NAVAL
POSTGRADUATE
SCHOOL

MONTEREY, CALIFORNIA

THESIS

OPTIMAL RANDOMIZED SURVEILLANCE PATTERNS TO DETECT INTRUDERS APPROACHING A MILITARY INSTALLATION

by

Trevor McLemore

June 2007

Thesis Advisor: Kyle Y. Lin
Second Reader: W. Matthew Carlyle

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
June 2007

3. REPORT TYPE AND DATES COVERED
Master's Thesis

4. TITLE AND SUBTITLE Optimal Randomized Surveillance Patterns to Detect Intruders Approaching a Military Installation

5. FUNDING NUMBERS

6. AUTHOR(S)
McLemore, Trevor D.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Naval Postgraduate School
Monterey, CA 93943-5000

8. PERFORMING ORGANIZATION REPORT NUMBER

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

10. SPONSORING/MONITORING AGENCY REPORT NUMBER

11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.

12a. DISTRIBUTION / AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited

12b. DISTRIBUTION CODE

13. ABSTRACT (maximum 200 words)

This thesis addresses a two-person zero-sum game between an intruder and a defender of a military installation. The intruder attempts to penetrate the military installation by choosing one of its many entry points, each of which is monitored by a surveillance camera and may require a different amount of time to transit. Although the real-time video of each surveillance camera is fed to a surveillance room simultaneously, the defender has only one surveillance monitor and can monitor only one entry point at a time. We consider a discrete-time model such that the intruder will be detected if, during his travel time, the defender spends one time unit monitoring the entry point chosen by the intruder. The problem facing the defender is how to switch among entry points to monitor from one time unit to the next, in order to maximize the detection probability of the intruder. The intruder's goal is, of course, to infiltrate without being detected, and so he wishes to minimize this probability.

We formulate the problem as a two-person zero-sum game, and develop a linear program to solve it. Numerical experiments provide insights into the design of such surveillance systems.

14. SUBJECT TERMS search and detection, border patrol, camera surveillance

15. NUMBER OF PAGES
47

16. PRICE CODE

17. SECURITY CLASSIFICATION OF REPORT
Unclassified

18. SECURITY CLASSIFICATION OF THIS PAGE
Unclassified

19. SECURITY CLASSIFICATION OF ABSTRACT
Unclassified

20. LIMITATION OF ABSTRACT
UL

THIS PAGE INTENTIONALLY LEFT BLANK

OPTIMAL RANDOMIZED SURVEILLANCE PATTERNS TO DETECT INTRUDERS APPROACHING A MILITARY INSTALLATION

Trevor D. McLemore
Ensign, US Navy
B.S. United States Naval Academy, 2006

Submitted in partial fulfillment of the
requirements for the degree of

**MASTER OF SCIENCE IN APPLIED SCIENCE
(OPERATIONS RESEARCH)**

from the

**NAVAL POSTGRADUATE SCHOOL
June 2007**

Author: Trevor McLemore

Approved by: Kyle Y. Lin
Thesis Advisor

W. Matthew Carlyle
Second Reader

James N. Eagle
Chairman, Department of Operations Research

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

This thesis addresses a two-person zero-sum game between an intruder and a defender of a military installation. The intruder attempts to penetrate the military installation by choosing one of its many entry points, each of which is monitored by a surveillance camera and may require a different amount of time to transit. Although the real-time video of each surveillance camera is fed to a surveillance room simultaneously, the defender has only one surveillance monitor and can monitor only one entry point at a time. We consider a discrete-time model such that the intruder will be detected if, during his travel time, the defender spends one time unit monitoring the entry point chosen by the intruder. The problem facing the defender is how to switch among entry points to monitor from one time unit to the next, in order to maximize the detection probability of the intruder. The intruder's goal is, of course, to infiltrate without being detected, and so he wishes to minimize this probability.

We formulate the problem as a two-person zero-sum game, and develop a linear program to solve it. Numerical experiments provide insights into the design of such surveillance systems.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF FIGURES

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

This thesis could not have been written without the help of my adviser, Kyle Lin, and my second reader, Matthew Carlyle. Kyle Lin came up with the thesis problem and guided me through the writing process. Matthew Carlyle provided tremendous amounts of help with the programming. The stack-based enumeration algorithm shown in this thesis is derived from one of Matthew Carlyle's class handouts.

THIS PAGE INTENTIONALLY LEFT BLANK

EXECUTIVE SUMMARY

This thesis concerns how to use surveillance cameras optimally to detect intruders that approach a military installation. Surrounding the military installation are many entry points that are vulnerable to an intruder’s penetration. Although each entry point is equipped with one camera that provides continuous surveillance and feeds real-time video back to a surveillance room, the defender in the surveillance room has only one monitor, and can therefore monitor only one entry point at a time. The problem facing the defender is to determine how to switch among different cameras in order to maximize the steady-state probability of detecting an intruder.

We model this problem as a two-person zero-sum game between an intruder and a defender. The intruder chooses one entry point to penetrate, while the defender designs a surveillance pattern among entry point cameras. By formulating the problem as a discrete-time Markov process, it is possible to solve for the optimal policy of both players by a single linear program. However, it becomes computationally intensive to solve problems with a large number of entry points to be monitored because of a dramatic increase in the size of the state space for the underlying Markov process. In addition to the formulation of the optimal policy, this thesis presents an efficient way to compute bounds on the detection probability if the optimal policy cannot be produced with the computing resources available. The techniques used to derive the bounds can also produce effective heuristic policies.

