy achieves a success rate (J) at three different levels: success rate greater than 90%, success rate greater than 95%, and success rate equal to 100%. These metrics may be interesting if, for example, there is a minimum tolerance for cruise missile impacts. The 100% level measure is specifically useful if the likelihood of optimal-equivalent performance is important. Tables 11-13 report these results, ordered generally by decreasing problem difficulty, with the most difficult instances (i.e., rapid missile arrivals and slow UCAVs) appearing at the top of the tables.

Table 11. Sensitivity Analysis Experiment Results for Success Rate (J) Frequency > 90%

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Success Rate (J) Frequency > 90%	Benchmark Success Rate (J) Frequency > 90%	API-LSTD Improvement
1	20	400	1	$31.80\% \pm 1.66\%$	$7.00\% \pm 1.06\%$	54.29%
2	20	400	0	$36.95\% \pm 1.67\%$	$1.40\% \pm 0.50\%$	2539.29%
3	20	500	1	$70.25\% \pm 2.12\%$	$19.55\% \pm 1.11\%$	259.34%
4	20	500	0	$63.45\% \pm 2.41\%$	$7.30\% \pm 1.54\%$	69.18%
5	20	600	1	$90.70\% \pm 0.73\%$	$41.35\% \pm 1.70\%$	119.35%
6	20	600	0	$83.05\% \pm 0.78\%$	$27.40\% \pm 1.03\%$	203.10%
7	30	400	1	$71.00\% \pm 1.81\%$	$42.55\% \pm 1.99\%$	66.86%
8	30	400	0	$74.25\% \pm 2.57\%$	$28.05\% \pm 1.44\%$	164.71%
9	30	500	1	$91.60\% \pm 0.59\%$	$65.95\% \pm 1.17\%$	38.89%
10	30	500	0	$89.25\% \pm 1.81\%$	$47.35\% \pm 1.06\%$	88.49%
11	30	600	1	$98.05\% \pm 0.52\%$	$84.15\% \pm 0.87\%$	16.52%
12	30	600	0	$96.35\% \pm 0.29\%$	$73.05\% \pm 1.85\%$	31.90%
13	60	400	1	$88.50\% \pm 0.83\%$	$70.70\% \pm 1.47\%$	25.18%
14	60	400	0	$86.70\% \pm 1.34\%$	$54.15\% \pm 2.37\%$	60.11%
15	60	500	1	$96.80\% \pm 0.42\%$	$84.75\% \pm 1.28\%$	14.22%
16	60	500	0	$94.75\% \pm 0.82\%$	$73.30\% \pm 1.82\%$	29.26%
17	60	600	1	$98.90\% \pm 0.29\%$	$94.80\% \pm 0.78\%$	4.32%
18	60	600	0	$98.75\% \pm 0.27\%$	$88.25\% \pm 1.68\%$	11.90%

We observe that the API-LSTD-generated policy outperforms the benchmark policy in meeting all thresholds more often under every combination of problem environmental factors. Although both policies struggle to achieve consistent results above

Table 12. Sensitivity Analysis Experiment Results for Success Rate (J) Frequency = 95%

