2014-01-01

A Visualization and Simulation Tool That Will Generate Effective Patrolling Strategies to Protect the U.S. Borders From Illegal Intrusion Using Game Theoretic Methods and Models

Eric Gutierrez
University of Texas at El Paso, ejgutierrez@miners.utep.edu

A VISUALIZATION AND SIMULATION TOOL THAT WILL GENERATE EFFECTIVE PATROLLING STRATEGIES TO PROTECT THE U.S. BORDERS FROM ILLEGAL INTRUSION USING GAME THEORETIC METHODS AND MODELS

ERIC GUTIERREZ

Intelligence and National Security Studies Program

APPROVED:

____
Larry Valero, Ph.D., Chair

____
Christopher Kiekintveld, Ph.D. Co-Chair

____
Sarah McGuire, Ph.D.

____
Bess Sirmon-Taylor, Ph.D.
Interim Dean of the Graduate School

A VISUALIZATION AND SIMULATION TOOL THAT WILL GENERATE EFFECTIVE PATROLLING STRATEGIES TO PROTECT THE U.S. BORDERS FROM ILLEGAL INTRUSION USING GAME THEORETIC METHODS AND MODELS

By

Eric Gutierrez

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at El Paso

in Partial Fulfillment

of the Requirements

for the Degree of

Master of Science

Intelligence and National Security Studies Program

THE UNIVERSITY OF TEXAS AT EL PASO

May 2014

Acknowledgements

I would like to thank Dr. Christopher Kiekintveld for his guidance in my research interests and his help with developing GAMMASys. I also appreciate the guidance of Dr. Larry Valero for allowing me to use this topic as my thesis. His mentorship in the field of homeland security and intelligence is also much appreciated. I would like to thank Dr. Sara McGuire for her insight during my research for my thesis topic. I would like to thank Jonathan Juett who initially helped develop GAMMASys and the first version of the genetic algorithm, in their earliest stages. I would like to thank my brother, Marcus Gutierrez, for assisting me in the debugging process of the revamped genetic algorithm in GAMMASys. I would also like to thank the IASRL researchers for the brainstorming and discussion sessions we had weekly, which gave me ideas for this research project. Finally, I would like to thank the UTEP Computer Science Department, for allowing me to conduct research in their laboratories and the Intelligence and National Security Studies Program to allow me to combine my interests of border security and game theory to assemble an interesting research topic that could be used for my thesis.

Abstract

In the recent decade, the United States Border Patrol has increased the presence of border security officers at points of entry along the U.S.- Mexico border. In response to this increased presence of border security officials at ports of entries, illegal intruders have rerouted their intrusions in between ports of entry, in harsher terrain areas of the U.S. (e.g., the Arizona desert). The U.S. Border Patrol could benefit from a tool that plans effective border patrolling routes, and helps border security officials to make risk-based decisions for resource allocation. We are currently developing a tool called Genetic Algorithm for a Map-based Multi-Agent System, or GAMMASys, which contains tools, that allows for the creation of effective border patrolling strategies using visualization, simulation, randomization, and automation. These features allow border security officials to observe the relationship between adversaries (i.e., patrollers and intruders) using a visualized terrain tool. This allows for multiple patrolling strategies to be simulated through games using a defined adversarial game theoretic model; allows for intelligent and unpredictable decisions to be made by adversaries using randomization; and allows for effective patrolling strategies to be generated automatically using crossover and mutation techniques with the genetic algorithm. The contribution of this research is to provide a sophisticated tool that can apply all these aspects to generate effective patrolling strategies for the real-world border zones, and answer questions about the cause and effect relationship of intelligent decisions made by adversaries in the border security domain. These questions regard: altering patroller/intruder intelligence levels in the game theoretic model, efficient parameter combinations for high solution quality in lower amounts of time, and resource allocation. The ultimate intention is to construct scenarios to evaluate the relationships between intelligent adversaries and the effects of their decisions.

Table of Contents

Chapter 1: Introduction

Defending the borders of the United States is a primary national security focus of the United States government. In order to defend the borders, the United States Border Patrol (USBP) allocates resources near regions that lie along the Northern, Southern, and coastal borders of the U.S., between ports of entry (POEs).[1] The USBP, which operates under the Department of Homeland Security (DHS) and Customs and Border Protection (CBP), is responsible for the detection and interdiction of illegal entrants. These entrants can be terrorists or criminal operators that can be involved in illegal narcotic-, human-, and arms- trafficking. Illegal intruders can also be people with no terrorist or criminal affiliations that are unauthorized to enter the U.S.

After fiscal year 2000, the USBP's budget increased from the previous decade (the budget of USBP in FY2000 was $1.06B), as a result the amount of personnel in the border security ranks tripled (By 2009, USBP had 17,499 agents in its ranks), according to Nunez-Neto, a U.S. Senate committee member on Homeland Security and Government Affairs (Nunez-Neto, 2008). The budget and personnel increase was due to congressional concerns about an increase in illegal immigration and terrorism along the U.S.-Mexico border. Although the resources allocated to this agency increased, the strategy for securing the border remained the same, as it had in the early 1990's.[2] This strategy was a control-based resource allocation strategy, where highly populated areas along the U.S. borders were prioritized with personnel power. The strategy was termed "prevention through deterrence," according to Nunez-Neto, where the USBP

---

[1] Aguirre, Oswaldo; Lopez, Nicholas; Gutierrez, Eric; Toboada, Heidi; Espiritu, Jose; Kiekintveld, Christopher. "Towards the Integration of Multi-Attribute Optimization and Game Theory for Border Security Patrolling Strategies." *Applied Adversarial Reasoning and Risk Modeling: Papers from the 2011 AAAI Workshop* (2011).