Some optimal surveillance patterns and clear insights into installation protection strategies emerge from numerical experiments. For instance, the defender can increase the detection probability by increasing the time it takes to penetrate through an entry point, or by reducing the number of entry points. One immediate result is that if the defender has resources available to increase the penetration time of only one entry point (through, for example, physical reconfiguration), then it is best to do so to the entry point that has the shortest penetration time.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. BACKGROUND

This thesis models a two-person zero-sum game in which an intruder attempts to infiltrate a defender’s military installation. There are several possible entry points, and the time it takes for the intruder to penetrate through each entry point may be different. Each entry point has a dedicated surveillance camera, but the defender can monitor only one entry point at a time. The intruder chooses the entry point to penetrate, and the defender designs a surveillance pattern that randomly cycles among different surveillance cameras. The intruder’s objective is to maximize the probability of penetrating without getting detected, while the defender’s objective is to minimize this same probability. A simple diagram of a hypothetical military installation is shown in Figure 1. There are 5 entry points and the time it takes to penetrate through these entry points are 2, 4, 4, 4, and 3 time units, respectively.

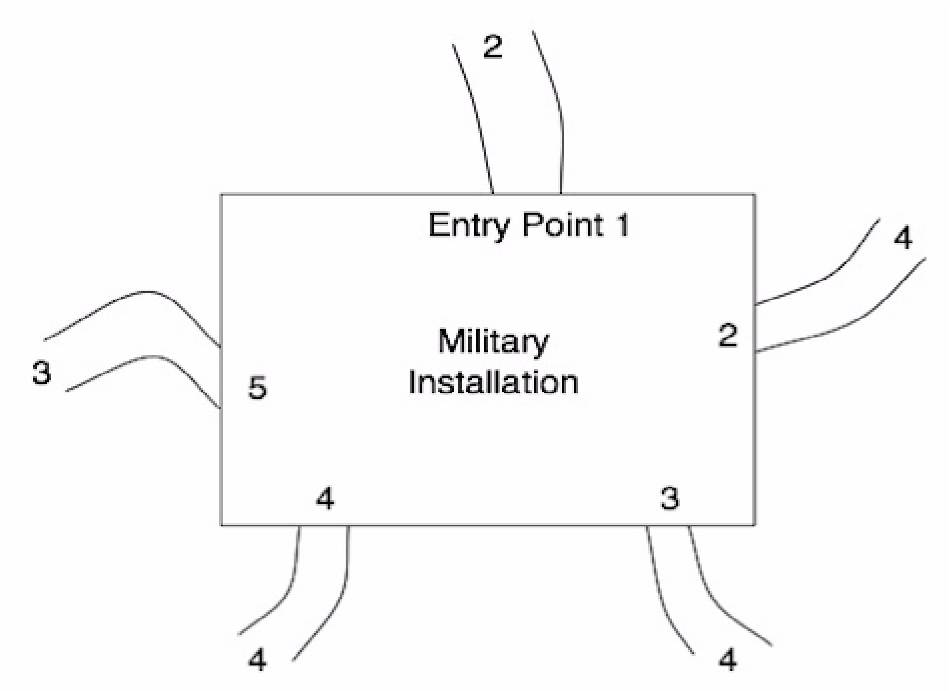

Figure 1. A military installation with 5 entry points; the entry point number is listed inside the installation and the travel times are listed outside the installation for each entry point.

Most existing research on border surveillance focuses on finding detection probabilities of linear patrols [3], [5], [6]. Extensions that maximize the effectiveness of linear patrols usually assume that the behavior of the enemy can be accurately predicted through a probability distribution [2]. In this thesis, we employ a game-theoretic model where the enemy is actively choosing the most effective randomized course of action.

There has been some recent work that uses interdiction models for border patrol problems. This line of research typically looks at a network with different cost arcs crossing the border and the aim is to best allocate a limited number of resources, such as check points, that increase the probability of detection along that arc [1], [4]. The model used in this thesis works similarly to these models in the sense that the defender can allocate resource to different entry points to increase their respective travel times. Unlike other interdiction models, this thesis employs a probabilistic surveillance pattern, which makes it more difficult for the intruder to predict the defender's actions to protect the military installation.

B. OBJECTIVES

Our overall goal is to improve the efficiency of randomized monitoring of security cameras in a secure area. There are two main objectives of this thesis. The first objective is to develop an algorithm to compute the optimal randomized surveillance pattern to maximize surveillance effectiveness against a smart intruder. This objective includes the development of a stochastic model that describes the surveillance problem around a military installation, and the solution to the model with an optimal policy for either player.

The second objective is to conduct sensitivity analysis with numerical experiments to improve security. In other words, the model will be tested over a range of different entry point characteristics. During this sensitivity analysis, we examine how the optimal policy changes, as well as the probability of detection, when the travel time at each entry point changes, or when the number of entry points changes.

C. SCOPE OF MODEL AND ASSUMPTIONS

We assume a discrete-time model, such that one time unit is the minimum time to observe one entry point, and the time to traverse each entry point is an integer multiple of this time; the defender can choose one entry point to monitor in each time unit. Below is a list of assumptions used in our model:

(1) The model is transparent. The intruder knows the layout of the military installation—the number of entry points and their respective travel times. In other words, the intruder has figured out which entry points are available to attempt to pass through unnoticed and has an understanding of the model similar to that of the defender. In reality, it is likely that the intruder can obtain such intelligence about the defender's military installation. Naive intruders will have at best the same probability to make it through unnoticed as the intruder that knows the layout of the military installations. We also make the (very conservative) assumption that the infiltrator can observe the camera entry-point assignments for any finite length of time, and either detect a repeating pattern of observations, or infer a probabilistic transition rule for the observations.

(2) There is one camera per entry point, and all surveillance cameras are assigned to one monitor. There are no split screens so only one camera can be viewed at a time. We realize that if resources are available so that each camera can have its own monitor with enough security personnel to continuously watch them all, there is no need for this model. However, our model is applicable to other detection environments besides secure facility infiltration, and this assumption is an important component of those other applications.