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Success Rate (J) Frequency = 95%	Benchmark Success Rate (J) Frequency = 95%	API-LSTD Improvement
1	20	400	1	$6.75\% \pm 0.46\%$	$0.85\% \pm 0.61\%$	694.12%
2	20	400	0	$8.95\% \pm 0.59\%$	$0.15\% \pm 0.20\%$	5866.67%
3	20	500	1	$29.85\% \pm 1.74\%$	$3.55\% \pm 0.95\%$	740.85%
4	20	500	0	$26.45\% \pm 1.11\%$	$1.55\% \pm 0.52\%$	1606.45%
5	20	600	1	$56.10\% \pm 0.67\%$	$13.70\% \pm 0.80\%$	309.49%
6	20	600	0	$47.55\% \pm 1.08\%$	$5.25\% \pm 0.71\%$	805.71%
7	30	400	1	$39.35\% \pm 2.18\%$	$18.05\% \pm 1.19\%$	118.01%
8	30	400	0	$37.90\% \pm 1.34\%$	$10.55\% \pm 0.97\%$	259.24%
9	30	500	1	$63.05\% \pm 1.36\%$	$32.45\% \pm 1.47\%$	94.30%
10	30	500	0	$63.55\% \pm 2.74\%$	$20.95\% \pm 1.45\%$	203.34%
11	30	600	1	$85.95\% \pm 0.80\%$	$51.20\% \pm 1.01\%$	67.87%
12	30	600	0	$79.95\% \pm 2.00\%$	$38.95\% \pm 1.94\%$	105.26%
13	60	400	1	$59.00\% \pm 1.88\%$	$43.55\% \pm 1.73\%$	35.48%
14	60	400	0	$60.00\% \pm 2.31\%$	$27.25\% \pm 2.07\%$	120.18%
15	60	500	1	$78.90\% \pm 2.48\%$	$58.10\% \pm 0.86\%$	35.80%
16	60	500	0	$78.00\% \pm 1.25\%$	$43.80\% \pm 1.92\%$	78.08%
17	60	600	1	$92.25\% \pm 1.53\%$	$75.45\% \pm 1.72\%$	22.27%
18	60	600	0	$91.65\% \pm 0.33\%$	$65.45\% \pm 1.79\%$	40.03%

the thresholds when UCAV speed is reduced in combination with an increased cruise missile arrival rate, the API-LSTD policy shows a larger improvement over the benchmark policy in these more difficult circumstances.

Table 13. Sensitivity Analysis Experiment Results for Success Rate (J) Frequency = 100%

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Success Rate (J) Frequency = 100%	Benchmark Success Rate (J) Frequency = 100%	API-LSTD Improvement
1	20	400	1	$0.65\% \pm 0.12\%$	$0.15\% \pm 0.20\%$	333.33%
2	20	400	0	$0.80\% \pm 0.29\%$	0.00%	N/A
3	20	500	1	$5.60\% \pm 0.70\%$	$0.80\% \pm 0.50\%$	600.00%
4	20	500	0	$4.00\% \pm 0.98\%$	$0.10\% \pm 0.12\%$	3900.00%
5	20	600	1	$17.00\% \pm 1.68\%$	$2.35\% \pm 0.83\%$	623.40%
6	20	600	0	$12.20\% \pm 0.84\%$	$0.80\% \pm 0.29\%$	1425.00%
7	30	400	1	$14.20\% \pm 1.81\%$	$5.85\% \pm 0.65\%$	142.74%
8	30	400	0	$16.50\% \pm 0.38\%$	$2.30\% \pm 0.39\%$	617.39%
9	30	500	1	$31.20\% \pm 2.49\%$	$11.95\% \pm 1.53\%$	161.09%
10	30	500	0	$32.80\% \pm 2.05\%$	$6.55\% \pm 0.48\%$	400.76%
11	30	600	1	$57.15\% \pm 1.96\%$	$23.80\% \pm 1.22\%$	140.13%
12	30	600	0	$52.80\% \pm 2.52\%$	$14.55\% \pm 1.12\%$	262.89%
13	60	400	1	$34.70\% \pm 1.07\%$	$24.10\% \pm 1.21\%$	43.98%
14	60	400	0	$36.90\% \pm 1.47\%$	$13.50\% \pm 1.74\%$	173.33%
15	60	500	1	$54.05\% \pm 1.95\%$	$34.45\% \pm 1.00\%$	56.89%
16	60	500	0	$53.00\% \pm 2.52\%$	$23.95\% \pm 1.01\%$	121.29%
17	60	600	1	$76.60\% \pm 1.89\%$	$50.25\% \pm 1.13\%$	52.44%
18	60	600	0	$75.80\% \pm 1.71\%$	$43.25\% \pm 1.61\%$	75.26%

