The Eye and the Fist: Optimizing Search and Interdiction

M. Kress, J. O. Royset, and N. Rozen
Operations Research Department, Naval Postgraduate School, Monterey, California

Abstract. Interdiction operations involving search, identification, and interception of suspected objects are of great interest and high operational importance to military and naval forces as well as nation's coast guards and border patrols. The interdiction scenario discussed in this paper includes an area of interest with multiple neutral and hostile objects moving through this area, and an interdiction force, consisting of an airborne sensor and an intercepting surface vessel or ground vehicle, whose objectives are to search, identify, track, and intercept hostile objects within a given time frame. The main contributions of this paper are addressing both airborne sensor and surface vessel simultaneously, developing a stochastic dynamic-programming model for optimizing their employment, and deriving operational insight. In addition, the search and identification process of the airborne sensor addresses both physical (appearance) and behavioral (movement pattern) signatures of a potentially hostile object. As the model is computationally intractable for real-world scenarios, we propose a simple heuristic policy, which is shown, using a bounding technique, to be quite effective. Based on a numerical case study of maritime interdiction operations, which includes several representative scenarios, we show that the expected number of intercepted hostile objects, following the heuristic decision policy, is at least 60% of the number of hostile objects intercepted following an optimal decision policy.

1. Introduction

Interdiction operations involving search, identification, and interception of suspected objects are of great interest and high operational importance to military and naval forces as well as nation's coast guards and border patrols [1]. There are two key assets in interdiction operations that we consider in this paper: an airborne sensor – an "eye" – such as surveillance (fixed-wing) aircraft, patrol helicopter, or unmanned aerial vehicle (UAV), whose mission is to search, detect, track, and identify potential targets, and a surface vessel or ground vehicle – a "fist" – which is dispatched following a cue from the sensor to investigate and potentially apprehended a suspicious object. This study is motivated by current operational needs in maritime counter-terrorism, counter-drug, and counter-piracy missions. In such targeted and focused missions only a single airborne

asset and a single surface vessel may operate in a certain part of a region of interest [2]. In this paper, we develop a stochastic dynamic-programming (DP) model for optimizing the combined operation of these two assets. In principle, the model is solvable by the Backward DP Algorithm (see for example [3], p. 50), but in real-world scenarios that approach may not be computationally feasible due to the model size. Consequently, we develop a greedy heuristic algorithm that can be used in real-time to effectively deploy and employ the two assets. We verify the quality of the heuristic by constructing a relaxation of the model and showing that for some realistic scenarios the heuristic generates solutions that are at most 40% from optimality.

The field of classical search theory, addressing the problem of optimal search for static or mobile targets, has been extensively studied for over seven decades, since the groundbreaking research of Koopman [4], through the seminal works of Washburn [5] and Stone [6], to the recent surge in publications; for example see [7-19]. The problem of coordinating search and interception—the topic of this paper—is more involved. Wein and Atkinson [20] study a radiation detection system, combined with interception efforts, for protecting an urban area from nuclear terrorist attack. Jeffcoat et al. [21] deal with searching and engaging multiple targets where each search or engagement asset can engage at most one target. Barton et al. [22] consider a team of UAVs comprising two groups: searchers that use dynamic co-fields to avoid obstacles, and disposable UAVs that are called in, when targets are found, to kill the targets; see also [23] for a related study. The balance between search for unknown targets and interception of known targets represents a classical exploration versus exploitation trade-off [24], which is known to be difficult to carry out optimally. We refer to [25] for a recent study of algorithms and complexity results and [26] for heuristics. A related study is also [27], which deals with the placement of stationary perimeter cameras while accounting for interceptions by an unmanned helicopter following detections by the cameras. We refer to [28] for a study of object identification without the need for search and interception.

In contrast to many of the above studies, which mostly focus on technical and command-and-control aspects of employing a large number of search and interception assets, we take an operational approach, which reflects typical current situations in maritime

missions, where interdiction assets are scarce [2]. We account for possible identification errors, consider both the physical signature of a suspicious object and its movement pattern, and optimize routing and scheduling decisions taken by a task-force commander. The measure of performance is the expected number of targets successfully interdicted. The main contribution of this paper is threefold: We model the combined effect of the "eye and the fist," incorporate information about physical signature and movement pattern of suspicious objects, and derive operational insight about when to trigger investigation by the surface vessel. In an earlier study [29] we deal with a similar situation. However, that study does not consider tracking of suspicious objects, information about movement patterns of objects, and lacks the analytical rigor and the solution-quality bounds for the proposed heuristic algorithm presented in the current paper. Our modeling approach is similar to that found in the extensive literature on stochastic and dynamic task allocation and vehicle routing (e.g., [3,30] and references therein), but is specialized to the unique features of interdiction operations.

The next section defines the operational scenario. Section 3 presents the stochastic DP model. Section 4 describes a heuristic algorithm for solving the model and an associated model that is used to construct a bound on the optimal value of the original model. Section 5 presents a case study for maritime interdiction missions.

2. Scenario

We consider an area of interest (*AOI*) that contains multiple mobile *objects.* Some of the objects are hostile, called *targets,* and the remaining are *neutrals*. The objective of the interdiction force is to intercept as many targets as possible within a finite time horizon discretized into time periods. The number of objects, which enter, move about, and (eventually) exit the AOI is unknown. The AOI is subdivided into a finite number of area cells (*ACs*). The objects are oblivious to the presence of the interdiction force and therefore they do not act strategically; they move independently of each other according to a known Markov chain defined on the set of ACs. The movements of targets and neutrals may follow different Markov chains. An object enters and departs the AOI according to a Bernoulli process. We assume stationarity in the sense that neither the entry probabilities nor the in-AOI transition or exit probabilities depend on the time

period. Motivated by our discretization of space and time, with resolution that can be arbitrarily high, and assuming that the AOI is relatively large compared to the (unknown) number of objects, we neglect the possibility of more than one object in any specific AC at any given time period. This is a reasonable approximation to the situation in open-sea scenarios and it simplifies the model. A similar assumption is made in [31]. The interdiction force comprises two assets: an airborne sensor, called a *Recognizer*, whose mission is to search, detect, track, and identify targets, and a ground vehicle or surface vessel, called an *Interceptor*, capable of intercepting and apprehending a target.

We assume that the Recognizer has perfect detection capabilities, i.e., it can determine with certainty whether the AC, in which it is currently located, contains an object. This is a reasonable assumption as radars usually detect objects such as fishing vessels and go-fast boats at a substantial range. The Recognizer examines one AC at a time until it detects an object. Following detection, the Recognizer tracks the object for one time period and then determines the nature of the object using a threshold policy described in Section 3. The Recognizer is subject to both false positive and false negative errors when identifying an object. The modeling of the tracking process is based on a series of "looks", as described in Section 3.2. For more details on tracking see [32,33]. If the object is identified as a neutral, the Recognizer proceeds with its search. Otherwise, the Recognizer flags the suspected target and calls in the Interceptor. We do not describe in detail the "pursuer-evader game" (see for example [34-36]) that may take place after an object is flagged and make the simplifying assumption that once flagged, the object remains stationary at its location until the arrival of the Interceptor. This assumption simplifies considerably the model while affecting only marginally the operational reality because of the different time scales of airborne and surface vehicles. The Recognizer remains with the object until the Interceptor arrives and completes the interception, at which time the Recognizer returns to its search. Any object that is tracked by the Recognizer is tagged (e.g., electronically) as "examined" and is of no further interest.

The Interceptor has perfect identification capability; it can distinguish with certainty between a target and a neutral. When not involved in an intercepting mission, the Interceptor moves according to a given deterministic policy. For simplicity of exposition,

we throughout the paper assume that the policy is to remain stationary. Thus, the Interceptor is stationary at the location of its last interception (or initial deployment absent interceptions), waiting for calls by the Recognizer. However, other policies can trivially be incorporated in the model. The goal of the interdiction force is to maximize the expected total time-discounted number of intercepted targets during the time horizon.

3. Model Development

The dynamic program in this paper is constructed based on conventions presented in [3], pp. 129–178. We first present the main components of the model and then discuss its details.