[2] "CBP Border Security Spotlight." CBP.gov - home page. http://www.cbp.gov/xp/cgov/border_security/ (accessed July 5, 2013).

deployed a combination of personnel and technologies near different POEs along the U.S. borders. By using a deterrence strategy at POEs, an impression of a militarized border was given to potential intruders, and part of the motivation of this strategy was to prevent future intrusion attempts by intimidating the intruder.[3]

This preventive strategy has been a major factor in rerouting illegal entrants into more hazardous terrain. Since POEs had become difficult to cross due to a militarized border, intruders were pushed from entering through urban locations into harsh and dangerous territory, such as the Arizona desert. Many of these entrants were ill-prepared to travel in such terrain, and as a result, many died while trying to cross the border into the United States.[3]

This adaptive response enacted by illegal intruders brings up two important problems that must be dealt with by the government. First, it is apparent that illegal intruders are attempting to cross into U.S. territory between POEs, which have fewer border security resources compared to the urban entry points.[4] Second, an increase in illegal entry through harsh terrain has resulted in deaths along the southern U.S. border.[5] An issue with border security that arises regardless of the border strategy that is being used relates to the fact that the USBP does not have the capability of perfect hindsight. As yet, there is no way of accurately determining whether the strategy being used is highly successful in preventing illegal entry or in increasing the rate of interdiction.[6,7] This is due to the fact that it is not yet possible to tell how many intruders have

---

[3] Nunez-Neto, Blas. "Border Security: The Role of the U.S. Border Patrol." *Cornell University ILR School* (2008).

[4] "Border Patrol National Strategy 2012-2016."(2012). http://www.krgv.com/files/2012-2016_BP_Strategy.pdf (accessed March 22, 2013).

[5] Cornelius, Wayne. "Evaluating Enhanced US Border Enforcement." *Migration Information Source: Fresh Thought, Authoritative, Global Reach* (2004). (Accessed March 22, 2013).

[6] Predd, Joel, Henry Willis, Claude Setodji, and Chuck Stelzner. "Using Pattern Analysis and Systematic Randomness to Allocate U.S. Border Security Resources."*RAND: Homeland Security and Defense Center* (2012).

been captured, and how many have eluded interdiction. Technology such as unmanned aerial vehicles (UAVs), manned patrolling aircraft, sensors, and cameras, cannot determine if all intruders have been detected. One problem is that technology can fail. For instance, a sensor could alert officials when an animal activates it, or it could fail to detect an intruder that entered the sensor's proximity. Even if an increase of apprehension occurs, this may not be the result of a completely effective strategy, but because the amount of intrusion attempts has increased for some unknown reason, or because more resources are available for interdiction.[4]

Ultimately, one major issue of border security is that unauthorized intruders can adapt to an agency's deterrence and interdiction strategies. These strategies need to remain resilient towards a change in the intruder's strategy to cross the border, and changes in the amount of resources available. Due resource and budget limitations, the USBP cannot simply place agents evenly along the U.S. borders along with high priced technologies to deter intruders from entering U.S. territory. Therefore, a solution must be sought that takes into account the fact that resources for border defense may not always be abundant and the location where an intruder may cross into the country between POEs may not be predictable.

Game theory is a mathematical tool that can be applied to study complex multi-agent decision problems, including border security. This tool can be used to conceptualize a problem and to form an adversarial model. In such a model, decision-making between adversaries can occur, and can be observed by researchers to better understand the relationship between adversaries. For instance, an unauthorized intruder attempting to cross the U.S. border can make several decisions that a border security officer cannot directly observe. In game theory,

---

[7] Alden, Edward. "Immigration and Border Control." *CATO Journal* 32, no. 1 (Winter 2012): 107-124. *Academic Search Complete*, EBSCO*host* (accessed March 24, 2013).

adversaries can make many decisions and can use strategies, which are plans of actions.[8] This is similar to what illegal entrants do in real-world scenarios when they attempt border intrusions. For example, the entrant can make many decisions during a crossing attempt such as: deciding which section of the border to cross, the pace at which they cross the border, the time of day they choose to cross, how many people they choose to cross with, or whether they abort the crossing before they reach U.S. territory. With game theory, these kinds of variables can be integrated in a stylized game, and an observable space (or a grid which allows for adversarial action to take place) can be constructed, so that an observer (a border security official or the user of a computational model) can evaluate the relationship between the border security official and the illegal entrant.[9]

This thesis describes a visualization and simulation tool that produces patrol routes for USBP officials to use in order to effectively cover large amounts of territory with a small amount of patrolling personnel. The tool is based on a game theory model that uses a genetic algorithm to generate patrolling strategies that use randomization to prevent adversaries from exploiting weaknesses in a predictable strategy.[5] Our goal is for the USBP to be able to use these results in real-world patrolling problems.

---

[8] Davis, Morton D. *Game Theory: a Nontechnical Introduction*. Courier Dover Publications, 1983.

[9] Camerer, Colin. *Behavioral Game Theory: Experiments in Strategic Interaction*. New York, N.Y.: Russell Sage Foundation ; 2003.

Chapter 2: Problem

2.1. Relationship to Border Patrol Strategic Plan

The USBP National Strategy for 2012 to 2016 set goals for the agency to accomplish with regards to improving U.S. border security. In keeping in the ways in which criminal organizations have adapted to the deterrence strategies imposed by border security officials, the USBP is attempting to adapt to the migration of irregular intruders through harsher terrain to cross U.S. borders.[3] This objective is laid out in the first goal of the USBP, where the agency states that a fundamental mission of the Border Patrol is to secure U.S. borders between POEs. The USBP states that it wishes to take a risk-based and outcome-focused approached to securing the border by utilizing new information, integration of strategy, and a rapid response plan when detection of an intruder occurs. These are the objectives of the USBP, so the goals of this research can be molded by keeping the goals of the Border Patrol in mind.[3] Risk-based approaches are important because border zones face a lack of resources (e.g.: number of patrols, proper funds, etc.) and risk-based actions taken by USBP officers in these border zones can maximize efficiency of the resources in these areas of responsibility.

We discuss in more detail the USBP objectives that are related to this research. The three objectives of the first goal are: preventing intrusion between POEs using intelligence-driven operations along with strategic planning; effectively managing risk through the expansion of sophisticated tactics, techniques, and procedures; and disrupting transnational criminal organizations (TCOs) by targeting the highest priority threats and reducing smuggling across the border. These objectives conceptualize prevention and intelligent, risk-based resource allocation. By using intelligence-based communication, detections of intruders can be shared with the USBP and patrols can be sent to interdict the detected intruders.[3] By analyzing data

about detected intrusion attempts, USBP officers can keep track of the locations in between POEs that are frequently crossed by intruders.[5] From this information, detected intrusion data can be used in combination with GAMMASys to generate effective patrolling strategies to defend these locations from intrusion. Border security officers will map the location on the visualization tool, apply the amount of patrollers allotted to that border zone, place the starting position of the patrollers and the types of vehicles they will be given, apply the predicted starting locations of the intruders, and run the genetic algorithm to generate patrol paths to cover that area. It should be noted that GAMMASys does not currently contain capabilities such as intelligence communication or data collection on intrusion attempts, but the previous scenario is an example of how our simulation system can be applied to assist in meeting USBP goals.