(3) If the defender is monitoring an entry point and an intruder is using that entry point, the intruder is detected (*i.e.*, the probability of detection is always one if the defender observes the correct entry point).

(4) All intruders require the same amount of time to infiltrate through a given entry point.

(5) Each camera takes the same amount of time to scan its responsible entry point and detect all of the intruders present on that entry point.

(6) We adopt a discrete-time model, with the time unit being the amount of time it takes for the defender to scan through one entry point and detect all intruders there. The time it takes for an intruder to penetrate through one entry point needs to be an integer multiple of the time unit. Otherwise, the travel times can be rounded to the nearest integer for an approximation.

D. THESIS ORGANIZATION

In Chapter II, we formulate the model and develop an algorithm to compute the optimal policy for either player. In Chapter III, we carry out numerical experiments and conduct sensitivity analysis. In Chapter IV, we conclude the thesis and point out future research directions

II. METHODOLOGY

In this section, we discuss the model and the notation used. In Section A, we introduce the basic model. In Section B, we discuss how the model is formulated as a linear programming problem. In Section C, we discuss a few issues concerning the implementation of the model to solve problems with many entry points. In Section D we discuss a method to solve for an upper bound.

A. MODEL PRELIMINARIES

Consider a military installation as shown in Figure 1 from the previous chapter. There are *n* possible entry points, with a security camera covering each entry point, where *n* is a positive integer. Intruders (such as terrorists, espionage agents, etc.) make an attempt to infiltrate the installation from the surrounding territory to the interior of the installation by moving unobserved through one of the *n* entry points. The travel time on different entry points can be different because of different camera coverage, the difficulty of getting past certain obstacles such as walls and locked doors, or other factors. For example, in Figure 1, we let *n*=5 and the five entry points are numbered from 1 through 5 in a clockwise fashion, and the travel times on the five entry points are 2, 4, 4, 4, and 3 time units, respectively. Without loss of generality we can represent any situation as the vector of traversal times, e.g. (2, 4, 4, 4, 3).

A defender sits in a surveillance room equipped with a monitor such that the defender can monitor one entry point at a time. We assume that by monitoring an entry point continuously for one time unit, the defender is able to detect all intruders on that entry point. The problem facing the defender is to select a sequence of entry points to monitor, one at a time from one time unit to the next, in order to maximize the long-run detection probability of an intruder.

A surveillance pattern is a sequence of entry points observed by the defender from one time unit to another. For instance, as in Figure 1, a surveillance pattern can be (1, 2, 3, 1, 4, 5), such that the defender spends one time unit observing entry point 1, the next time unit observing entry point 2, and then another time unit each observing entry points 3, 1, 4, and 5, respectively. With a naïve approach the defender could just repeat this surveillance pattern indefinitely. If the infiltrator knows this pattern, he can figure out the time between observations of each zone, and, given the length of each zone, choose the one that has the largest ratio of zone travel time to inter-observation time. For example, in Figure 1, with a naïve surveillance pattern (1, 2, 3, 4, 5)—the defender spends one time unit on each entry point for every five time units—a smart intruder would just take entry point 1. Because penetrating the installation's defenses via entry point 1 takes only 2 time units and the defender only revisits entry point 1 every five time units, the defender has a chance of only 2/5 to detect an intruder. A different surveillance pattern (1, 2, 3, 1, 4, 5) can improve the detection probability to 1/2 (a smart intruder would then take entry point 5 because this surveillance pattern detects 2/3 of intruders that take entry point 1).

A surveillance pattern can be made more complicated by choosing a long sequence of entry points, but at a certain point it repeats, and is therefore predictable by the infiltrator. If we allow the surveillance pattern to be probabilistic, by keeping track of the "state" of the system and allowing probabilistic transitions between entry points based on this state, we avoid this issue of predictability. The difficulty of selecting an effective surveillance pattern lies in the fact that intruders are intelligent and can learn over time. In general, it is difficult to find the optimal surveillance pattern by trial and error. The research problem is to design a probabilistic surveillance pattern among the entry points for the defender to follow so as to maximize this detection probability over any possibly infiltration scheme.

The interaction between the intruder and the defender can then be modeled as a two-person zero-sum game: The intruder's pure strategies are the choices of each entry point, and the defender's strategies are the surveillance patterns (or a finite subset thereof). The intruder's objective is to minimize the probability of getting detected when attempting to penetrate through the chosen entry point, while the defender's objective is to maximize this probability. In general, the solution to this game will involve mixed strategies: the defender will typically choose a randomized surveillance strategy to minimize predictability, and the intruder will, at the very least, be indifferent to several entry points, because they all have the same probability of detection.

B. FORMULATION

There are *n* possible entry points to penetrate the installation's defenses, and the travel time on entry point *i* is denoted by c_i, $i = 1, \ldots, n$. The strategy of the intruder is to choose one of the *n* entry points to take. The strategy of the security monitor is a surveillance pattern, denoted by σ. For instance, a naïve approach is for the defender to cycle through all *n* entry points and repeat the same pattern, in which case the pattern $\sigma = (1, 2, \ldots, n)$.

Let $f(\sigma, i)$ denote the long-run proportion of intruders on entry point *i* that will be detected if the defender follows the surveillance pattern σ. Because the intruders are smart and will choose the safest entry point to avoid detection, the optimization problem facing the defender can be formulated as follows:

$$\max_{\sigma} \min_{i} f(\sigma, i)$$

Similarly from the intruder's standpoint, the optimization problem can be formulated as follows:

$$\min_{i} \max_{\sigma} f(\sigma, i)$$

Based on the theory of a two-person zero-sum game, there exists a pair of mixed strategies that simultaneously solve both of these optimization problems. In other words, a solution to the defender's problem leads to a solution to the intruder's problem and vice versa. We next propose an approach to solve this two-person zero-sum game by a linear program that solves the optimization problem facing the defender.