3.1 The Main Components of the Model

Let $\mathcal{A}$ and $\mathcal{A}_0$ denote the set of ACs in the AOI and the area outside the AOI, respectively. Let $t = 1, 2, ..., T$ denote the (discrete) time index. While we could have formulated the dynamic program in the classical manner with a possible decision at each time period, we choose to adopt a somewhat unconventional approach that is event- rather than time-driven. The reason is that the situation we consider involves substantial blocks of time periods during which no decisions are required. Specifically, while the Recognizer is travelling to an AC, or tracking an object, or waiting for the Interceptor, no decisions are expected to be made. We utilize this special situation and develop an event-driven formulation where decisions are only made at random time periods when certain events occur. This construct is described in detail below. Our approach results in a state space of smaller cardinality, which we utilize computationally in Sections 4 and 5. Thus, we define a *state* as a vector $s = (t, r, i, \pi, \theta)$, where $r \in \mathcal{A}$ and $i \in \mathcal{A}$ are the Recognizer's and Interceptor's locations at time *t*, respectively, and π and θ are vectors of probabilities with components π_a and θ_a, $a \in \mathcal{A}$, respectively. Here, π_a (respectively, θ_a) is the probability that a neutral (target) is present in AC a at time period *t*. Let $\mathcal{S} \subset \{1, 2, \ldots, T\} \times \mathcal{A} \times \mathcal{A} \times [0,1]^{|\mathcal{A}|} \times [0,1]^{|\mathcal{A}|}$ be the space of all possible state vectors. The inclusion of $\mathcal{S}$ in the right-hand side is strict because the probabilities π_a

and θ_a may only take on a finite number of values in a given problem instance due to the finite number of detection and interception opportunities within the finite time horizon. Hence, the state-space $\mathcal{S}$ is of finite, but extremely high, cardinality.

A decision $x \in \mathcal{A}$ determines the next AC to be visited by the Recognizer; this decision is made either at $t = 0$ or when the existing decision is *fathomed.* A decision $x \in \mathcal{A}$ is said to be fathomed in one of the following three situations: (i) no object is found by the Recognizer in AC x, (ii) an object is found in AC *x* but identified as a neutral, or (iii) an object is found in AC x, identified as a target, intercepted, and determined to be either a target or a neutral. As soon as a decision is fathomed, a new decision is made. Each new decision constitutes a *stage* in the detection-interception process.

Let $w = (\Delta t_w, r_w, i_w, z_w)$ denote the vector of random variables representing the information available when a decision is fathomed. The first component Δt_w is the duration of a stage (i.e., the time between when a decision is made and when it is fathomed), and the variables r_w and i_w denote the Recognizer's and Interceptor's locations at the end of a stage, respectively. The Bernoulli random variable z_w equals 1 if the stage ends with a target interception and 0 otherwise. Let $\mathcal{W}$ denote the space of possible realizations of w. The probability distribution of w, which depends on the state s and the decision x, is derived in Section 3.3. To simplify notation, we write w for both the random vector and its realization.

The next state is determined by the *state-transition function* $s^M : \mathcal{S} \times \mathcal{A} \times \mathcal{W} \to \mathcal{S}$, which depends on the current state, the decision, and the information obtained when the decision is fathomed; see Section 3.3 for details. The *reward* associated with state $s = (t, r, i, \pi, \theta)$ and the following realization $w = (\Delta t_w, r_w, i_w, z_w)$ is given by

$$c(w, s) = \begin{cases} z_w \cdot (1+\gamma)^{-(t+\Delta t_w)}, & t + \Delta t_w \le T \\ 0, & t + \Delta t_w > T. \end{cases} \tag{1}$$

The reward is 0 if no target is intercepted or if the time of interception is beyond the time horizon, and is a discounted value, with discount factor γ, otherwise. The discount factor

captures the property that the sooner a target is captures the better the operational effect of the interception. The Bellman equation for state $s=(t,r,i,\pi,\theta)$ takes the form

$$V(s)=\begin{cases}\max\limits_{x} E\left[c(w,s)+V\left(s^{M}(s,x,w)\right)\right], & t<T\\ 0, & t\geq T\end{cases} \tag{2}$$

where $V(s)$ is the value of being in state s, and the expectation is with respect to the probability distribution of w (see Section 3.3). The stochastic DP model in (2) is denoted by SDP, and the corresponding optimal policy is referred to as the SDP policy.

3.2 Probability Updates

Let $P(a',a)$ denote the single time-period transition probability from AC a' to AC a of a neutral, and let $P=\left[P(a',a)\right]$, $a',a\in\mathcal{A}\cup\mathcal{A}_0$, be the corresponding matrix. Similarly, we define $Q=\left[Q(a',a)\right]$ for a target. Let α_a and β_a denote the single time-period arrival probabilities of a neutral and a target, respectively, to AC a. Absent the interdiction force, let π_a^0 (respectively, θ_a^0) be the steady-state probability of a neutral (target) in AC $a\in\mathcal{A}$. Note that both neutrals and targets are assumed to adhere to a homogeneous regular Markov chain (no index t to P and Q) and therefore, despite the final horizon T considered in our model, the steady-state probabilities are well defined as follows:

$$\pi_a^0=1-(1-\alpha_a)\prod_{l\in\mathcal{A}}\prod_{k=1}^{\infty}\left(1-\alpha_l P^k(l,a)\right) \tag{3}$$

where $P^k(l,a)$ is the (l,a) entry of P^k, the transition matrix P raised to the k^{th} power. Similarly, for targets we obtain that

$$\theta_a^0=1-(1-\beta_a)\prod_{l\in\mathcal{A}}\prod_{k=1}^{\infty}\left(1-\beta_l Q^k(l,a)\right). \tag{4}$$

In the presence of an interdiction force, these probabilities may be updated as described in Section 3.3. Let π_a^t and θ_a^t denote the updated probabilities of a neutral and a target in AC a at time t, respectively. Given π_a^t and θ_a^t $a\in\mathcal{A}$, and no updates during $(t,t']$, $t'>t$,

$$\pi_{a'}^{t'} = 1-\left(1-\alpha_{a'}\right)\left(\prod_{a\in\mathcal{A}}\left(1-\pi_a^t P^{t'-t}\left(a,a'\right)\right)\right)\left(\prod_{a\in\mathcal{A}}\prod_{k=1}^{t-t'-1}\left(1-\alpha_a P^k\left(a,a'\right)\right)\right) \tag{5}$$

where the second product in (5) is equal to 1 if $t'-t=1$. Similarly, for a target,

$$\theta_{a'}^{t'} = 1-\left(1-\beta_{a'}\right)\left(\prod_{a\in\mathcal{A}}\left(1-\theta_a^t Q^{t'-t}\left(a,a'\right)\right)\right)\left(\prod_{a\in\mathcal{A}}\prod_{k=1}^{t-t'-1}\left(1-\beta_a Q^k\left(a,a'\right)\right)\right) \tag{6}$$

Suppose that the Recognizer visits AC *a* at time *t* and let $\pi_a^{t,Det}$ and $\theta_a^{t,Det}$ denote the updated probabilities following that visit. If the AC is void of objects then $\pi_a^{t,Det}=\theta_a^{t,Det}=0$. Otherwise,

$$\pi_a^{t,Det} = \frac{\pi_a^t}{\pi_a^t+\theta_a^t} \tag{7}$$

$$\theta_a^{t,Det} = \frac{\theta_a^t}{\pi_a^t+\theta_a^t} = 1-\pi_a^{t,Det}. \tag{8}$$

Following a detection of an object, the Recognizer tracks the object for one time period and utilizes two modes of recognition: *signature recognition* (e.g., using an electro-optical sensor) and *movement recognition*, in which the Recognizer tries to identify the movement pattern of the tracked object (i.e., leaving known shipping lanes or any other suspicious movement). The movement recognition relates to the extensive literature on anomaly detection; see, e.g., [37-39]. Without loss of generality, we assume that signature recognition takes place first and the Recognizer takes g looks (glimpses) at the tracked object. The glimpses are conditionally independent given the presence of the object in that AC. Let $1-u$ and $1-v$ denote the single glimpse false negative probability of identifying a target as a neutral, and the false positive probability of identifying a neutral as a target, respectively. Suppose that n glimpses result in "neutral" cues, $g-n$ glimpses result in "target" cues, and the object moves from AC *a* to AC *j* (if the objects leaves the AOI, the decision is fathomed). Let $\pi_j^{t+1,Sig}$ denote the signature-posterior probability of a neutral following the *g* glimpses, where

$$\pi_j^{t+1,Sig} = \frac{\binom{g}{n}v^n\left(1-v\right)^{g-n}\pi_x^{t,Det}}{\binom{g}{n}v^n\left(1-v\right)^{g-n}\pi_x^{t,Det}+\binom{g}{n}\left(1-u\right)^n\left(u\right)^{g-n}\theta_x^{t,Det}} \tag{9}$$

and, similarly, the signature-posterior probability of a target is

$$\theta_j^{t+1,Sig} = \frac{\binom{g}{n}(1-u)^n (u)^{g-n} \theta_x^{t,Det}}{\binom{g}{n} v^n (1-v)^{g-n} \pi_x^{t,Det} + \binom{g}{n}(1-u)^n (u)^{g-n} \theta_x^{t,Det}} = 1 - \pi_j^{t+1,Sig}. \quad (10)$$