5.7 Focused Analysis of UCAV Speed

In this section, we consider the specific effect of UCAV speed on defensive performance in the ABM Problem. The sensitivity analysis in the previous section shows clear and significant relationships between policy performance and several ABM Problem environmental factors: cruise missile arrival rate, UCAV speed, and cruise missile arrival directional symmetry. The UCAV speed factor differs from the other two factors in that friendly forces are able to actively control UCAV speed through force management or acquisitions activities, whereas the cruise missile arrivals rates and directions are considered uncontrollable characteristics of the combat environment. The results reported in Table 14 show how varying speed affects mean success rate across all problem instances under both the API-LSTD and benchmark policies.

Table 14. Effect of UCAV Speed on Success Rate (J)

Flight Speed	API-LSTD Mean Success Rate (J)	Benchmark Mean Success Rate (J)
-20%	91.89%	85.09%
Baseline	94.87%	88.85%
+20%	97.01%	92.31%

Recall that UCAV flight speed is shown to be a statistically significant predictor of API-LSTD policy success rate in the multiple linear regression metamodel in Table 10. The observations in Table 14 offer several meaningful insights. First, as expected, friendly forces will observe noticeable improvements in defensive performance by adopting a faster and more maneuverable UCAV for intercepting hostile forces, which is an observation that can aid in informing force management and acquisitions decisions. Second, and perhaps more importantly, we observe that simply moving from the benchmark decision policy to the improved ADP policy that incorporates stochastic information in decision-making offers a similar mean benefit to increasing UCAV speed by a total of 50%. A decision-making policy of this type is likely implementable with minimal cost and offers significant improvements.

5.8 Focused Analysis of Intercept Proximity to the Defended Asset

This section analyzes the distance from the defended asset at which the UCAVs intercept missiles in all problem instances and under both policies. We define this dependent variable as the mean distance from the defended asset of all UCAV intercept actions. Although the previous analysis of mean success rate translates directly to the ability of a policy to protect forces and facilities, it does not consider the practical consideration of intercept distance. Considering only mean success rate is somewhat shortsighted, as equivalently performing policies in terms of success rate may vary drastically in desirability. That is, if one equivalently performing policy consistently

intercepts missiles far away from the FOB, while the other policy consistently intercepts missiles at the last available opportunity, there is increased risk that the latter policy may not be robust to increases in the missile arrival rate. Moreover, there is a psychological component to consider, wherein a greater perception of safety is likely to positively affect the well-being of forces stationed at the FOB. Recall that the contribution function given in Equation (4) utilizes a constant value for penalizing missile impacts; thus, the UCAVs are not offered incentives to intercept missiles farther away from the defended asset. Table 15 reports the results of this analysis.

Table 15. Intercept Proximity to the Defended Asset

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Mean Intercept Proximity (mi)	Benchmark Mean Intercept Proximity (mi)	API-LSTD Mean Difference
1	20	400	1	5.18 ± 0.92	7.32 ± 1.05	-29.26%
2	20	400	0	2.14 ± 0.21	4.56 ± 0.70	-53.16%
3	20	500	1	4.35 ± 0.52	7.60 ± 0.84	-42.78%
4	20	500	0	2.46 ± 0.24	5.17 ± 0.59	-52.39%
5	20	600	1	5.28 ± 0.28	7.85 ± 0.70	-32.74%
6	20	600	0	3.54 ± 0.53	5.06 ± 0.35	-30.09%
7	30	400	1	4.82 ± 0.80	7.02 ± 0.56	-31.39%
8	30	400	0	3.03 ± 0.50	4.45 ± 0.28	-31.88%
9	30	500	1	4.89 ± 0.80	7.36 ± 0.61	-33.56%
10	30	500	0	3.47 ± 0.70	5.17 ± 0.54	-33.00%
11	30	600	1	6.17 ± 0.79	7.75 ± 0.65	-20.30%
12	30	600	0	4.46 ± 0.57	5.77 ± 0.69	-22.67%
13	60	400	1	4.88 ± 1.29	6.47 ± 0.75	-24.65%
14	60	400	0	3.39 ± 0.54	4.74 ± 0.37	-28.39%
15	60	500	1	4.99 ± 1.11	6.56 ± 0.78	-23.98%
16	60	500	0	3.86 ± 0.54	5.43 ± 0.60	-28.93%
17	60	600	1	6.43 ± 1.05	7.11 ± 0.54	-9.56%
18	60	600	0	5.28 ± 0.96	6.09 ± 0.67	-13.39%