The first step to solve the optimization problem facing the defender is to properly define a feasible policy. Although a surveillance pattern is an easily understood representation of a security policy, it is mathematically advantageous to use a different policy definition, so that we can use the theory of Markov process. To do so, first note that in each time unit (we consider a discrete-time model), the state of the defender can be delineated by a vector s =(s_1, s_2, …, s_n), with the interpretation that entry point i was last monitored by the defender s_i time units ago, i = 1, 2, …, n. For instance, if n = 3, the state (0, 1, 2) indicates that the defender is currently monitoring entry point 1, was monitoring entry point 2 one time unit ago, and entry point 3 two time units ago. If the defender moves to monitor entry point 2 in the next time unit, written as (0, 1, 2; 2), then the state in the next time unit becomes (1, 0, 3). This state definition provides all information necessary for the defender to decide which entry point to observe in the next time unit.

In order to allow a more flexible formulation, we consider a randomized policy. That is, whenever in state s, the defender will choose entry point i to observe in the next time unit with probability $P(s,i)$. In a two-person zero-sum game, it is often the case that the optimal policy is a randomized policy, because a deterministic policy is easier for the opponent to predict. In addition, if the optimal policy turns out to be a deterministic (nonrandomized) policy, then in the solution each $P(s,i)$ will be either 1 or 0. In other words, a deterministic policy is simply a special case of a randomized policy.

Let (s_1, s_2, …, s_n; i) denote the transition such that the defender is currently in state (s_1, s_2, … s_n) and next selects entry point i to monitor. Let $\pi(s;i)$ denote the limiting (or steady-state) probability that the defender will be in state s and will choose entry point i to monitor in the next time unit. The values of $\pi(s,i)$ need to satisfy the following equations:

$$\pi(s,i) \geq 0, \qquad \forall\ s,\ i. \tag{1}$$

$$\sum_s \sum_i \pi(s,i) = 1 \tag{2}$$

$$\sum_i \pi(s,i) = \sum_{t \in A(s)} \pi(t, L(s)) \quad \forall s. \tag{3}$$

Equation (1) requires the limiting probability to be nonnegative, and Equation (2) requires that all probabilities add up to 1. Equation (3) requires the in-flow (right-hand side) and out-flow (left-hand side) of each state are balanced, where $L(s)$ represents the entry point the defender currently monitors in state *s* (for instance *L*(0, 1, 2) = 1 because the defender is currently observing entry point 1 in state (0, 1, 2)) and

$$A(s) = \{t \mid t_j = s_j - 1, j \neq L(s)\}.$$

This is, of course, an infinite state space; we can have state vectors with extremely large values of s_i for some entry point *i*, by simply never observing that entry point. For any practical application we will observe every entry point within a finite amount of time from any state, and we can limit the size of the state space by placing a (somewhat arbitrary) bound on the value of each s_i. For the remainder of this work we assume that a constant, *gap*, has been defined, and the largest that any s_i can attain is t_i+*gap*. This immediately yields a finite state-space, and it also puts an upper bound on the time between observations of each entry point.

To complete the formulation of this optimization problem, we need to define the objective function. Let R_i denote the long-run proportion (or equivalently, the detection probability in equilibrium) of intruders that the defender can detect on entry point *i*. In state *s*, if the defender chooses entry point *i* to monitor in the next time unit, then the benefit for the defender is to detect those intruders who (1) moved into camera range after entry point *i* was last monitored and (2) have not made it past the security camera's range. Therefore, the benefit can be captured by $\min(c_i, s_i + 1)$. Hence, we can write

$$R_i = \sum_s \pi(s,i)\min(c_i, s_i + 1), \text{ for } i = 1, 2, \ldots, n. \tag{4}$$

Because the intruders are intelligent and will choose the entry point that has the smallest value of R_i , the defender wants the $\min_i R_i$ to be as large as possible. Consequently, we can write the objective function and bounding constraints as follows:

$$\text{maximize } R \tag{5}$$

$$\text{subject to } R \leq R_i, \quad \forall i, \tag{6}$$

where, of course, the values of $\pi(s,i)$ are constrained as in (1)-(3), and the individual R_i are defined by (4).

Given a solution to this model for a specific value of *gap*, we can easily determine if it is optimal for the infinite-state-space version by observing the transitions in the solution: when the maximum time between glimpses for all entry points in the problem is never attained, the solution is optimal. Conversely, if any entry point in the problem is only observed when the maximum time between observations is reached, then we may not conclude the solution is optimal. In the example given in Figure 2, using a *gap* of zero we see the solution is not guaranteed to be optimal because three time units pass before entry point two or three is observed and this travel time is equal to their lengths plus zero.

(0, 1, 3; 3)	0.25
(0, 3, 1; 2)	0.25
(1, 0, 2; 1)	0.25
(1, 2, 0; 1)	0.25

Figure 2. Steady-state probabilities given as the output for the three entry point problem with travel times (1, 3, 3) and a gap of zero.

In fact, we see when this same problem is solved with a *gap* of five (which is sufficiently large) instead of zero, the optimal value is 0.6, given by the solution in Figure 3; it is not the 0.5 probability of detection that is given when the problem is solved with a *gap* of zero.

(0, 2, 1; 1)	0.2
(0, 3, 2; 1)	0.2
(0, 4, 3; 2)	0.2
(1, 0, 4; 3)	0.2
(2, 1, 0; 1)	0.2

Figure 3. Steady-state probabilities given as the output for the three entry point problem with travel times (1, 3, 3) and a gap of five.