The second goal has four objectives, three of which are relevant to this research. The objectives of this goal are: training and support of USBP personnel; improving reporting, planning, and introduction of tools to improve interdiction outcomes; and enhancing the efficiency of planning, resource allocation, and acquisition processes.[3] This research is primarily geared towards the last objective. This tool satisfies the requirement in objective three, by providing effective patrolling strategies that can speed up the planning process and can help increase interdiction rates.

The goal of this research is to assist in satisfying the most recent mission of the USBP, but in the sense of generating effective patrolling strategies that can be used in real-world situations and environments that will increase the interdiction rate with limited resources in border zones. This research will focus on creating improved strategies by making both the patrolling and intruding agents in the simulation system more competitive, by improving the decision making policies of these agents and making them more adversarial. Improvement in

decision-making will be done by ***reward***. This means that when an agent achieves their objective, they will receive a positive reward value, which allows the genetic algorithm to weigh the effectiveness of adversarial decisions and improve future decisions.

The patrolling strategies for real-world application will be created through ***automation*** (after the user sets the system parameters, simulations and the genetic algorithm will run automatically to generate results), which will increase the speed at which these patrol routes can be constructed. The patrol routes will also be ***randomized***, in order to be more unpredictable and avoid exploitation of these patrolling strategies. The routes will be produced by a genetic algorithm that has been programmed for this research, in order to increase the interdiction rates of these strategies through ***simulation***. Stronger strategies, or patrol routes more/ increased simulation captures, will be systematically combined with other strong strategies in order to search for more effective strategies. This will all be done using a terrain-based simulation tool called GAMMASys (Genetic Algorithm for a Map-based Multi-Agent System).[10] This simulation tool allows the operator to ***visualize*** on going games, in order to observe the decisions made by both the patrolling and intruding agents.

2.2. Research Contributions

The overall contribution of this research study is to deliver a computational model that is based off of adversarial interaction. We have implemented an algorithm that can be used to compare patrolling strategies relevant to the border security domain. This system uses randomization to improve performance against intelligent adversaries, and allows for the allocation of resources in a very large action space of possible patrolling/intrusion strategies.

[10] Gutierrez, Eric, Jonathan Juett, and Christopher Kiekintveld. "Generating Effective Patrol Strategies to Enhance U.S. Border Security." *Journal of Strategic Security* 6, no. 3 Suppl. (2013): 152-159.

The first contribution is to develop a game model. This model is game theoretic and based on the border security domain. It captures adversarial interactions on a realistic landscape with terrain features. This computational model will allow for series of simulations to be executed on a system that produces patrol paths which are the results of many adversarial interactions.

The second contribution is developing a scalable algorithm for solving large adversarial patrolling problems. I developed and implemented a genetic algorithm which has been integrated into the simulation tool. This algorithm is scalable, which means that its parameters can be altered to allow the accuracy of the results, and the speed at which the results are generated, to fit the needs of the border security officer using the simulation tool. The algorithm is designed to produce effective patrolling strategies based on the characteristics of the computational model.

The third contribution is the evaluation of varying intelligence levels in adversaries. Within the model two types of adversaries are competing against one another to achieve their objectives: the intruders and the patrollers. Intelligence levels (competitive capabilities of the agents) of adversaries affect their behavior during simulation. As such, by evaluating these intelligence levels, strong patroller strategies can be used to mitigate the effectiveness of intruder strategies, which is useful in a setting with a similar terrain layout. Patroller and intruder behavior and strategies are evaluated based on their use of randomness, for unpredictability, to defeat their opponents in a non-cooperative game. With intelligence levels, the relationship between patrollers and intruders are evaluated by employing different combinations of vehicles types, apply random strategies to the patrollers and intruders, employ caution-based strategies to intruders, apply different movement strategies for patrollers (e.g., Markov strategies), and

introducing multiple source and target locations for the intruders. In a sense, we are applying different levels of sophistication to patroller and intruder strategies. We evaluate how effective these strategies are, and develop scenarios to evaluate these strategies.

The fourth contribution is a tool for resource allocation. We also conduct experiments using the simulation tool to evaluate the effectiveness of resource combinations in specified zones. By using the simulation tool, border security officials will be able to optimize their use of resources and personnel to improve the coverage of their border zones. This means that border security officials will be able to save funds by using the correct amount of resources without using too many, or too few personnel to patrol a zone, and by allocating the right travel means for the patrollers.

2.3. Research Questions

What effects does facing a more intelligent intruder have on the effectiveness of a patrolling strategy? The computational model of GAMMASys is programmed to represent an adversarial game theoretic model. It is important to observe the effects of actions made by adversarial agents, because these actions affect the decisions of all agents within the game model. If an agent is more competitive, or formidable, the decisions of that agent's adversary must be effective and trump these actions in order to accomplish their own game objectives. If an agent is not making intelligent or effective decisions, or if an agent is not basing their decisions off an adversary's actions, then a game model will not produce effective results.

Can more intelligent patrolling strategies increase capture rates against intelligent intruders? If an adversary is conducting an intrusion route between U.S. ports of entry, a patrolling agent must be able to intercept the intruding agent before they reach their goal.

However, if the intruding agent is using a more evasive strategy and the patroller is using a less intelligent and mobile strategy, the intruding agent might be successful in their attack route more often. Patrollers operating within the system must have the capability of being successful against formidable intrusion strategies. Tests are constructed that allow for the observation of intelligent adversarial interaction.

Can the computational model be used to evaluate and optimize resource allocations for a given zone? It is important for analysts to be able to use the system for resource allocation, because analysts can observe the best combination of patroller types to employ, and the number of patrollers to use in a simulated zone. If an analyst is able to arrive at this information, then a border agency can effectively cover a border zone without over spending for effective coverage. This also prevents an adversary from gaining at a tactical advantage because a zone lacks coverage from enough patrollers.