We observe that the mean intercept distance of the API-LSTD-generated policy is lower than the mean intercept distance of the benchmark policy in every instance, with some instances being statistically significant and others not. The UCAVs under the API-LSTD policy tend to stay closer to the defended asset overall, decreasing the size of the actively defended region considerably. Although this tendency is partially

responsible for the API-LSTD policy's improved success rates, the measure of intercept distance is important to consider. Over all problem instances, the API-LSTD policy achieves a mean intercept distance of 4.4 miles, whereas the benchmark policy achieves a mean intercept distance of 6.2 miles. Neither policy elicits concerns due to consistent close-call intercepts.

5.9 Focused Analysis of UCAV Idle Time

In this section, we examine UCAV idle time under each policy and every problem instance. We define this dependent variable as the percentage of time that a UCAV is not actively assigned to intercept a missile. Although idle time understandably increases as missile arrival rate decreases, the aspect of idle time has several implications. First, given a specific problem instance, it may be possible to find a policy that increases success rate by decreasing idle time, indicating a more effective use of resources. Second, we assume that UCAVs consume less fuel while not actively engaging a target; thus increased idle time may indicate some amount of fuel savings. Table 16 reports the results of this analysis.

In all problem instances, the API-LSTD-generated policy decreases UCAV idle time by a statistically significant amount while also improving success rate. This result indicates that the policy is making more effective use of resources but at the cost of increased fuel consumption.

Table 16. UCAV Idle Time Comparison

Run	$\frac{1}{\lambda}$	v (mph)	ψ	API-LSTD Idle Time	Benchmark Idle Time	API-LSTD Mean Difference
1	20	400	1	$4.87\% \pm 0.11\%$	$10.21\% \pm 0.18\%$	-52.29%
2	20	400	0	$5.28\% \pm 0.14\%$	$10.90\% \pm 0.22\%$	-51.52%
3	20	500	1	$5.46\% \pm 0.14\%$	$11.87\% \pm 0.18\%$	-53.98%
4	20	500	0	$5.47\% \pm 0.16\%$	$11.76\% \pm 0.26\%$	-53.47%
5	20	600	1	$6.32\% \pm 0.12\%$	$13.91\% \pm 0.18\%$	-54.55%
6	20	600	0	$5.73\% \pm 0.14\%$	$12.76\% \pm 0.23\%$	-55.13%
7	30	400	1	$10.48\% \pm 0.17\%$	$20.35\% \pm 0.32\%$	-48.47%
8	30	400	0	$11.34\% \pm 0.25\%$	$20.23\% \pm 0.36\%$	-43.97%
9	30	500	1	$11.96\% \pm 0.15\%$	$23.16\% \pm 0.24\%$	-48.36%
10	30	500	0	$11.89\% \pm 0.28\%$	$22.23\% \pm 0.38\%$	-46.51%
11	30	600	1	$14.61\% \pm 0.15\%$	$26.23\% \pm 0.16\%$	-44.31%
12	30	600	0	$13.23\% \pm 0.26\%$	$24.19\% \pm 0.46\%$	-45.33%
13	60	400	1	$17.22\% \pm 0.29\%$	$29.86\% \pm 0.38\%$	-42.34%
14	60	400	0	$18.30\% \pm 0.32\%$	$29.34\% \pm 0.39\%$	-37.65%
15	60	500	1	$19.29\% \pm 0.28\%$	$32.97\% \pm 0.36\%$	-41.49%
16	60	500	0	$19.43\% \pm 0.28\%$	$31.70\% \pm 0.32\%$	-38.72%
17	60	600	1	$22.89\% \pm 0.27\%$	$36.08\% \pm 0.24\%$	-36.54%
18	60	600	0	$21.70\% \pm 0.30\%$	$34.20\% \pm 0.40\%$	-36.57%