Since outputs are given as steady-state probabilities, to implement algorithms for practical use they needs to be converted into surveillance patterns. To convert outputs into surveillance patterns we can start at any state, but for simplicity let us start at the first state in the output. We find the zero and that is the first entry point listed in the surveillance pattern. Then we look at the entry point it transitions to and find the entry point with the zero value and list that entry point as the second entry point in the surveillance pattern. This process continues until the original state vector that was started with is reached and at that point stops. For the output used in Figure 2, (even though this output is not optimal) the surveillance pattern would be (3, 1, 2, 1) and for Figure 3 the surveillance pattern would be (1, 1, 1, 2, 3).

There is a chance that there are two or more of the same state vectors transitioning to different states with steady state probabilities given for each of them. If there are identical state vectors with different transition states and probabilities assigned to both of them, a weighted probability must be taken among all of the potential transition states that can occur next as in Figure 4.

(0, 1, 2; 2)	0.1
(0, 1, 2; 3)	0.2

Figure 4. Example of the same state vector transitioning to two different states for the three entry point problem with travel times (0, 1, 2)

In other words when in state (0, 1, 2), the defender will observe camera 2 next with a probability of 0.333 and will observe camera 3 next with a probability of 0.667. The defender will be in state (0, 1, 2) 3/10 of the time.

C. MODEL IMPLEMENTATION

Because the problem is formulated as a linear program, it is straightforward to solve in most optimization software and the solution given is optimal, as long as all possible transitions are listed. However, before obtaining the optimal solution, it is difficult to know beforehand how large the state space needs to be, because the time between consecutive visits to a particular entry point can be very large, especially if the travel time of that entry point is large. To control the size of the state space we introduce a value called *gap*, which is defined as the number of additional time units beyond each entry point length that can elapse before that entry point must be checked.

The *gap* is very important in this algorithm because it decides how many transition states are to be tested for optimality. One *gap* value is chosen for an entire scenario. If all of the entry point lengths are close to the number of entry points, it is likely an optimal solution will be found with little or no *gap*. The problem comes when the time required to traverse an entry point is much shorter than the number of entry points. Because the number of transitions that can be made is dependent on the entry point length, this situation will show up as infeasible or suboptimal with a zero gap. Of course, we can always solve a sequence of models, with increasing values of *gap*, until we either find a solution that is optimal for the infinite-state-space version, or we cannot handle the resulting state-space based on memory requirements for the linear programming model.

As this problem gets bigger (more zones with higher variance in zone length), it becomes difficult to solve because the number of possible state vectors increases exponentially. In the case where the lengths of all of the zones are equal to the number of zones, the number of state vectors is equal to $n(n!)$. Although this scenario would have probability detection = 1 with a naïve surveillance pattern of (1, 2, …, n), it is a good estimation of the size of the problems. Because of the rapid growth of the number of possible state vectors, it is important to make the algorithm that creates the list of possible state vectors efficiently. The algorithm that is used is shown in Figure 5.

Algorithm: path-enum

```
top = 1;
si[top]=-1;
while(top>0){ //states remain to enumerate
  if(si[top]>=0)
    sUsed[si[top]]--;
  //find next feasible state for this zone
  si[top]++;
  while(si[top]<=ti[top]+gap && sUsed[si[top]]>0)
    si[top]++;
  if(si[top]<=ti[top]+gap){ //we found a feasible state
    sUsed[si[top]]++;
    if(top==zones && sUsed[0]==1){ //we've got a feasible state
      //print out this state vector
      //print transitions from this state vector
    } else if (top<zones){
      //not there yet, move to next zone
      si[++top]=-1;
    }
  } else { //we're done with this zone
    top--;
  }
}
```

Figure 5: pseudo code for stack-based state-enumeration algorithm

To increase the efficiency of the enumeration algorithm, a few rules were implemented to limit the state vectors to only what is feasible and to cut out as many of the state vectors that will never be used as possible. To insure that only the state vectors we intend to write are written, stack base enumeration is used to implement these rules.

The first rule is to limit the max amount of time lapsed among any zone to equal that zone length plus a gap number that is fixed over all of the zones. The amount of time that the defender observes the shorter entry points will be more than the time the defender observes the longer entry points because the optimal surveillance pattern tries to make all entry points equally dangerous for the intruder. This rule takes advantage of the shorter entry points not having as much time lapse before they are checked and makes problems with higher variances among zone lengths much more efficient.

The second rule is there is exactly one zero in each state vector. The single zero exists because the defender is always observing a camera and only one camera at a time.

This condition is necessary for feasibility, but also serves to make the algorithm more efficient because there are fewer possible state vectors.

The third rule is that each number is unique in the state vector. If a number shows up more than once, the defender would have observed more than one camera at the same time at some point in the surveillance pattern. Since the problem does not allow the defender to observe multiple cameras at the same time, this condition is necessary to make the state vectors feasible.

To implement these rules in an algorithm, stack-based enumeration is used. This algorithm is based on network theory so the state vectors are equated to paths in the network. Stack based enumeration is used to enumerate all paths and their possible transitions and each path is listed only once. The algorithm does a depth first search to find a possible path. In this problem, the first path would be (0, 1, …, n). Then the algorithm back tracks one node from the path completion and searches for other ways to complete the path from this partial path. After all the paths are completed from the partial path one node removed from the complete path, the algorithm moves back another node and completes new paths from that point. This process is repeated until all paths are enumerated. The pseudo-code for this process is shown in Figure 5. This algorithm enumerates the paths that are used and prints them out as state vectors. The stack is the array si[], and the index *top* points to the top of the stack. Each entry on the stack represents a different entry point, and when *top*==*n*, we have a complete state vector defined. We then print it out (along with all the other relevant information, such as feasible transitions from that state), pop the stack, and move on to the next feasible state.