Can the genetic algorithm find effective patrolling strategies fast enough for practical use? In real-world scenarios, analysts must act on information they receive to provide timely intelligence. If information is received on possible border intrusions, effective and time efficient patrolling strategies must be constructed. In order to create effective and efficient strategies, certain variables must be altered in the system to allow for timely solutions.

2.4. Other Benefits of the Research

This research can also contribute to the USBP by exploring new possibilities in both manned and automated patrolling. Automated patrolling can refer to the patrolling of autonomous entities such as land-based robots or even air drones such as UAVs. Similar research has been done on automated patrolling, and these experiments are reviewed in the related works section.[11] [12] This research will also benefit border security in the form of planning, by generating effective strategies using automation, meaning that many patrol routes can be stored for future use. The intelligence community can also make use of this simulation tool, because it can be used in the intelligence cycle. GAMMASys simulation can be used within the "Analysis" phase of the Intelligence Cycle.[10] The field of game theory can also benefit from this research by running a high volume of simulations and by observing the relationship between the two types of agents (attacker/intruder and defender/patroller). These simulations will be based on non-cooperative game theory and adversarial game theory. This means that the games will ensure that the agents make their decisions independently, and try to achieve their own objectives while forcing the other agents to fail at their objectives.[13] [14] In this case, the objectives are that intruder cross the border, without being detected, and that the patroller captures the intruder. Through testing these simulations, and analyzing the results, the findings of this

[11] Marino, Alessandro, Lynne Parker, Gianluca Antonelli, and Fabrizio Caccavale. "Behavioral Control for Multi-Robot Perimeter Patrol: A Finite State Automata Approach." (2009): ieeexplore.ieee.org (accessed March 18, 2013).

[12] Basilico, Nicola, Nicola Gatti, Sofia Ceppi, and Francesco Amigoni. "Extending Algorithms for Mobile Robot Patrolling in the Presence of Adversaries to More Realistic Settings." *Proceeding WI-IAT '09 Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology* 02 (2009): 557-564.

[13] Liu, Wei, and Sanjay Chawla. "A Game Theoretical Model for Adversarial Learning." In *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*, pp. 25-30. IEEE, 2009.

[14] Bonanno, Giacomo. *Non-Cooperative Game Theory*. No. 86. 2008.

research will benefit the communities within these subfields of game theory, and allow for research to build upon this study.

It is difficult for even a highly sophisticated simulation system to account for the amount of detail and complexities of the terrain found in the real world; thus, it is possible that the strategies that are created by GAMMASys may not be practical for real-world patrolling. Even if this were the case, the USBP, law enforcement, or military entities that wish to use generated patrolling routes can alter these patrol routes to fit their needs. These generated routes are developed with the consideration of limited resources, such as small amounts of personnel. The USBP will not have to change their field tactics at all, because techniques such as sign-cutting, a form of path tracking, can be used while travelling these routes.[15] One risk in using this simulation is that of spent time (running long algorithms), but GAMMASys can be fully automated and can run in the background of a server. This feature saves the operator the trouble of checking in on the algorithm and running it multiple times to get better results.

2.5. Definition of the Model

The game model that we have constructed contains ***players*** (***agents***) that compete against one another to achieve their own ***objectives*** (***payoffs***). In this case, the agents are ***intruders*** and ***patrollers,*** also called ***attackers*** and ***defenders***, respectively labeled on Figure 1. The intruders, or individuals attempting to illegally cross into U.S. territory, while the patrollers represent U.S. border security personnel. The objectives of these agents are as follows: the intruders must reach their chosen target node, or target location, from their node (starting location) without being

[15] North American Publishing Co. "US Border Patrol 50th Anniversary, 1924-1974." *State Police Officers Journal* 16, no. 37 (1974): 50-56.

captured by a patroller (an ***intrusion***); the patrollers must capture as many intruders as possible before an intruder reaches their target location (successful ***interdiction***). When an agent achieves their objective, the scenario is halted.

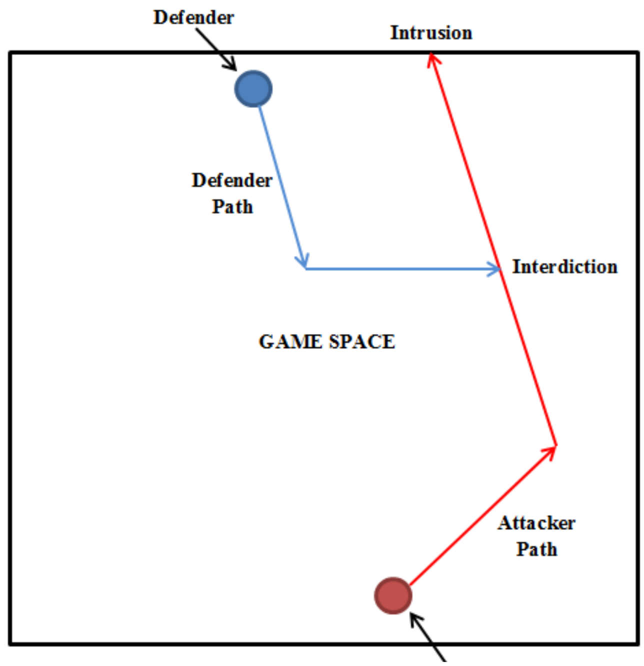

Figure 1: Shows the objectives of the intruder and patroller (no nodes or grid present). Attacker is an Intruder, Defender is a Patroller.

In this model, a capture occurs when a patroller is in the same location, or ***node***, as the intruder at any given ***time-step***, which is a fraction of a decision unit in the scenario. A node is characterized as a hexagon. When an agent arrives at a node, he has the option of travelling in the six facial directions of the hexagon (north, northeast, southeast, south, southwest, northwest); while the agents have the ability to stay in that node for a certain period of time. When an agent does travel, he must arrive at an adjacent node.