VI. Conclusion

This research examines the air battle management (ABM) problem wherein a set of friendly unmanned combat aerial vehicles (UCAV) is tasked to defend a central asset from incoming cruise missiles. The effectiveness of the UCAVs is measured by their ability over time to maintain air superiority by successfully targeting and intercepting these cruise missiles. The intent of this research is to develop a representative combat scenario and determine high-quality policies for UCAV tasking that maximize their ability to defend the central asset. We develop a Markov decision process (MDP) model to explain each component of the ABM problem with the understanding that the continuous state space of the problem renders a traditional dynamic programming solution to the MDP computationally intractable.

To accurately characterize how this dynamic system evolves over time, we develop and implement a modular simulation system. We utilize an approximate dynamic programming (ADP) technique known as approximate policy iteration with least squares temporal differences (API-LSTD) to find high-quality solutions to the problem. The architecture of the API-LSTD algorithm is defined in part by several tunable hyperparameters. These hyperparameters differ from other system parameters in that the ideal settings cannot be determined directly from the data and must be discovered by experimentation. We design and conduct a sequential computational experiment, consisting of an initial experiment to investigate a wide range of hyperparameter settings followed by an optimization experiment to investigate a more specific range of settings identified by the initial experiment. We create a multiple linear regression metamodel based on the results of these experiments to identify the superlative hyperparameter settings which are then used to formally compare the performance of a reasonable benchmark policy against the performance of the ADP policy. Finally, we design and conduct a series of sensitivity analysis experiments to determine how mod-

ifications to several problem features, such as cruise missile arrival rate, affect solution quality. These experiments require a total of approximately 75 hours of computation time on a processing system using an Intel Core i7-9700k with 8 cores at 5.2GHz and 32GB RAM. We implement the API-LSTD algorithm in MATLAB R2020b and use MATLAB's Parallel Computing Toolbox alongside built-in functionality for solving large systems of linear equations via matrix inverse operations.

In the baseline scenario, the ADP policy improves mean success rate by 6.8% compared to the benchmark policy at the cost of a 46.5% decrease in UCAV idle time, indicating a trade-off between success rate and fuel cost. More specifically, the ADP policy offers a 318% increase over the benchmark in frequency of optimal-equivalent performance. That is, the UCAVs perform equivalently to an optimal policy by intercepting 100% of the incoming cruise missiles before they impact the defended asset.

The improved performance of the ADP policy also comes with the cost of intercepting cruise missiles on average 1.7 miles closer to the defended asset when compared to the benchmark policy. Although the ADP policy mean intercept distance is a non-concerning 4.4 miles, a potential extension to research would determine whether there is subjective value in intercepting missiles farther away, and it may implement model changes such as a reward for intercepting missiles that decays with time or proximity to the defended asset.

Two major limiting assumptions of this work are that the battle time horizon is short enough such that fuel is not a concern, and the UCAVs are able to intercept any number of incoming missiles without needing to rearm. An extension to improve model realism would be the addition of UCAV fuel capacity and weapons load, wherein a high-quality policy would need to determine the best times to remove a UCAV from intercept assignments temporarily to refuel and rearm. Results of this re-

search could better inform policies representative of long-term, steady-state defensive counterair (DCA) operations. Additional worthwhile extensions to this work include a comparison of the API-LSTD-generated policy with policies generated using other ADP techniques, such as those using a neural network for learning value function approximations.