Finally, observing that the object has moved from AC a to AC j, the movement recognition mode takes the posteriors of the signature recognition mode as priors and

$$\pi_j^{t+1,Rec} = \frac{P(a,j)\pi_j^{t+1,Sig}}{P(a,j)\pi_j^{t+1,Sig} + Q(a,j)\theta_j^{t+1,Sig}}. \quad (11)$$

for a neutral, and

$$\theta_j^{t+1,Rec} = \frac{Q(a,j)\theta_j^{t+1,Sig}}{P(a,j)\pi_j^{t+1,Sig} + Q(a,j)\theta_j^{t+1,Sig}} = 1 - \pi_j^{t+1,Rec} \quad (12)$$

for a target. If (12) exceeds a predetermined threshold M, then the object is considered to be a suspected target and the interceptor is called in.

3.3 State Transitions

Given the state $s = (t, r, i, \pi, \theta)$ at the beginning of a decision stage, the decision *x*, and the realization of the information vector $w = (\Delta t_w, r_w, i_w, z_w)$, the state-transition function is $s^M(s,x,w) = (t + \Delta t_w, r_w, i_w, \pi^M, \theta^M)$, where π^M and θ^M are the probability vectors π and θ of the next state, just prior to making the next decision. The superscript *M* (*M* for *Markov*) indicates the dependence of the state transition on the underlying Markov process that governs the movements of both neutrals and targets. There are three time intervals (cases) we potentially need to account for when computing π^M and θ^M. First, the time between making the decision x and the Recognizer's arrival at x, second, the tracking and identification time of the detected object (a single time period), and third, the waiting time for the Interceptor to arrive and complete the interception. Figure 1 summarizes the three different cases that may occur, where $T_{r,x}^R$ is the time required by the Recognizer to move from AC *r* to AC *x*.

In Case 1, there is no object in AC *x* and therefore $\pi_x^M = 0$ and, for $a \neq x$, π_a^M is given by (5) with t' replaced by $t + T_{r,x}^R$. Similarly, $\theta_x^M = 0$ and, for $a \neq x$, θ_a^M is given by (6)

with t' replaced by $t+T^R_{r,x}$. In Case 2, we first compute π^M_a and θ^M_a as described for Case 1 and denote these values by $\pi^{M,temp}_a$ and $\theta^{M,temp}_a$. Then, we update these values to account for the single time period tracking and set $\pi^M_a = 0$ if a is the AC into which the tracked object's has transited at time t+1. Otherwise, we set π^M_a as given by (5) with t' and π^t_a replaced by $t+1$ and $\pi^{M,temp}_a$, respectively. Similar computation applies to θ^M_a.

Case 3 is computed by applying the computations of Case 2 repeatedly, until the Interceptor arrives at the AC of the object, the interception is completed, and the stage is over (i.e., decision is fathomed).

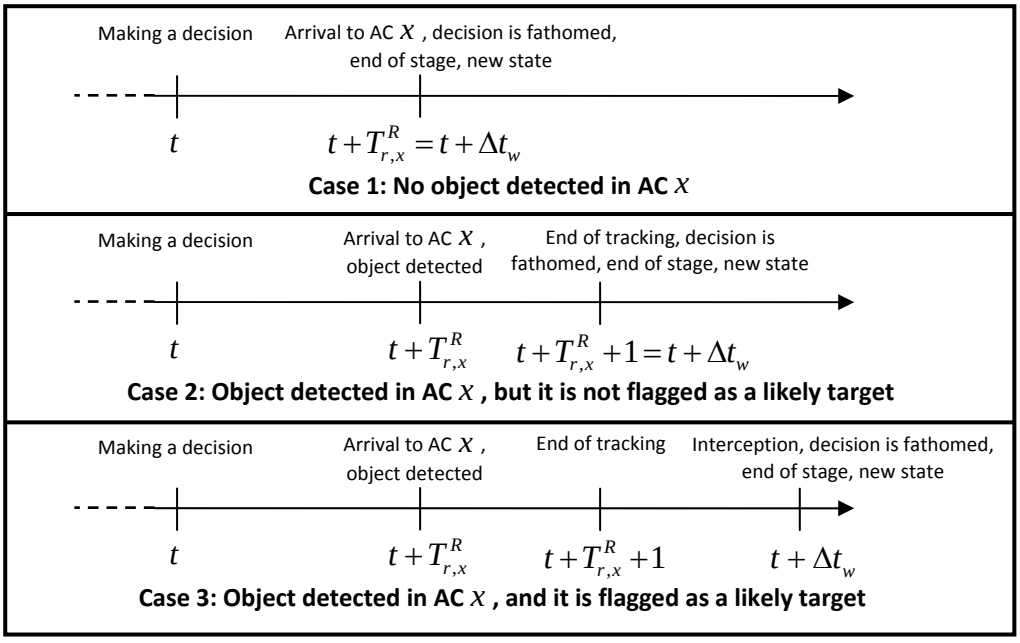

Figure 1. Timeline of state transitions

3.4 Probability Distribution of the Information Vector *w*

The final piece in formulating SDP is the probability mass function of the information vector $w=(\Delta t_w, r_w, i_w, z_w)$. Recall that w describes the consequences of a decision to visit a certain AC x: the time until the decision is fathomed, the locations of the Recognizer and Interceptor when this happens, and whether a target has been intercepted. Since our setting is discrete so is also *w*. Let $T^I_{i,j}$ denote the time it takes the Interceptor to

travel from AC i to AC j and to complete the processing of a suspected target in j. We assume that this time is fixed and given.

We consider five different and exhaustive events that may occur given state $s=(t,r,i,\pi,\theta)$ and decision x:

(i) no object is detected in AC x, which results in $w=(T^R_{r,x},x,i,0)$;

(ii) an object is present in AC x and it exits the AOI while being tracked, i.e., $w=(T^R_{r,x}+1,x,i,0)$;

(iii) an object is present in AC x, it moves to AC $j\in\mathcal{A}$ and is identified by the Recognizer as a neutral, i.e., $w=(T^R_{r,x}+1,j,i,0)$;

(iv) an object is present in AC x, it moves to AC $j\in\mathcal{A}$, is identified by the Recognizer as a target, and when intercepted is confirmed as a neutral, i.e., $w=(T^R_{r,x}+1+T^I_{i,j},j,j,0)$;

(v) as event (iv) but when intercepted the object is confirmed as a target, i.e., $w=(T^R_{r,x}+1+T^I_{i,j},j,j,1)$.

Next we introduce notation that describes the various possible events following a detection of an object in a certain AC x. The random variable d represents the outcome of a search in AC x, i.e., $d = -1$ if there is no object in AC x, $d = 0$ if there is an object in AC x and while being tracked it exits $\mathcal{A}$, and $d = j$ if there is an object in AC x and while being tracked it moves to AC j. The random variable f represents the result of tracking: $f = 0$ if there is an object in AC x and, following tracking, it is not identified as a target, and $f = 1$ if there is an object in AC x that is identified as a target. Note that $f = 0$ can either imply that the tracked object is identified by the Recognizer as a neutral, or that the object has left the AOI. Let $\hat{\pi}_x$ and $\hat{\theta}_x$ denote the probabilities given by (5) and (6), respectively, when t' is replaced by $t+T^R_{r,x}$. We consider the five events in turn.

Event (i) is equivalent to $\{d=-1\}$ and, hence, $\Pr\{\text{(i)}\}=\Pr\{d=-1\}=1-\hat{\pi}_x-\hat{\theta}_x$. Event (ii) is equivalent to $\{d=0\}$ and, hence, $\Pr\{\text{(ii)}\}=\Pr\{d=0\}=P(x,\mathcal{A}_0)\hat{\pi}_x+Q(x,\mathcal{A}_0)\hat{\theta}_x$. To compute the probabilities of the other three events, let θ^{Rec} denote the probability that, following tracking, the Recognizer identifies the object as a target; see (12). Recall that a tracked object is identified as target if $\theta^{Rec}\geq M$, where M is a given probability threshold. With a slight abuse of notation, let $\{x=\text{target}\}$ and $\{x=\text{neutral}\}$ denote the events that AC x contains a target and a neutral, respectively, at the time when the Recognizer arrives at AC x. We defer the calculation of event (iii) and next compute the probability of event (iv).