D. UPPER BOUND ON DETECTION PROBABILITY

Ideally, the defender should monitor entry point i every c_i time units in order to detect any intruder who attempts to penetrate through entry point i, where c_i is the travel time of entry point i. In order to equalize the detection probability on each entry point, intuitively the proportion of time spent on each entry point would be inversely proportional to the amount of time it takes to move through the entry point. If solved in this manner, no matter which entry point the intruder chooses they will have the same probability of being detected. The solution, however, is not always feasible because the solution may require the defender to monitor two entry points simultaneously. Instead, the solution of this method can serve as an upper bound for the detection probability.

Here is an example of the process to find the upper bound for probability of detection with the problem with three entry points of lengths (1, 2, 3). The least common multiple of these three entry points is 6. Then we divide the least common multiple by each entry point to find the amount of time that entry point is observed before the surveillance pattern repeats. This division gives us 6 time units for the first entry point, 3 time units for the second entry point, and 2 time units for the third entry point. Then we normalize the time spent on each entry point. When the time is normalized 6/11 of the time is spent observing the first entry point, 3/11 of the time is spent observing the second entry point, and 2/11 of the time is spent observing the third entry point. To find the probability of detection, take the proportion of time spent on the entry point and multiply it by the length of the entry point. In this problem, the solution is 6/11 for all three entry points. This detection probability is also the optimal detection probability produced by the linear program. One optimal surveillance pattern is to following the sequence (1, 2, 1, 3, 1, 2, 1, 3, 1, 2, 1) and to repeat the sequence indefinitely. This method for solving the upper bound for the probability of detection can be broken into these steps:

1. Find the least common multiple for all of the entry points.
2. Divide the least common multiple by the entry point length for each entry point. This step makes the time spent on each entry point inversely proportional to the entry point length.
3. Normalize the time spent on each entry point so when all the times are added up they are equal to one.
4. Multiply the entry point length by the normalized time spent on it to find the upper bound on the probability of detection.

The solution to this method can only be used as an upper bound because the discrete time surveillance pattern cannot be implemented efficiently in all problems. This difference can be seen when writing out the surveillance pattern. The surveillance pattern will be at least as long as the denominator in the proportion of time that the entry point is observed. The observations of the different entry points should be spaced in the surveillance pattern so that they do not observe an entry point more than would be needed for a probability of detection of one. If this efficiency can be achieved the upper bound is feasible and it is the optimal solution. If it cannot be achieved, the upper bound is not feasible due to the discrete surveillance pattern.

An example of when the upper bound is not feasible can be seen with the three entry point problem with lengths (2, 3, 6). The least common multiple is 6. The time observed for one surveillance cycle for the first entry point is 3, for the second entry point 2 and for the third entry point 1. The proportion of time spent on the entry points are 3/6, 2/6, and 1/6 respectively. The upper bound for this problem is therefore a probability of detection of one. This solution, however, is not feasible because the defender needs to monitor entry point 1 once every other time unit and entry point 2 once every 3 time units and it is impossible for these two patterns not to overlap. The optimal mixed strategy gives a probability of detection of 0.9231, and uses the following surveillance patterns: 41.67% of the time use the pattern (1, 1, 2, 3, 1, 2, 1), 8.33% of the time use (2, 1, 2, 1, 1, 2, 1, 2, 3, 1, 2, 1), and 50.00% of the time use (2, 1, 3, 1, 2, 1).

III. NUMERICAL ANALYSIS

In section A, we report numerical results for problems of three and four entry points. In section B, we conduct sensitivity and numerical analysis and discuss the patterns that emerge and the conditions in which the patterns hold true.

A. NUMERICAL DEMONSTRATIONS OF MODEL

This section presents numerical results in four tables. The first two tables present three entry point problems with two of the entry points varied. The next two tables present four entry point problems with two of the entry points varied. The notation for problems is (c_1, c_2, . . ., c_n). As stated in section II B, c_i is the travel time through entry point *i*. Therefore a three entry point problem with travel times of 2, 3 and 3 respectively, can be written as (2, 3, 3).

Table 1. The probability of detection matrix for the problem (1, *x*, *y*); if the cell has two values listed, these values are lower and upper bounds.

	y					
x	1	2	3	4	5	6
1	0.3333	0.4000	0.4286	0.4444	0.4545	0.4583, 0.4615
2	0.4000	0.5000	0.5455	0.5714	0.5882	0.6000
3	0.4286	0.5455	0.6000	0.6316	0.6522	0.6667
4	0.4444	0.5714	0.6316	0.6667	0.6896, 0.6897	0.7059
5	0.4545	0.5882	0.6522	0.6896, 0.6897	0.7143	0.7317
6	0.4583, 0.4615	0.6000	0.6667	0.7059	0.7317	0.7500

Table 2. The probability of detection matrix for the problem (2, *x*, *y*); if the cell has two values listed, these values are lower and upper bounds.

	y					
x	1	2	3	4	5	6
1	0.4000	0.5000	0.5455	0.5714	0.5882	0.6000
2	0.5000	0.6667	0.7500	0.8000	0.8333	0.8571
3	0.5455	0.7500	0.8571	0.8889, 0.9231	0.9091, 0.9677	0.9231, 1.0000
4	0.5714	0.8000	0.8889, 0.9231	1.0000	1.0000	1.0000
5	0.5882	0.8333	0.9091, 0.9677	1.0000	1.0000	1.0000
6	0.6000	0.8571	0.9231, 1.0000	1.0000	1.0000	1.0000