Both agents have different means of travel (e.g., cars, ATVs, trucks, boats, person on foot),), all of which affect their travel times. When an agent travels through nodes, his movement time is affected by the terrain type of the node he is traveling to. Each mode of transportation, or vehicle, of the agent has different movement speeds based on the terrain though which the vehicle is traversing.

2.6. The Visualization/Simulation Tool

GAMMASys is the visualization and simulation tool that we developed in this research project. Figure 2 below is a snapshot of GAMMASys, which simulates border security games. It contains a grid for visualization, options to alter the game space (i.e., edit the terrain on the grid), and a method to run a genetic algorithm which is fully automated, in order to simulate a large number of games and produce effective patrolling strategies for real-world applications.

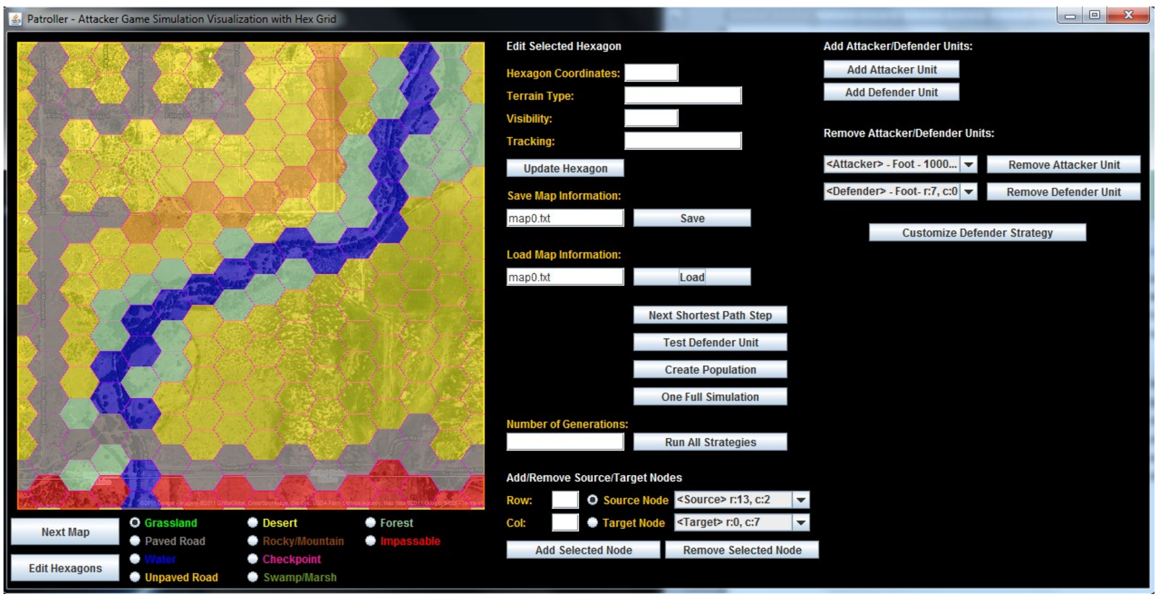

Figure 1: GAMMASys Tool depicting a 16x16 grid with terrain representing an outlay of a portion of Colorado National Monument, Satellite Image from Google Maps.[10]

The visualization shown in Figure 2 depicts a 16x16 hexagonal grid that overlays a satellite image from Google Maps. This grid is an interactive component of the system that allows the user to overlay each node, or hexagon, on the hexagonal grid. Each hexagon that represents a terrain type will be represented by a different color. Each terrain type affects the travel speed of an agent depending on their travel type. The system allows for these terrain types:

- **Grassland** – Green
- **Paved Road** – Grey
- **Water** – Blue
- **Unpaved Road** – Orange
- **Desert** – Yellow
- **Rocky/Mountainous** – Brown
- **Checkpoint** – Pink
- **Swamp/Marsh** – Dark Green
- **Forest** – Pale Green
- **Impassible** – Red

The system allows the user (e.g., border security official) to edit each hexagon, save a created map, load an existing map, run a genetic algorithm with input parameters (e.g., such as the number of test cases to run, the number of patrolling strategies to create, etc.), create a new set of randomized patrolling strategies known as a population, run a single simulation of a synthesized game that can be visualized, add or remove source and target nodes, and add or remove agents from the game. With these tools the border security official can customize the game space to represent to the real-world situation at hand, which can allow the production of more accurate patrolling solutions.

The adversaries in the simulation are the patroller and the intruder agents. Currently, a simulation can have multiple patrolling agents, but only one intruding agent in order for it to run correctly. The system has the capability to allow the use of multiple intruding agents, but scenarios containing multiple intruders have not been thoroughly tested. The patrolling agent travels with the use of randomly generated probabilities, where each face of the hexagon contains a different probability. By using probabilities, this allows for one side of the hexagon to be randomly chosen based on the probability distribution, and the patroller will travel in that given direction.[10] This approach is used for every hexagon on the grid, and is known as a

Markov Strategy.[16] The Markov strategy represents a ***patroller strategy*** or ***defender strategy.*** These patrolling strategies are restricted, which means that the optimal solution for patrolling may not be representable within the strategy space. These strategies, however, are highly efficient from a computational perspective. The intruder uses an algorithm called Dijkstra's Shortest Path algorithm. It is based on the travel weight of each hexagon's terrain type, and the distance from the intruder's ***source node***, or starting location, on the intruder's ***target node***, or their preferred ending location. The intruder can have multiple source and target nodes, but based on the distance between the two sets of nodes, the intruder chooses the optimal routes based off distance. The intruder also uses ***caution***, which determines the desired path of the intruder based upon probability of capture, using the probability that the patroller will visit certain nodes in their patrolling strategy. Intruder caution allows this agent to avoid areas that the patroller frequently patrols. The game ends when either agent achieves their objective. The patroller's objective is to capture the intruder by appearing in the same node as the intruder, and the intruder's objective is to reach its target node without being captured by the patroller.