For any $j\in\mathcal{A}$,

$$\begin{aligned}
&\Pr\{\text{(iv)}\}=\Pr\{f=1,d=j,x=\text{neutral}\}\\
&=\Pr\{d=j,x=\text{neutral}\}\Pr\{f=1\mid d=j,x=\text{neutral}\}\\
&=\Pr\{d=j,x=\text{neutral}\}\Pr\{\theta^{Rec}\geq M\mid d=j,x=\text{neutral}\}\\
&=\Pr\{d=j,x=\text{neutral}\}\sum_{n'=0}^{g}\Pr\{\theta^{Rec}\geq M\mid d=j,x=\text{neutral},n=n'\}\Pr\{n=n'\mid d=j,x=\text{neutral}\}
\end{aligned}\tag{13}$$

where g is the given total number of glimpses the Recognizer takes while tracking the object, and $n\leq g$ is the number of glimpses that returned "neutral" cues. Note that for every $j\in\mathcal{A}$, we can calculate the maximum value of n for which $\theta^{\text{Rec}}\geq M$. Let n_j^* denote this value. Thus,

$$\Pr\{\theta^{Rec}\geq M\mid d=j,n=n'\}=\begin{cases}1, & \text{if } n'\leq n_j^*\\ 0, & \text{if } n'>n_j^*\end{cases}\tag{14}$$

Hence, in view of (13),

$$\begin{aligned}
&\Pr\{d=j,x=\text{neutral}\}\sum_{n'=0}^{g}\Pr\{\theta^{Rec}\geq M\mid d=j,x=\text{neutral},n=n'\}\Pr\{n=n'\mid d=j,x=\text{neutral}\}\\
&=\Pr\{d=j,x=\text{neutral}\}\sum_{n'=0}^{n_j^*}\Pr\{n=n'\mid d=j,x=\text{neutral}\}
\end{aligned}\tag{15}$$

Using Bayes' rule for the first multiplicative term on the right-hand-side of (15),

$$\begin{aligned}\Pr\{(\text{iv})\} &= \Pr\{d=j, x=\text{neutral}\}\sum_{n'=0}^{n_j^*}\Pr\{n=n' \mid d=j, x=\text{neutral}\}\\ &= \Pr\{d=j \mid x=\text{neutral}\}\Pr\{x=\text{neutral}\}\sum_{n'=0}^{n_j^*}\Pr\{n=n' \mid d=j, x=\text{neutral}\}\\ &= P(x,j)\hat{\pi}_x\sum_{n'=0}^{n_j^*}\left(\binom{g}{n'}v^{n'}(1-v)^{g-n'}\right).\end{aligned} \tag{16}$$

Following a similar derivation, we obtain for event (v) that

$$\Pr\{(\text{v})\} = \Pr\{f=1, d=j, x=\text{target}\} = Q(x,j)\hat{\theta}_x\sum_{n'=0}^{n_j^*}\left(\binom{g}{n'}(1-u)^{n'}(u)^{g-n'}\right). \tag{17}$$

Finally, for event (iii) we follow the derivation in events (iv) and (v) and obtain that for $j \in \mathcal{A}$,

$$\begin{aligned}&\Pr\{(\text{iii})\} = \Pr\{f=0, d=j\} =\\ &= P(x,j)\hat{\pi}_x\sum_{n'=n_j^*+1}^{g}\left(\binom{g}{n'}v^{n'}(1-v)^{g-n'}\right)+Q(x,j)\hat{\theta}_x\sum_{n'=n_j^*+1}^{g}\left(\binom{g}{n'}(1-u)^{n'}u^{g-n'}\right).\end{aligned} \tag{18}$$

3.5 Computation of Bellman's Equation

Given a state $s=(t,r,i,\pi,\theta)$ and information $w=(\Delta t_w, r_w, i_w, z_w)$, we see from (1) that $c(w,s)$ is only a function of t, Δt_w, and z_w. Hence, for the computation of $E[c(w,s)]=\sum_{w'\in\mathcal{W}}c(w',s)\Pr\{w=w'\}$ we only need the joint probability distribution of Δt_w, and z_w. Similarly, $s^M(s,w)$ is only a function of Δt_w, r_w, and i_w. Hence, we only need the joint probability distribution of these three random variables for the calculation of $E\left[V\left(s^M(s,x,w)\right)\right]=\sum_{w'\in\mathcal{W}}V\left(s^M(s,x,w)\right)\Pr\{w=w'\}$. The detailed derivation of Bellman's equation is given in Appendix A. The resulting size of SDP is large; the number of different paths the Recognizer can take during the time horizon T is no larger than $|\mathcal{A}|^T$, and therefore the number of different values of π and θ is no larger than $|\mathcal{A}|^T$. Hence, the state space size is $|\mathcal{S}|=T\cdot|\mathcal{A}|\cdot|\mathcal{A}|\cdot|\mathcal{A}|^T=T\cdot|\mathcal{A}|^{T+2}$. The size of the information space is $|\mathcal{W}|=3\cdot|\mathcal{A}|+2$. While in principle a SDP policy can be determined using the Backward DP Algorithm (see for example [3], p. 50), most situations result in a

model that renders that algorithm impractical due to its run time of $O\left(T \cdot |\mathcal{A}|^{T+3} \cdot \left(3 \cdot |\mathcal{A}| + 2\right)\right)$, which is exponential in the number of time periods. Thus, we consider a heuristic algorithm.

4. Heuristic Algorithm and Model Relaxation

In this section we develop a simple greedy heuristic for solving SDP and examine its effectiveness using a relaxation. Of course, numerous heuristics could be considered, but in this paper we focus on modeling and do not explore such possibilities further.

4.1 Heuristic Algorithm

For any state $s = (t, r, i, \pi, \theta)$, we define the heuristic policy

$$x^H(s) \in \arg\max_{a \in \mathcal{A}} \left\{ \frac{\hat{\theta}_a}{T^R_{r,a} + 1 + T^I_{i,a}} \right\} \tag{19}$$

where the numerator ($\hat{\theta}_a$) is the probability of a target in AC a at the time the Recognizer reaches AC a computed by (6), and the denominator is the approximated total time to interception. This is a greedy policy that balances the likelihood of a target in a certain AC and the "cost" in time that such a visit would incur. In related search situations similar greedy policies are proven to be optimal (see e.g., [6], [14]).

4.2 Model Relaxation

The heuristic policy results in a lower bound on the optimal value of SDP. To assess the quality of that heuristics, we define a relaxation of SDP, denoted by rSDP, which provides an upper bound for the SDP policy. In rSDP, a decision *x* is identical to that in SDP, and the information is similar, but its probability distribution is different. The state transition functions and the Bellman equations are essentially the same.

The main difference between SDP and rSDP is that the state space in the latter becomes considerably smaller by eliminating the two probability vectors π and θ. Each time a decision is fathomed, we "reset" the two probability vectors π and θ to their initial, steady-state values at time $t = 0$ and therefore these two vectors need not be part of the state vector. In other words, the Recognizer is memory-less. By not nullifying the

probabilities in an AC following a visit (see Sections 3.2 and 3.3), rSDP assigns each ACs a probability of containing a target no smaller than the corresponding probability in SDP. Hence, rSDP is a relaxation of SDP. Having this memory-less property, rSDP may generate a policy that "traps" the Recognizer in an AC that has a high probability of a target. To avoid these traps in rSDP, we temporarily drop the probability of an object in the Recognizer's AC down to 0. This temporary update holds until the current decision is fathomed. Once we complete the current state transition, we ignore this temporary update and reset to the steady-state probabilities. We next define rSDP, where bars are used to denote parameters and variables.

We define a state in rSDP by $\bar{s} = (\bar{t}, \bar{r}, \bar{i})$, where $(\bar{t}, \bar{r}, \bar{i})$ are the time, Recognizer's location, and Interceptor's location, respectively. The state space is denoted by $\bar{\mathcal{S}}$. As in SDP, a decision $\bar{x} \in \mathcal{A}$ is selecting the next AC to be visited by the Recognizer. Let the random vector $\bar{w} = (\Delta\bar{t}_w, \bar{r}_w, \bar{i}_w, \bar{z}_w)$ denote the information obtained when a decision is fathomed in rSDP. The definitions of the components of $\bar{w}$ and its space of possible values are exactly the same as in SDP, but the probability mass function is different.