Table 3. The probability of detection matrix for the problem (2, 2, *x*, *y*); if the cell has two values listed, these values are lower and upper bounds.

	y					
x	1	2	3	4	5	6
1	0.3333	0.4000	0.4286	0.4444	0.4545	0.4583, 0.4615
2	0.4000	0.5000	0.5455	0.5714	0.5882	0.6000
3	0.4286	0.5455	0.6000	0.6316	0.6522	0.6667
4	0.4444	0.5714	0.6316	0.6667	0.6897	0.7059
5	0.4545	0.5882	0.6522	0.6897	0.7143	0.7317
6	0.4583, 0.4615	0.6000	0.6667	0.7059	0.7317	0.7500

Table 4. The probability of detection matrix for the problem (1, 3, *x*, *y*); if the cell has two values listed, these values are lower and upper bounds.

	y					
x	1	2	3	4	5	6
1	0.2963, 0.3000	0.3529	0.3750	0.3857, 0.3871	0.3896, 0.3947	0.3929, 0.4000
2	0.3529	0.4286	0.4615	0.4800	0.4918	0.5000
3	0.3750	0.4615	0.5000	0.5217	0.5357	0.5455
4	0.3857, 0.3871	0.4800	0.5217	0.5455	0.5607	0.5714
5	0.3896, 0.3947	0.4918	0.5357	0.5607	0.5769	0.5882
6	0.3929, 0.4000	0.5000	0.5455	0.5714	0.5882	0.6000

B. OPTIMAL SURVEILLANCE PATTERNS

Below we conduct a numerical analysis from the results presented in Tables 1, 2, 3 and 4. The range of data used to conduct numerical analysis is all possible entry point lengths from 1 to 6 with the number of entry points ranging from 2 to 4. The gap used in our sensitivity analysis is 5. This range is wide enough to pick out certain patterns and understand the results of the model.

In conducting sensitivity analysis with different entry point length, three patterns emerge consistently.

1. To increase the number of entry points and keep all of the path lengths the same, decreases the probability of detection.

2. If everything is the same except that one entry point's travel time is increased, the probability of detecting an intruder either stays the same or increases.

3. When there is an ability to increase the length of any entry point by one, it is most effective to increase the entry point with the shortest length.

The probability of detection decreases when all entry points remain the same length and a new entry point is added. For instance, a two entry point problem with entry point lengths (1, 2) has a probability of detection of 0.667, a three entry point problem with entry point lengths(1, 2, 2) has a probability of detection of 0.500, and a four entry point problem with entry point lengths (1, 2, 2, 2) has a probability of detection of 0.400. The intruder has all the same options as before with the addition of the new entry point, which if left unguarded guarantees undetected travel through the military installation's defenses. Conversely the defender has the same defensive obligations as before and must now spend time observing the new entry point.

The monotonic results of the model intuitively make sense because the model can always use the same algorithm it used in any similar problems with shorter paths for the same probability of detection if the probability of detection does not increase. This monotonicity is evident because all of the constraints are linear which guarantees an optimal probability of detection from the state vectors listed. Because we can not assign an unlimited gap, there may be state vectors with very large values that are not listed. In these cases, we have a suboptimal solution that can be detected as previously shown. However, due to the algorithm in which we list the path lengths and the state vectors are generated, all the same state vectors are listed for problems with the same number of entry points and longer entry point lengths. There is also the addition of the state vectors to account for the longer wait time for longer entry points which allows the probability of detection to be greater. An illustration of the monotonic results for the three entry point problem is illustrated in Tables 1 and 2.

Table 5. The difference in probability of detection for the two matrices (2, 2, *x*, *y*) and (1, 3, *x*, *y*); if the cell has two values listed, these values are lower and upper bounds.

	y					
x	1	2	3	4	5	6
1	0.0333, 0.0370	0.0471	0.0536	0.0573, 0.0587	0.0598, 0.0649	0.0583, 0.0655
2	0.0471	0.0714	0.0839	0.0914	0.0964	0.1000
3	0.0536	0.0839	0.1000	0.1098	0.1165	0.1212
4	0.0573, 0.0587	0.0914	0.1098	0.1212	0.1289	0.1345
5	0.0598, 0.0649	0.0964	0.1165	0.1289	0.1374	0.1435
6	0.0583, 0.0686	0.1000	0.1212	0.1345	0.1435	0.1500

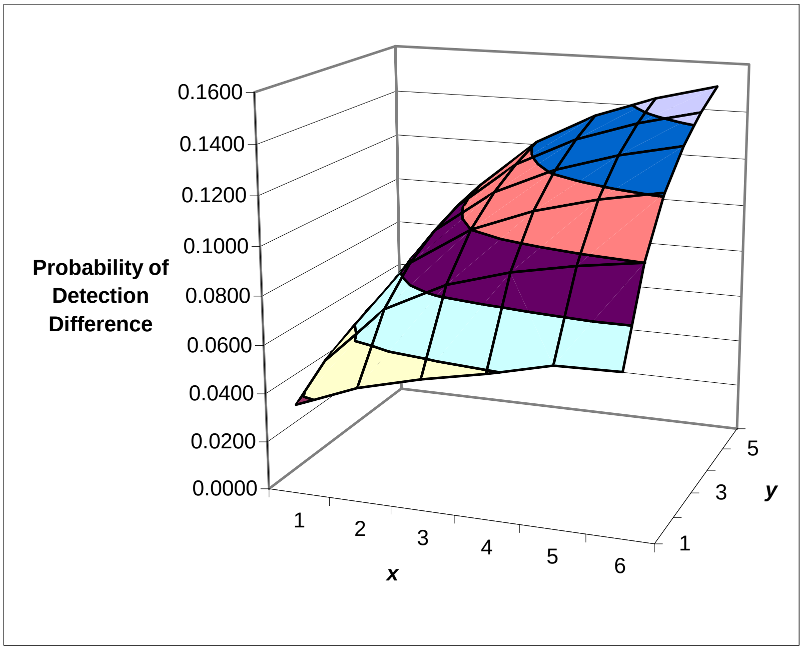

Figure 6: The difference in the probability of detection between the problems (2, 2, *x*, *y*) and (1, 3, *x*, *y*).