The system uses a ***genetic algorithm***, which is automated to increase the number of simulations that can be executed. It only needs a human operator to execute the genetic algorithm initially. The genetic algorithm evaluates the effectiveness of the patroller strategies with a fitness score, which is measured by the percentage of capture. Once all strategies in the population have been simulated numerous times, the elite strategies (or strategies with the highest fitness score) are then altered using ***mutation***, ***crossover***, and ***averaged strategies***. The elite strategies maintain their place in the population, and the rest of the strategies in the population are recycled and replaced with modified strategies which are offspring of the elite

[16] Geyer, Charles J. "Practical Markov Chain Monte Carlo." *Statistical Science* 7, no. 4 (1992): 473-483.

strategies. Mutated strategies are copies of non-elite strategies that are re-normalized based on elite parent strategy distributions. Crossover strategies are distributions of hexagons from one of two parent strategies for each hexagon. For instance, each hexagon in the grid will have a 50% chance to inherit either parent's probabilities of that given hexagon. Averaged strategies are created by altering previous strategies, and re-normalizing the strategy distribution based on new random probabilities. This process is done for however many generations the user has specified. The average number of captures, the best capture percentages, the lowest elite capture percentages, and the median capture percentages are recorded. This process is done for numerous trials, all of which are specified by the user. By the end of this process, the user can evaluate the effectiveness of the generated strategies. This process is loosely motivated by biological evolution, and is an effective heuristic approach for finding good strategies quickly in a very large space.[9]

2.7. Genetic Algorithm

This section briefly explains the process by which an effective strategy is produced by the GAMMASys genetic algorithm. Let n represent the number of generations to run, m is the number of mutations to be executed, k is the number of simulations per generation, and l is the number of crossover strategies to be used.

1. Map is imported.
2. Terrain types are placed on hexagonal grid.
3. Intruder source and target nodes are specified.
4. Load a patroller strategy from a patroller population.
5. Monte Carlo Simulation is executed (intrusion attempt simulation).

6. Intruder chooses the path with the least travel cost.
7. Patroller uses the loaded strategy to attempt to capture the intruder (Jump to step 4 for the *n-1 (n minus one)* times, jump to step 8 on the n^{th} time).
8. The elite strategies are retained, the non-elites are altered.
9. Add *m* mutation, l crossover strategies, and k – n – l average strategies (Jump to step 4 until finished).

This multi-step process does not include the human interaction between the border security official and the system. Steps 1 and 2 can be automated (if the terrain map already exists), or the user can do this manually. Note that the terrain maps cannot be completed without a human user. Step 3 can also be specified by a user, or automated. Step 5 refers to the simulation that is run between the two adversaries as a practice run for the intruder to observe a sample patroller strategy. Step 6 refers to the intruders selecting their intrusion routes based on chosen source and target nodes. This process can be randomized to allow for a less predictable intruder strategy. Step 7 refers to the patroller choosing their strategy out of the patroller strategy population, and repeats steps 4-7 until the specified simulation count limit has been reached (this is specified by the user as “thoroughness”). From here, the real simulation between adversaries occurs. Step 8 refers to the preservation of elite strategies. The top percentage of patroller strategies will be kept, and the lower percentages will be discarded (these percentages are specified by the user). Step 9 refers to the empty portion of the population, which originally included the discarded or altered patroller strategies. This portion of the population is replaced by new strategies that are created by mutating the probabilities of the elite strategies. The crossover strategies determine which probabilities will be used in each referenced hexagonal node of the new strategy (travel probability distributions), and the mutation strategies determine

which strategies will be parents of the new strategy, and use the parents' probability distribution for alteration of the strategy. This process does not activate the visual simulation because this would be computationally costly, but when the algorithm is running, the results of each trial can be observed in the server prompt, and the results are recorded in a log file to be used for analysis and graphing.

2.8. Reasoning

GAMMASys is based on an adversarial game theoretic model. It uses a genetic algorithm to explore a multitude of decisions that can be made by patrolling agents to give them an advantage over their adversary, while at the same time allowing for patrolling and intruding agents to make decisions that facilitate their competition in the game model. Several decisions have been made in order to construct GAMMASys in a certain way, such as the decisions to construct a grid that is visualized by the user, implementing a genetic algorithm for automatically searching for effective patrolling strategies, and creating a simulation feature. These decisions are intended to contribute to generating effective patrolling strategies for the patrolling agent to use against an intruding agent.

If adversaries in GAMMASys are intelligent, or react in a formidable manner based on previous experience, then an agent can make effective responses from these adversarial decisions that make their strategies less predictable and less exploitable. By making an intruder more intelligent and formidable, the patroller must respond with effective strategies that can improve their chance of being successful. In this sense, the patroller can cover an action-space effectively if an intruder is less predictable than an unintelligent agent, or an agent that does not make their decisions on past adversarial interactions. If agents use unpredictable methods, and a patroller

can develop strategies that improve coverage of a zone, then an analyst could also use this system for resource allocation. Resource allocation is and important objective because funds for border security officials are limited. Therefore, if a border analysis can determine how many agents should cover a zone at a given time, and with what travel means, then a zone can be effectively covered given limited resources. It should be noted, however, that even though these responsive agents are considered intelligent within this research study, the choices of these agents may not compare to the choices or strategies that a human would select.

Our research is relevant to ***behavioral game theory***. Simon Gächter, author of the "Blackwell Handbook of Judgment and Decision Making," states that behavioral game theory is an approach to seek empirical information about how human beings behave in strategic situations, as opposed to the behavior of highly rational beings or programmed strategies. Gächter states that the goal of behavioral game theory is to discover theories that rest on plausible psychological foundations, and is about making game theory a more powerful tool for the analysis of strategic situations (Gächter, 2004). Behavioral game theory is relevant to our research, because adversaries within our game model make adaptive decisions. Xing & Viriyasuthee state that behavioral game theory is a study of actual individual's behaviors in games (or strategic situations), in which individual's decisions depend on their adversary's or partner's choices. They also state that the aim of behavioral game theory is to predict how people actually behave (Xing & Viriyasuthee, 2011).

The system uses a grid that is visible to the user and represents the models that are used. This grid is used and weighted by terrain values to allow the user to observe the interactions between adversaries within the game and allow for adversaries to make movement decisions. The grid is constructed with hexagonal nodes, which allow for movement in six directions, as

well as the ability to stay in place. The more directions an agent can move, the more realistic the movement patterns (as opposed to a rectangular grid). The issue with this is that the more directions an agent can travel, the more complex the model is. This is the reason that the study uses a hexagonal grid. The grid also allows for each node to the weighted, which affects the travel time of vehicles. Using multiple types of terrain can allow adversarial agents to produce complex strategies which can be unpredictable. It should be noted that the area of nodes on the grid is based on the zoom level of a satellite map, and the location of a hexagonal node may depict an area on the map, instead of an exact location. Also, the relationship between the terrain weights and a vehicle's travel time is selected to be representative for this project, and may not be fully accurate. Moreover, the model can accommodate my values.