The state transition function $\bar{s}^M : \bar{\mathcal{S}} \times \mathcal{W} \to \bar{\mathcal{S}}$ in rSDP differs from that in SDP because $\bar{x}$ is not included explicitly as an argument of the function but only implicitly by affecting the probability mass function of $\bar{w}$. We define $\bar{s}^M(\bar{s}, \bar{w}) = (\bar{t} + \Delta\bar{t}_w, \bar{r}_w, \bar{i}_w)$, where $\bar{s} = (\bar{t}, \bar{r}, \bar{i})$ is the state and $\bar{w} = (\Delta\bar{t}_w, \bar{r}_w, \bar{i}_w, \bar{z}_w)$ is the obtained information. The reward $\bar{c} : \mathcal{W} \times \bar{\mathcal{S}} \to \mathbb{R}$, which is a function of $\bar{w}$ and the state $\bar{s}$, is defined by

$$\bar{c}(\bar{w}, \bar{s}) = \begin{cases} \bar{z}_w \cdot (1+\gamma)^{-(\bar{t}+\Delta\bar{t}_w)}, & \bar{t} + \Delta\bar{t}_w \le T \\ 0, & \bar{t} + \Delta\bar{t}_w > T \end{cases} \tag{20}$$

The value $\bar{V}(\bar{s})$ is given by the Bellman equation

$$\bar{V}(\bar{s}) = \begin{cases} \max_{\bar{x}} \left\{ E\left[\bar{c}(\bar{w}, \bar{s}) + \bar{V}(\bar{s}^M(\bar{s}, \bar{w})) \right] \right\}, & \bar{t} < T \\ 0, & \bar{t} \ge T \end{cases} \tag{21}$$

Computing $\bar{V}(\bar{s})$ for rSDP is similar to computing $V(s)$ in SDP. The only difference is the change in probability mass function. Specifically, the updated probabilities θ and π are replaced by

$$\bar{\theta}_a^0 = \begin{cases} \theta_a^0, & a \neq \bar{r} \\ 0, & a = \bar{r} \end{cases} \tag{22}$$

$$\bar{\pi}_a^0 = \begin{cases} \pi_a^0, & a \neq \bar{r} \\ 0, & a = \bar{r} \end{cases} \tag{23}$$

where θ^0 and π^0 are the steady-state probabilities; see (3) and (4). The derivation of Bellman equation for rSDP is given in Appendix B. The state space in rSDP has cardinality $\left|\bar{\mathcal{S}}\right| = T \cdot \left|\mathcal{A}\right|^2$ and the run time of the backward DP algorithm is $O\left(T \cdot \left|\mathcal{A}\right|^3 \cdot \left(3 \cdot \left|\mathcal{A}\right| + 2\right)\right)$. Hence, solving rSDP may be possible in reasonable time.

5. Model Implementation

We consider a maritime interdiction mission in an AOI comprising 25 ACs and a time horizon of 48 time steps. We also briefly consider a situation with 64 ACs. The relaxation rSDP is in these situations a tractable dynamic program and is optimally solved using the Backward DP Algorithm (see for example [3], p. 50). Direct calculation of the value of the heuristic policy is impractical and we estimate it by Monte-Carlo simulation. All models and algorithms were implemented and analyzed using MATLAB on a MacBook Pro with Dual-Core 2.53GHz CPU and 4GB of RAM.

5.1 Scenario Data

We are unable to present results for actual interdiction missions due to security constraints on operational data. However, we generate realistic scenarios based on unclassified information we obtained from active-duty naval officers who have operational experience with counter-drug operations [2]. The analysis comprises a base scenario, and several variations thereof. The baseline scenario represents a strait-like AOI, with land on the North and South edges of the AOI (i.e., no arrivals from or departures to the North and South of the AOI). The AOI is a square grid comprising 25 ACs, each of size 5nm x 5nm. The time horizon is 12 hours, divided into 48 time steps of

15 minutes each. Arrivals are only possible to ACs 1–10, that is $\alpha_a = \beta_a = 0$ for $a = 11,...,25$. We assume that $\alpha_a = .05, \beta_a = .01$ for $a = 1,...,10$. The transition probabilities of neutrals (*P*) and targets (*Q*) are different, representing different movement patterns. In a single time period, an object can only move to one of the four immediate neighboring ACs, or remain in the current AC. We assume that neutrals tend to move along the strait (West-East traffic), while targets tend to move perpendicular to the shipping lanes (North-South traffic).

For any object the probability to stay in its AC during a time-step is 0.1 and the transition probability East (North) is equal to the transition probability West (South). For neutrals these probabilities are 0.3 East and 0.15 North, while for targets these probabilities are reversed. Objects exiting the AOI do not return.

In the base scenario both the Recognizer and the Interceptor start in center AC. We assume that the Interceptor has roughly the same velocity as both the neutrals and targets, which is one AC per time period (approximately 20 knots in real-life). The Recognizer velocity is assumed to be four times the velocity of the Interceptor. The Recognizer's and Interceptor's transition times between ACs include the travel time and processing time (detection time for the Recognizer and boarding time for the Interceptor).

The Recognizer's sensor takes three glimpses at a tracked object ($g = 3$). The false positive and false negative detection probabilities of a target are 0.2 ($u = v = 0.8$). The discount factor is $\gamma = 0.05$, which means that the reward obtained from a target intercepted at the end of the 12 hours time horizon is approximately $1/10$ of the reward obtained at $t = 0$. The value of the probability threshold M for calling in the Interceptor is varied to examine its effects on the results.

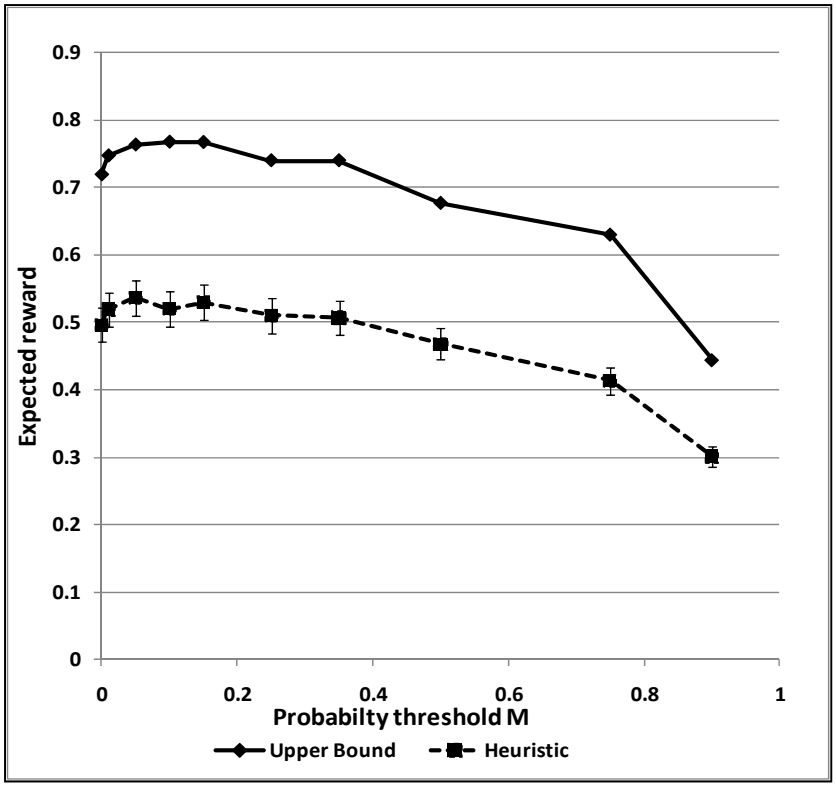

Figure 2. Expected rewards for Heuristic and Upper Bounding (rSDP) policies in Baseline scenario for various probability threshold values *M*

With the given hardware and software, rSDP is solved in approximately 30 minutes and estimating the expected total reward under the heuristic policy, using Monte Carlo simulation and stopping when the 95% confidence interval has width less than 5% of its center, needs about 6 minutes.

In addition to the base scenario, we also considered scenarios with zero-discounting, longer transition time for the Interceptor, 96-hour time horizon, and an 1600nm^2 AOI.