It is observed in Table 5 that the most efficient way to increase probability of detection is to increase the time of the quickest entry point. This observation is demonstrated with the use of four entry point models as an example. For the example, we take the matrix of probabilities of detection for the installation (2, 2, *x*, *y*), as shown in Table 3, and subtract from it the matrix (1, 3, *x*, *y*), as shown in Table 4. The *x* and *y* values are the entry point lengths that are varied from one to six in the matrices. The resulting matrix is shown in Table 5. Table 5 demonstrates that it is more efficient to spread the defensive measures when protecting an installation with surveillance cameras that are cycled through as in this model.

In Table 5, another observation is made that as the variable path lengths of *x* and *y* grow larger there is a greater difference in probability of detection in the model. This greater difference in probability of detection is visually displayed for our example in Figure 6. The mathematical explanation is as *x* and *y* approach infinity, we no longer need to monitor these entry points because it takes an infinite amount of time to make it across. Thus, we can reduce the two problems to (2, 2) and (1, 3) respectively. The problem (2, 2, *x*, *y*) therefore converges to a probability of detection of one as *x* and *y* reach infinity. The problem (1, 3, *x*, *y*) converges to a probability of detection of 0.75. This difference in what the problems converge to is more apparent as *x* and *y* get longer and have less of an impact on the problem.

IV. CONCLUSIONS AND RECCOMMENDATIONS

In Section A, we summarize our findings. In Section B, we discuss other applications of the model and further research directions.

A. FINDINGS

The numerical analysis gives three methods for improving surveillance in a military installation that correspond to our intuition:

1. Decrease the number of entry points.

2. Increase the travel time it takes to cross through an entry point.

3. Spend resources on improving the entry point that has the lowest travel time first.

It is easier to operate surveillance in a military installation when everyone who enters is funneled into a small number of entry points. A decrease in the number of entry points makes it more difficult to penetrate the military installation’s defenses. The methods to cut off entry points depends on the set up of the military installation and which kind of intruders are intended to be kept out. The use of razor wire on a wall that is low enough to climb over or a secure lock on a door may be enough to make an entry point unusable. Some entry points to the military installations need to be kept open for the military installation’s intended users.

The next best method to increase a military installation’s defense besides completely removing an entry point is to increase the travel time it takes to cross through an entry point. Increasing travel time can be done by extending the length of monitored hallways and having them double back, installation of walls or doors, locks and obstacles, etc. The cost of increased travel time over an entry point is an increase in travel time for the entry point’s intended users.

The most efficient way to use resources to increase the defense of a military installation is to increase the travel times of the most accessible entry points. The most efficiently protected installations have equal travel times for all of their entry points.

B. FURTHER APPLICATIONS AND RESEARCH

Although we explain the research problem via an example involving cameras around a military installation, the research effort applies to some other situations. Another application of the model is for a reconnaissance satellite to monitor the traffic around several possible meeting locations of a terrorist group in order to locate its leaders. Vehicles are easier to detect when the satellite has good visibility from space. This good visibility corresponds to longer travel times. The terrorist chooses a location to meet, and the satellite uses a surveillance pattern to maximize the probability of identifying that meeting location.

With some slight modifications our model is also applicable to a border patrol problem. The travel time reflects the difficulty of crossing the terrain over each entry point. In this scenario, however, an extra constraint must be added to account for the physical movement of the defender. That is, in one time unit the defender can only stay at where he is or move to an adjacent entry point. The patrolling unit can be a vehicle or a UAV.

There are a few possible future research directions. One immediate extension is to allow the probability of detection to be less than 1, such that it is possible for the defender to miss an intruder. Another extension is to let the time needed to make detection depend on the entry point, because it may take a longer time to scan a longer entry point. In addition, one can consider a continuous-time model if a discrete-time model cannot produce a satisfactory approximation, or allow multiple surveillance agents to coordinate their surveillance patterns. These potential extensions could improve the model's applicability as well as give further insights into effective surveillance of military installations.

LIST OF REFERENCES

[1] G. Brown, M. Carlyle, J. Salmerón, K. Wood, Defending critical infrastructure. *Interfaces.* Vol36, No. 6, November–December 2006, pp. 530–544

[2] D. Cfir, M. Kress, K. Lin, R. Szechtman, Models of sensor operations for border surveillance. October 3, 2006

[3] B. O. Koopman. *Search and Screening*. Office of the Chief of Naval Operations, Operations Evaluation Group Report 56, 1946.

[4] H. Pulat, A two-sided optimization of border patrol interdiction. Master's thesis, Operations Research Department, Naval Postgraduate School, Monterey, CA. 2005

[5] A. R. Washburn, On patrolling a channel, *Naval Research Logistic Quarterly*, 29:609 615, 1982.

[6] A. R. Washburn. *Search and Detection*, Institute for Operations Research and the Management Sciences, 4th edition, 2002.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia

2. Dudley Knox Library
Naval Postgraduate School
Monterey, California

3. Kyle Y. Lin
Naval Postgraduate School
Monterey, California

4. W. Matthew Carlyle
Naval Postgraduate School
Monterey, California

5. Trevor D. McLemore
Naval Postgraduate School
Monterey, California