The genetic algorithm is automated and runs a variable number of generations and trials, which allow new and effective patrolling strategies to be created. Border security officers may need a fast and effective way to produce patrolling strategies for their respective zone, and a genetic algorithm can use exploration to find strategies that exploit weaknesses in the intruder's travel strategies. The genetic algorithm uses randomization to produce strategies, and relieves the human operator from making decisions that could be biased. . Since the computational model is based on an adversarial model, the algorithm can be altered according to this game theoretic model.

2.9. Scenarios

The characteristics of the game space and the agents have evolved in GAMMASys. Visualizing scenarios can assist in explaining how GAMMASys works and can assist in explaining how this project will progress to find better patrolling solutions for real-world use.

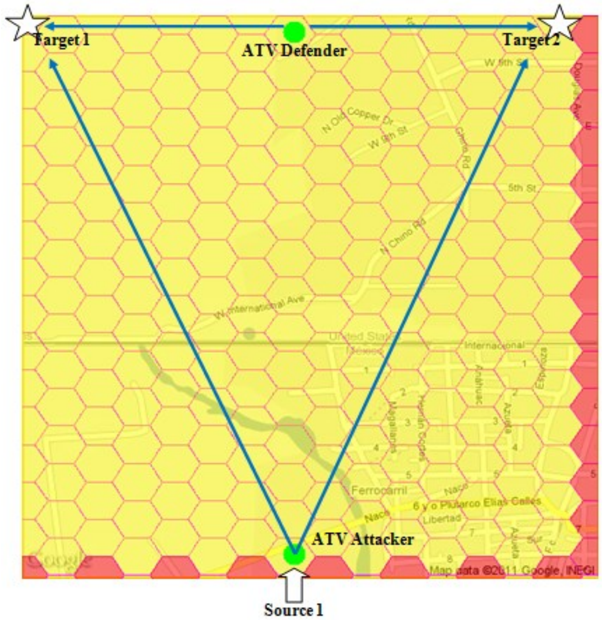

Figure 3: ATV Intruder/Patroller (1 Source, 2 Targets) (Gutierrez, 2013). Attacker is an Intruder, Defender is a Patroller.

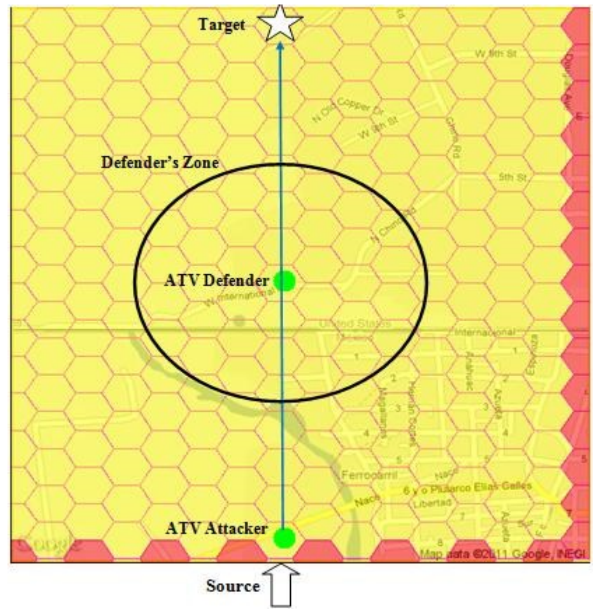

Figure 4: ATV takes shortest path to Target Node (Gutierrez, 2013). Attacker is an Intruder. Defender is a Patroller.

In Figure 3, an intruder is located at its source node at the bottom center of the grid. Two target locations are options for the intruder to attempt an intrusion; these are located at the top left and top right portions of the grid. The patrolling agent's starting location is at the top middle of the grid, and there exists only one of each agent. Both agents are using ATVs as their travel means. While this might not be a realistic scenario where an intruder would use an ATV, this setup within the game creates a more balanced scenario, where travel times across certain terrain between the adversaries will be similar. The intruder has a choice between both target nodes, and can choose either one. The patroller must respond by appearing at the node of the chosen target of intrusion at the same time the intruder arrives at that node. Due to the randomized patroller strategies, which represent choices of travel for the patroller, there is a chance that the patroller would take a direct route to the chosen target node. This is why a genetic algorithm must be used, in order to run numerous simulations in order to generate patroller strategies that have better probabilities that lead towards the direction of the chosen target node.

In Figure 3, it is shown that this game is simple, yet difficult for the patroller with an unimproved, random strategy, because that patroller has a minimal chance of appearing at either target node, or of intercepting the intruder during its route. Figure 4 is much simpler for the patroller to capture the intruder, because the patroller's starting location is right in the intruder's shortest path. If the intruder is not cautious, the intrusion path will travel right through the patroller's starting location, and the patroller can apprehend the intruder if the patroller stays in its starting area. Only one target and source node exists and they are in line with the patroller's starting location. The intruder has been improved though, because a shortest path may not always be beneficial if the patroller is dealt a better patrolling strategy by chance. The intruder can now stay at its source node and observe the patroller's initial moves. The area which the patroller travels during its initial moves of the simulation is known as the ***patroller's zone***, or the ***defender's zone*** in Figure 4. Currently, the patroller is given a random time step in which it is allowed to move, while the intruder does not take any action or make any movement on the grid. For each simulation that is executed, the intruder might begin their route at a different time step, resulting in the patroller being in a different position when the simulation begins. The random time step in which the intruder moves is not a high enough value for the patroller to capture the intruder while the intruder remains stationary. This also creates an opportunity for the intruder to observe the patroller, but currently, a Monte Carlo simulation runs a full practice simulation, before the real simulation commences, in order to allow the intruder to observe a sample patroller strategy. The reason for improving the intruder's strategy is that, in a real-world scenario, intruders can adapt to decisions made by the patrollers, thereby can making them formidable adversaries. In order for patrolling solutions to be effective in real-world application, they must be generated based off intelligent adversaries.