5.2 Numerical Results

We first examine the performance of the heuristic policy described in Section 4.1. Recall that the heuristic and rSDP policies provide lower and upper bounds, respectively, for the optimal expected reward of SDP. Figure 2 presents the expected reward for both policies in the baseline scenario, using various threshold values of *M*. The error bars in Figure 2 (and later in Figures 3 and 4) represent 95% confidence intervals of the estimated expected reward following the heuristic policy. The average gap between the two

expected rewards is about 30%, with relatively little sensitivity to the choice of M . This means that the heuristic policy results in an expected reward that is at least 70% of the optimal value in these situations.

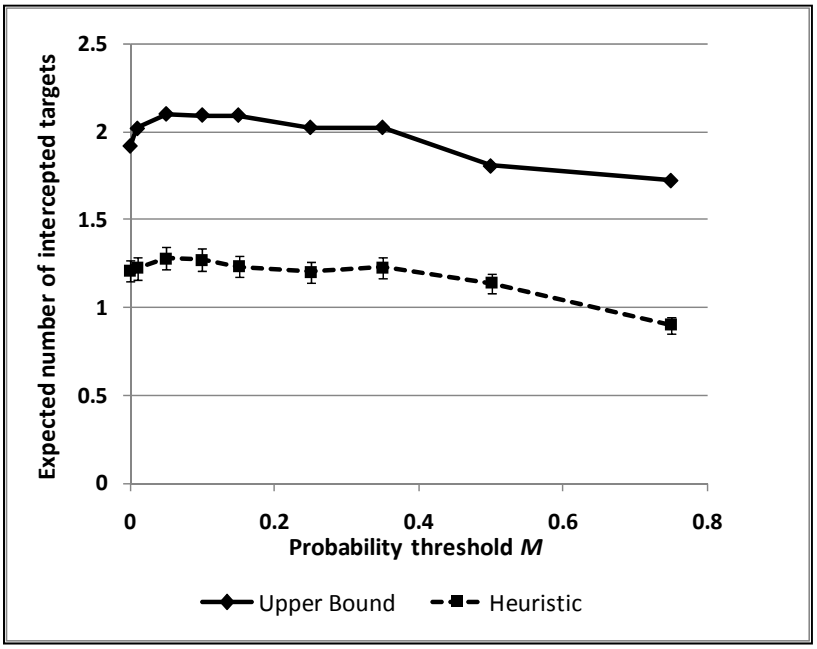

Figure 3. Expected rewards for Heuristic and Upper Bounding (rSDP) policies in a no-discounting scenario for various probability threshold values *M*

Figure 3 represents the same results for the case with a discount factor of zero. In this case the gap is slightly larger than in the baseline scenario, with an average gap of about 40%. The shapes of the graphs in Figures 2 and 3 are similar. The slightly better performance of the heuristics when discounting time may be explained by the greater focus on near-term rewards, rather than long-term, in SDP in that case.

From the baseline scenario (Figure 2), we observe that the expected reward is monotonically decreasing in the probability threshold *M* for $M \geq 0.05$. In other words, larger thresholds (than 0.05) result in worse performance of the interdiction force. This observation appears to be counter intuitive, as one would expect a larger threshold to be more efficient so that the Interceptor and the Recognizer do not waste time dealing with unlikely targets. In order to better understand these counter intuitive result, we evaluated two additional scenarios with longer interception times that are results of a longer on-

board inspection time ("boarding time"). Figure 4 compares the results of three interception times: (1) base scenario, (2) base scenario + 5 time periods, (3), base scenario + 20 time periods.

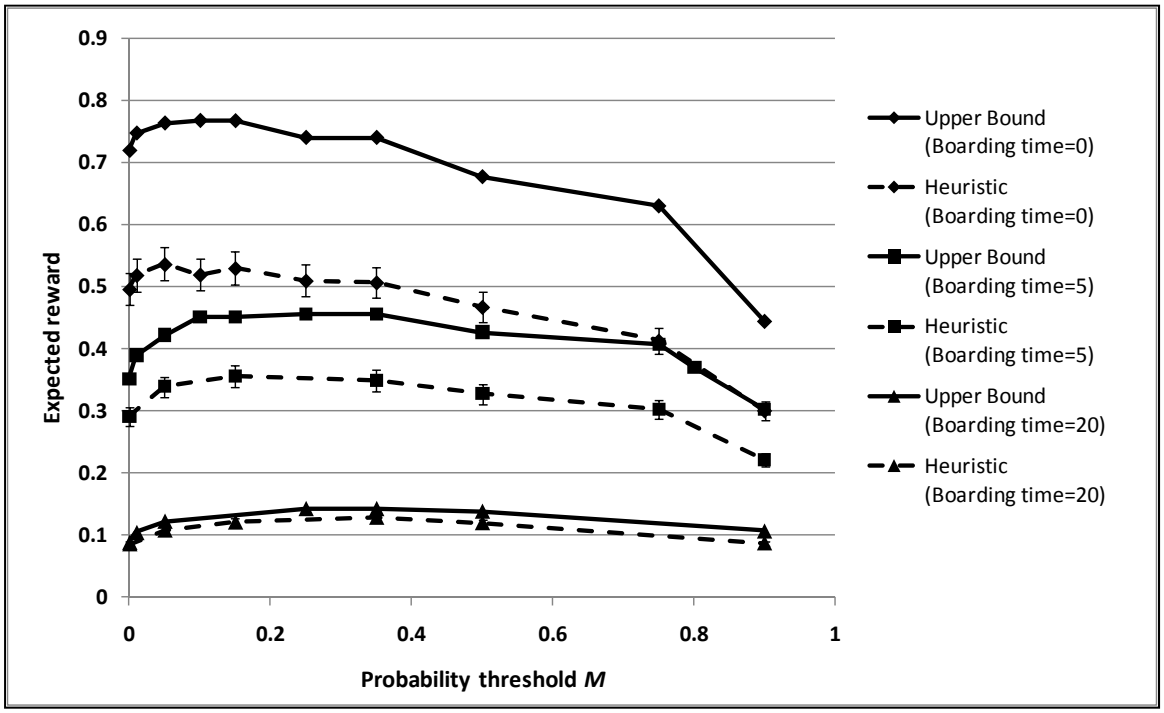

Figure 4. Sensitivity of expected reward for Heuristic and Upper Bounding (rSDP) policies to boarding time. (*x* marks scenarios which have not been calculated)

A threshold value of approximately 0.2 appears to be the best threshold in the scenario with boarding time of five time periods, while a value of approximately 0.4 is the best threshold in the scenario with boarding times of 20 time periods. In any case, the threshold M is relatively small. This result is consistent with common practice in which even the slightest suspicion triggers investigation. In a sparsely populated environment, such as the one modeled in this analysis, it is "better to be safe than sorry," even at the expense of many false alarms.

Finally, we investigate the heuristic's performance for a longer time horizon and larger AOI, where all other parameters remain the same as in the base scenario. For a 24 hour scenario ($T = 96$ time periods), the heuristic's expected reward is approximately 0.57 (with 95% confidence interval of width less than 0.03) and that of rSDP is 0.85, with a

gap of 33%, which is similar to the gap in the shorter scenario. For a 1600nm^2 AOI, with 8nm-by-8nm ACs. The heuristic expected reward is approximately 0.45 (with 95% confidence interval of width 0.02), while that of rSDP is 0.62, with a gap of 28%, which is also in agreement with previously presented results.

6. Conclusions

We developed a stochastic DP model for a combined search and interdiction operation. The operation comprises an airborne sensor for detection, identification, and tracking of suspected objects and a surface vessel or ground vehicle for subsequent interception. While the model is rich and reflects real-world military and naval operations, it is also intractable by standard algorithms. Thus, we developed a greedy heuristic policy, which results in a lower bound on the optimal expected number of successful interdictions within the planning horizon, and a relaxation of the model, which generates an upper bound. We show that for certain realistic maritime interdiction scenarios the gap between the two bounds is in the range of 30% - 40%. The study provides the operational insight that the threshold for triggering investigation by the surface vessel is quite low. For realistic situations examined in this paper, a target (posterior) probability as low as 0.1 after tracking and identification by the airborne sensor should result in interception of the potential target by the surface vessel.

Acknowledgements

The first two authors acknowledge financial support from the Office of Naval Research. The authors are thankful for invaluable advice regarding maritime interdiction operations from several active-duty and retired naval officers. We especially thank CDR E. Lednicky, CAPT J. Kline (Ret.), LCDR D. Bessman, and LT P. Gift as well as personnel at Joint Interagency Task Force South. Opinions and statements expressed in this paper, however, are solely those of the authors.