Overall, we find that the implementation of an ADP policy offers significant increases to DCA operation success rates with the identified increase to fuel costs. However, we find that the success rate increase of the ADP policy is, on average, equivalent to the success rate increase of the benchmark policy with a 50% faster UCAV. Understanding that slightly increased fuel costs pale in comparison to the acquisition cost of a faster UCAV, we conclude that implementing a ADP-generated policy for target assignment tasking in the ABM problem is a cost-effective means to improve protection of friendly forces and facilities.

Bibliography

Bellman, R. (1957), 'A Markovian Decision Process', *Journal of Mathematics and Mechanics* pp. 679–684.

Buckley, J. D. and Buckley, J. J. (1999), *Air Power in the Age of Total War*, Indiana University Press, Indianapolis, IN.

Chao, I.-M., Golden, B. L. and Wasil, E. A. (1996), 'The Team Orienteering Problem', *European Journal of Operational Research* **88**(3), 464–474.

Dantzig, G. B. and Ramser, J. H. (1959), 'The Truck Dispatching Problem', *Management Science* **6**(1), 80–91.

Dantzig, G., Fulkerson, R. and Johnson, S. (1954), 'Solution of a Large-Scale Traveling Salesman Problem', *Journal of the Operations Research Society of America* **2**(4), 393–410.

Davis, M. T., Robbins, M. J. and Lunday, B. J. (2017), 'Approximate Dynamic Programming for Missile Defense Interceptor Fire Control', *European Journal of Operational Research* **259**(3), 873–886.

Department of Defense (2014), *Unmanned Aircraft Systems: DoD Purpose and Operational Use*, Washington, DC.

Department of Defense, U. S. (2012), *Department of Defense Directive 3000.09, Autonomy in Weapon Systems*, Washington, DC.

Department of Defense, U. S. (2018*a*), *Joint Publication 3-0, Joint Operations*, Washington, DC.

Department of Defense, U. S. (2018*b*), *Joint Publication 3-01, Countering Air and Missile Threats*, Washington, DC.

Department of Defense, U. S. (2018*c*), *The National Defense Strategy of the United States*, Washington, DC.

Department of the Air Force (2011), *Air Force Doctrine Document 3-01, Counterair Operations*, Washington, DC.

Department of the Army (2019), *U.S. Army Field Manual 4-02.2*, Washington, DC.

Federal Aviation Administration (2020), 'UAS by the Numbers', `https://www.faa.gov/uas/resources/by_the_numbers/`.

Franke, U. E. (2014), 'The Global Diffusion of Unmanned Aerial Vehicles (UAVs) or Drones', *Precision Strike Warfare and International Intervention: Strategic, Ethico-Legal and Decisional Implications* pp. 27–109.

Golden, B. L., Levy, L. and Vohra, R. (1987), 'The Orienteering Problem', *Naval Research Logistics (NRL)* **34**(3), 307–318.

Jenkins, P. R., Robbins, M. J. and Lunday, B. J. (2021), 'Approximate Dynamic Programming for Military Medical Evacuation Dispatching Policies', *INFORMS Journal on Computing* **33**, 2–26.

Joint Targeting School (2017), *Joint Targeting Student Guide*, Dam Neck, VA.

Kantor, M. G. and Rosenwein, M. B. (1992), 'The Orienteering Problem with Time Windows', *Journal of the Operational Research Society* **43**(6), 629–635.

Karaboga, D. and Basturk, B. (2007), 'A Powerful and Efficient Algorithm for Numerical Function Optimization: Artificial Bee Colony (ABC) Algorithm', *Journal of Global Optimization* **39**(3), 459–471.

Kuhn, H. W. (1955), 'The Hungarian Method for the Assignment Problem', *Naval Research Logistics Quarterly* **2**(1-2), 83–97.

Kulkarni, V. G. (2017), *Modeling and Analysis of Stochastic Systems*, 3rd edn, CRC Press, Boca Raton, FL.

McKenna, R. S., Robbins, M. J., Lunday, B. J. and McCormack, I. M. (2020), 'Approximate Dynamic Programming for the Military Inventory Routing Problem', *Annals of Operations Research* **288**(1), 391–416.

Pentico, D. W. (2007), 'Assignment Problems: A Golden Anniversary Survey', *European Journal of Operational Research* **176**(2), 774–793.

Puterman, M. L. (2005), *Markov Decision Processes: Discrete Stochastic Dynamic Programming*, John Wiley & Sons, Hoboken, New Jersey.

Rettke, A. J., Robbins, M. J. and Lunday, B. J. (2016), 'Approximate Dynamic Programming for the Dispatch of Military Medical Evacuation Assets', *European Journal of Operational Research* **254**, 824–839.

Summers, D. S., Robbins, M. J. and Lunday, B. J. (2020), 'An Approximate Dynamic Programming Approach for Comparing Firing Policies in a Networked Air Defense Environment', *Computers and Operations Research* **117**, 1–15.

Sutton, R. S. and Barto, A. G. (2018), *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, MA.

Tsiligirides, T. (1984), 'Heuristic Methods Applied to Orienteering', *Journal of the Operational Research Society* **35**(9), 797–809.

United Nations (2018), '2018 Revision of World Urbanization Prospects', `https://www.un.org/development/desa/publications/2018-revision-of-world-urbanization-prospects.html`.

Vincent, F. Y., Jewpanya, P., Lin, S.-W. and Redi, A. P. (2019), 'Team Orienteering Problem with Time Windows and Time-Dependent Scores', *Computers & Industrial Engineering* **127**, 213–224.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE *(DD-MM-YYYY)*	2. REPORT TYPE	3. DATES COVERED *(From - To)*
25-03-2021	Master's Thesis	Aug 2019 - Mar 2021

4. TITLE AND SUBTITLE
Improving Air Battle Management Target Assignment Processes via Approximate Dynamic Programming

5a. CONTRACT NUMBER

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER

6. AUTHOR(S)
Liles IV, Joseph M, Lt Col, USAF

5d. PROJECT NUMBER

5e. TASK NUMBER

5f. WORK UNIT NUMBER

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Air Force Institute of Technology
Graduate School of Engineering and Management (AFIT/EN)
2950 Hobson Way
WPAFB OH 45433-7765

8. PERFORMING ORGANIZATION REPORT NUMBER
AFIT-ENS-MS-21-M-173

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Mr. David M. Panson
Strategic Development Planning & Experimentation (SDPE) Office
1864 4th Street
Wright-Patterson AFB, OH 45433
(937) 904-6539

10. SPONSOR/MONITOR'S ACRONYM(S)
SPDE

11. SPONSOR/MONITOR'S REPORT NUMBER(S)

12. DISTRIBUTION/AVAILABILITY STATEMENT
DISTRIBUTION STATEMENT A: APPROVED FOR PUBLIC RELEASE; DISTRIBUTION UNLIMITED.

13. SUPPLEMENTARY NOTES
This work is declared a work of the U.S. Government and is not subject to copyright protection in the United States.

14. ABSTRACT
Military air battle managers face many challenges when directing operations in quickly evolving combat scenarios. These scenarios require rapid decisions to engage moving and unpredictable targets. In defensive operations, the success of a sequence of air battle management decisions is reflected by the friendly force's ability to maintain air superiority by defending friendly assets. We develop a Markov decision process (MDP) model of the air battle management (ABM) problem, wherein a set of unmanned combat aerial vehicles (UCAV) is tasked to defend a central asset from cruise missiles that arrive stochastically over time.

15. SUBJECT TERMS
Markov decision process, approximate dynamic programming, air battle management, policy iteration, 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
					19b. TELEPHONE NUMBER *(Include area code)*
U	U	U	UU	69	937-255-3636; matthew.robbins@afit.edu