References

[1] Conway, J. T., Roughead, G., and Allan, T. W., “Naval Operations Concept – 2010, Implementing the Maritime Strategy”, *www.navy.mil/maritime/noc/NOC2010.pdf*

[2] Lednicky, E. Private communication with CDR Lednicky, United States Navy, June 9 and 22, 2010.

[3] W.B. Powell, *Approximate Dynamic Programming: Solving the curses of dimensionality*, Hoboken, NJ: Wiley-Interscience, 2006.

[4] B. O. Koopman, "Search and Screening", Operations Evaluation Group Report No. 56, Center for Naval Analysis, Rosslyn, VA, 1946.

[5] A.R. Washburn, *Search and Detection*, 4th Ed., Linthicum, MD: INFORMS, 2002.

[6] L.D. Stone, *Theory of Optimal Search*, 2nd Ed., Linthicum, MD: INFORMS, 2004.

[7] L. F. Bertuccelli and J. P. How, "Robust UAV Search for Environments with Imprecise Probability Maps," *Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, pp. 5680- 5685, Seville, Spain, 2005.

[8] E. Wong, F. Bourgault and T. Furukawa, "Multi-vehicle Bayesian search for multiple lost targets," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pp. 3169-3174, Barcelona, Spain, 2005.

[9] T. Furukawa, F. Bourgault, B. Lavis, and H. F. Durrant-Whyte, "Recursive bayesian search and tracking using coordinated UAVs for lost targets," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 2521-2526, Orlando, Florida, 2006.

[10] A. R. Washburn, "Piled-Slab Searches", *Operations Research*, vol. 54, pp 1193-1200, 2006.

[11] J. R. Riehl, G. E. Collins, and J. P. Hespanha, "Cooperative graph-based model predictive search," *Proceedings of the 46th IEEE Conference on Decision and Control*, pp. 2998-3004, New Orleans, Louisiana, 2007.

[12] D. A. Anisi, P. Ogren, X. Hu and T. Lindskog, "Cooperative Surveillance Missions with Multiple Unmanned Ground Vehicles (UGVs)," *Proceedings of the 47th IEEE Conference on Decision and Control*, pp. 2444-2449, Cancun, Mexico, 2008.

[13] G. Hollinger and S. Singh, "Proofs and experiments in scalable, near-optimal search by multiple robots," *Proceedings of Robotics: Science and Systems Conference*, Zurich, Switzerland, 2008.

[14] M. Kress, K. Lin and R. Szechtman, "Optimal Discrete Search with Imperfect Specificity", *Mathematical Methods of Operations Research*, vol. 68, pp 539-549, 2008.

[15] M. Kress, R. Szechtman, and J.S. Jones, "Efficient Employment of Non-Reactive Sensors," *Military Operations Research*, vol. 13, no. 4, pp 45-57, 2008.

[16] H. Lau, S. Huang, and G. Dissanayake, "Discounted mean bound for the optimal searcher path problem with non-uniform travel times", *European Journal of Operational Research*, vol. 190, no. 2, pp 383-397, 2008.

[17] B. J. Moore and K. M. Passino, "Decentralized Redistribution for Cooperative Patrol", *Int. J. Robust Nonlinear Control*, vol. 18, pp 165-195, 2008.

[18] J.O. Royset and H. Sato, "Route Optimization for Multiple Searchers," *Naval Research Logistics*, vol. 57, No. 8, pp 701-717.

[19] H. Sato and J.O. Royset, "Path Optimization for the Resource-Constrained Searcher," *Naval Research Logistics*, vol. 57, vol. 4, pp 422-440.

[20] L.M. Wein and M.P. Atkinson, "The Last Line of Defense: Designing Radiation Detection-interdiction Systems to Protect Cities From a Nuclear Terrorist Attack", *IEEE Transactions on Nuclear Science*, vol. 54, no. 3, pp 654-669, 2007.

[21] D. Jeffcoat, P. Krokhmal and O. Zhupanska, "Effects of Cueing in Cooperative Search", *Naval Research Logistics*, vol. 53, no. 8, pp 814-821, 2006.

[22] J. Barton, C. Chiu, S. Martin, W. Johnson, and R. Christansen, "Cooperative Hunter/Killer UAS Demonstration," AUVSI Unmanned North America, San Diego, CA, June 2008.

[23] J. Reimann and G. Vachtsevanos, "UAVs in Urban Operations: Target Interception and Containment", *J. Intelligent and Robotics Systems*, vol. 47, no. 4, pp 383-396, 2006.

[24] Y. Jin, Y. Liao, M.M Polycarpou, and A.A. Minai, "Balancing Search and Target Response in Cooperative UAV Teams," *Proceedings of the 43th IEEE Conference on Decision and Control*, pp. 2923-2928, Atlantis, Bahamas, 2004.

[25] S. Ferrari, R. Fierro, B. Perteet, C. Cai, and K. Baumgartner, "A geometric optimization approach to detecting and intercepting dynamic targets using a mobile sensor network", *SIAM J. Control and Optimization*, vol. 48, no. 1, pp 292-320, 2009.

[26] A. K.-P. Sun, *Cooperative UAV Search and Intercept*, Master's Thesis, University of Toronto, Toronto, Canada, 2009.

[27] M.J. Jackson, *Wide-Area Surveillance Systems using a UAV Helicopter Interceptor and Sensor Placement Planning Techniques*, Master's Thesis, University of Tennessee, Knoxville, Tennessee, 2008.

[28] M. Pachter, P. R. Chandler, and S. Darbha, "Optimal MAV Operations in an Uncertain Environment", *International J. Robust Nonlinear Control*, vol. 18, pp 248-262, 2008.

[29] T.H. Chung, M. Kress, and J.O. Royset, "Probabilistic Search Optimization and Mission Assignment for Heterogeneous Autonomous Agents," *Proceedings of the International Conference on Robotics and Automation, Kobe, Japan*, 2009.

[30] S.L. Smith, *Task Allocation and Vehicle Routing in Dynamic Environments*, Santa Barbara, CA: University of California, 2009.

[31] S. Nair, K. R. Chevva, H. Owhadi, and J. Marsden, "Multiple Target Detection using Bayesian Learning," *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 8549-8554 , Shanghai, P.R. China, 2009.

[32] L. D. Stone, C. A. Barlow, and T. Corwin, *Bayesian Multiple Target Tracking*, Norwood, MA, Artech House, 1999.

[33] A. Yilmaz, O. Javed, and M. Shah, "Object tracking: a survey," *ACM Computing Surveys*, vol. 38, no. 4, pp. 1-45, 2006.

[34] R. Vidal, O. Shakernia, H.J. Kim, D.H. Shim, and S. Sastry, "Probabilistic Pursuit-Evasion Games: Theory, Implementation, and Experimental Evaluation," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 5, pp. 662-669, 2002.

[35] F. Belkhouche and B. Belkhouche, "A Control Strategy for Tracking-Interception of Moving Objects using Wheeled Mobile Robots," *Proceedings of the 43th IEEE Conference on Decision and Control*, pp. 2129-2130 , Atlantis, Bahamas, 2004.

[36] D. Li, J. B. Cruz Jr, and C. J. Schumacher, "Stochastic Multi-Player Pursuit–Evasion Differential Games", *International J. Robust Nonlinear Control*, vol. 18, pp 218–247, 2008.

[37] T.G. Reynolds and R. J. Hansman, *Investigating Conformance Monitoring Issues in Air Traffic Control Using Fault Detection Approaches*, Report No. ICAT-2003-5, Department of Aeronautics & Astronautics, Massachusetts Institute of Technology, Cambridge, Massachusetts, 2003.

[38] S. Meyn, A. Surana, Y. Lin, and S. Narayanan, "Anomaly Detection using Projective Markov Models in a Distributed Sensor Network," *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 4662-4669, Shanghai, P.R. China, 2009.

[39] X. Jin, S. Sarkar, K. Mukherjee, A. Ray, "Suboptimal Partitioning of Time-Series Data for Anomaly Detection," *Proceedings of the 48th IEEE Conference on Decision and Control*, pp. 1020-1025, Shanghai, P.R. China, 2009.

Appendix A

This Appendix provides details about the calculations of Bellman's equation in SDP; see Section 3.5. For notational convenience, we define for any $x \in \mathcal{A}$, $s = (t, r, i, \pi, \theta)$, $w = (\Delta t_w, r_w, i_w, z_w)$, $\tilde{c}(\Delta t_w, z_w, s) \equiv c(w, s)$, and $\tilde{s}^M(s, x, \Delta t_w, r_w, i_w) \equiv s^M(s, x, w)$. Using these functions, we find that

$$E[c(w,s)] = E[\tilde{c}(\Delta t_w, z_w, s)] = \sum_{z'_w=0}^{1} \sum_{\Delta t'_w=0}^{\infty} \tilde{c}(\Delta t'_w, z'_w, s) \Pr\{\Delta t_w = \Delta t'_w, z_w = z'_w\} \tag{24}$$

$$\begin{aligned} &E\left[V\left(s^M(s,x,w)\right)\right] \\ &= E\left[V\left(\tilde{s}^M(s,x,\Delta t_w, r_w, i_w)\right)\right] = \sum_{\Delta t'_w=0}^{\infty} \sum_{r'_w \in \mathcal{A}} \sum_{i'_w \in \mathcal{A}} V\left(\tilde{s}^M(s,x,\Delta t'_w, r'_w, i'_w)\right) \Pr\{\Delta t_w = \Delta t'_w, r_w = r'_w, i_w = i'_w\} \end{aligned} \tag{25}$$

where

$$\Pr\{\Delta t_w = \Delta t'_w, z_w = z'_w\} = \sum_{r'_w \in \mathcal{A}} \sum_{i'_w \in \mathcal{A}} \Pr\{\Delta t_w = \Delta t'_w, r_w = r'_w, i_w = i'_w, z_w = z'_w\} \tag{26}$$

$$\Pr\{\Delta t_w = \Delta t'_w, r_w = r'_w, i_w = i'_w\} = \sum_{z'_w=0}^{1} \Pr\{\Delta t_w = \Delta t'_w, r_w = r'_w, i_w = i'_w, z_w = z'_w\} \tag{27}$$

Using (27) and the probability mass function of w, we find that

$$E\left[V\left(s^M(s,x,w)\right)\right] = (I)(II) + \sum_{j \in \mathcal{A}} \left((III)(IV)\right) + \sum_{j \in \mathcal{A}} \left((V)(VI)\right) + (VII)(VIII)$$

where

$$
\begin{aligned}
(I) &= V\left(\tilde{s}^M\left(s, x, T^R_{r,x}, x, i\right)\right) \\
(II) &= \left(1-\hat{\pi}_x-\hat{\theta}_x\right) \\
(III) &= V\left(\tilde{s}^M\left(s, x, T^R_{r,x}+1+T^I_{i,j}, j, j\right)\right) \\
(IV) &= \sum_{n'=0}^{n^*_j}\left(\binom{g}{n'}(1-u)^{n'}u^{g-n'}Q(x,j)\hat{\theta}_x+\binom{g}{n'}v^{n'}(1-v)^{g-n'}P(x,j)\hat{\pi}_x\right) \\
(V) &= V\left(\tilde{s}^M\left(s, x, T^R_{r,x}+1, j, i\right)\right) \\
(VI) &= \sum_{n'=n^*_j+1}^{g}\left(\binom{g}{n'}(1-u)^{n'}u^{g-n'}Q(x,j)\hat{\theta}_x+\binom{g}{n'}v^{n'}(1-v)^{g-n'}P(x,j)\hat{\pi}_x\right) \\
(VII) &= V\left(\tilde{s}^M\left(s, x, T^R_{r,x}+1, x, i\right)\right) \\
(VIII) &= Q(x,\mathcal{A}_0)\hat{\theta}_x+P(x,\mathcal{A}_0)\hat{\pi}_x
\end{aligned}
\tag{28}
$$

Similarly, we can use (26) and the probability mass function of w to compute

$$
E\left[c(w,s)\right]=\hat{\theta}_x\sum_{j\in\mathcal{A}}\left((1+\gamma)^{-\left(t+T^R_{r,x}+1+T^I_{i,j}\right)}Q(x,j)\sum_{n'=0}^{n^*_j}\binom{g}{n'}(1-u)^{n'}(u)^{g-n'}\right) \tag{29}
$$

Appendix B

In this Appendix we provide details about the calculations of Bellman's equation for rSDP. Let $\hat{\bar{\pi}}_{\bar{x}}$ denote the probability given by (5) when t' and π^t_a are replaced by $t+T^R_{r,x}$ and $\bar{\pi}^0_a$, respectively. Moreover, we let $\hat{\bar{\theta}}_{\bar{x}}$ denote the probability given by (6) when t' and θ^t_a are replaced by $t+T^R_{r,x}$ and $\bar{\theta}^0_a$. Substituting θ and π with (22) and (23) in (28) and (29), respectively, while explicitly computing the next state using the state transition function, we get the following formulas for computing the Bellman equation for rSDP:

$$
E\left[\bar{V}\left(\bar{t}+\Delta\bar{t}_w, \bar{r}_w, \bar{i}_w\right)\right]=(I)(II)+\sum_{j\in\mathcal{A}}\left((III)(IV)\right)+\sum_{j\in\mathcal{A}}\left((V)(VI)\right)+(VII)(VIII)
$$

where $\bar{s}^M\left(\bar{s},\bar{w}\right)=\left(\bar{t}+\Delta\bar{t}_w, \bar{r}_w, \bar{i}_w\right)$ is the next state given state $\bar{s}=\left(\bar{t},\bar{r},\bar{i}\right)$ and realization $\bar{w}=\left(\Delta\bar{t}_w, \bar{r}_w, \bar{i}_w, \bar{z}_w\right)$ and

$$
\begin{aligned}
(I) &= \bar{V}\left(\bar{t} + T^R_{\bar{r},\bar{x}}, \bar{x}, \bar{i}\right) \\
(II) &= \left(1 - \hat{\bar{\pi}}_{\bar{x}} - \hat{\bar{\theta}}_{\bar{x}}\right) \\
(III) &= \bar{V}\left(\bar{t} + T^R_{\bar{r},\bar{x}} + 1 + T^I_{\bar{i},j}, j, j\right) \\
(IV) &= \left(\sum_{n'=0}^{n^*_j} \left(\binom{g}{n'}(1-u)^{n'} u^{g-n'} Q(\bar{x}, j)\hat{\bar{\theta}}_{\bar{x}} + \binom{g}{n'} v^{n'}(1-v)^{g-n'} P(\bar{x}, j)\hat{\bar{\pi}}_{\bar{x}}\right)\right) \\
(V) &= \bar{V}\left(\bar{t} + T^R_{r,x} + 1, j, \bar{i}\right) \\
(VI) &= \left(\sum_{n'=n^*_j+1}^{g} \left(\binom{g}{n'}(1-u)^{n'} u^{g-n'} Q(\bar{x}, j)\hat{\bar{\theta}}_{\bar{x}} + \binom{g}{n'} v^{n'}(1-v)^{g-n'} P(\bar{x}, j)\hat{\bar{\pi}}_{\bar{x}}\right)\right) \\
(VII) &= \bar{V}\left(\bar{t} + T^R_{\bar{r},\bar{x}} + 1, \bar{x}, \bar{i}\right) \\
(VIII) &= \left(Q(\bar{x}, \mathcal{A}_0)\hat{\bar{\theta}}_{\bar{x}} + P(\bar{x}, \mathcal{A}_0)\hat{\bar{\pi}}_{\bar{x}}\right)
\end{aligned}
\tag{30}
$$

and

$$
E\left[\bar{c}\left(\bar{w}, \bar{s}\right)\right] = \hat{\bar{\theta}}_{\bar{x}} \sum_{j \in \mathcal{A}} \left((1+\gamma)^{-\left(\bar{t} + T^R_{\bar{r},\bar{x}} + 1 + T^I_{\bar{i},j}\right)} Q(\bar{x}, j) \sum_{n'=0}^{n^*_j} \binom{g}{n'} (1-u)^{n'} (u)^{g-n'} \right) \tag{31}
$$

The expressions in (30) are completely parallel to those of (28). The state space in rSDP has cardinality $\left|\bar{\mathcal{S}}\right| = T \cdot \left|\mathcal{A}\right|^2$ and the run time of the backward DP algorithm is $O\left(T \cdot \left|\mathcal{A}\right|^3 \cdot \left(3 \cdot \left|\mathcal{A}\right| + 2\right)\right)$, which is much faster than that for SDP.