Figure 5 depicts an attempt to increase the formidability of the intruder by implementing a new capability called ***caution***. Since the intruder can already observe the previous practice moves of the patroller, then the intruder can relocate his path outside the patroller's zone, and find the shortest path that avoids that area. Although the intruder travels outside this zone, the patroller can still capture the intruder depending on the probability distributions after the intruder starts moving. Theoretically, this should make it more difficult for the patroller to capture the intruder, thus increasing the his formidability. The patroller, during the Monte Carlo simulation executes a sample route of the strategy that is being observed by the intruder. After the Monte Carlo simulation, the patroller might execute a different route from the same patroller strategy, which can be misleading for the intruder. The intruder will have constructed an evasive route according to the sample strategy it observed during the Monte Carlo simulation, if the intruder was cautious.

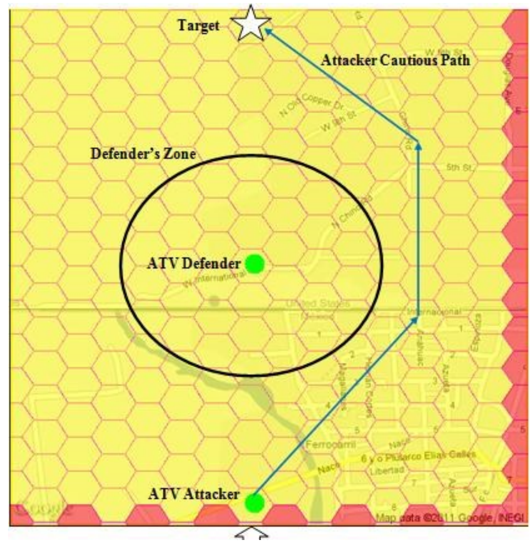

Figure 5: Intruder observing patroller's previous moves and reacting with caution (Gutierrez, 2013). Attacker is an Intruder, Defender is a Patroller.

The preceding section demonstrated the motivating factors behind the characteristics attributed to patrollers and intruders and decisions that these agents can make to become more formidable adversaries. As adversaries make more sophisticated decisions, the more compatible the patrolling solutions can be with real-world environments. The grids in Figures 3-5 are depicted as a uniform desert terrain with one adversarial agent on each side. GAMMASys allows for more complex scenarios to be constructed. Uniform desert terrain, like the 15x15

desert grids used in Figures 3-5 simplify the game scenario. It is important to construct more complex game scenarios, so we can observe decisions made by the agents in border zones with more complex terrain. If agents are making competitive and formidable decisions on highly complex terrain, then game spaces with realistic terrain layouts can be created for real-world use, in the future.

Chapter 3: Solution Alternatives

This section contains graphical representations of the results of test cases run on GAMMASys. Displaying these results in this document is important, since they will influence further test cases that will be conducted to improve the system. Future test cases can then be compared to previous ones in order to evaluate solution improvement. For instance, if the results show that a certain patroller strategy is dominant against other patroller strategies, analysis on that strategy will be conducted in future work, in order to improve upon that patroller strategy. This approach will also be taken with genetic algorithm parameters that are used during the tests. If better genetic algorithm parameter combinations are found, they will be further tested, and possibly used in future experiments.

3.1. Adversarial Interactions

In the previous section, Figure 3 depicted a game with one patroller and one intruder with two target nodes, one on each side of the patroller's starting location. Both agents are using the same unit type, the ATV. Tests can be conducted in order to analyze the results of a game as a result of questions that the researcher might have about the relationship between adversaries. In this case, the agents are evenly matched, and the only uncertainties that lie within this game are the direction-decisions the patroller will take at each turn, and the node that the intruder will choose to target. What if the patroller and intruder had different travel speeds? Would this affect the patroller's rate of interdiction? Figures 5 and 6 show graphical representations of the solutions of a test case where the speeds are altered between the patrollers and intruders. In this test scenario, the speed of the patrollers and intruders are delayed by different multipliers (e.g., made slower by two and three times the normal speed of the agent). This is based on travel

delay, or the amount of time steps it takes for an agent to travel through a certain type of terrain on the grid space. To simplify the test scenario, the grid is of uniform desert terrain.

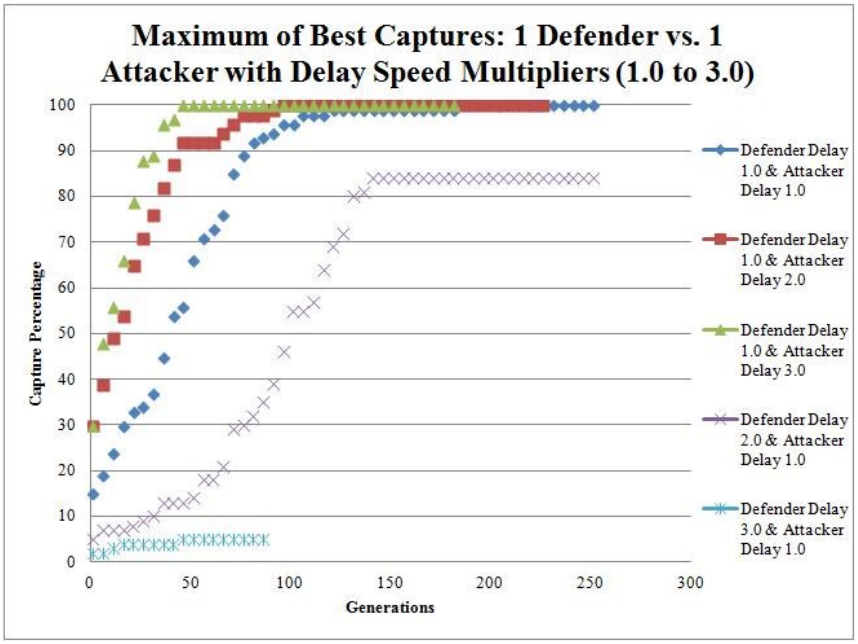

Figure 6: Average of best captures based on delay (Gutierrez, 2013). Attacker is an Intruder, Defender is a Patroller.

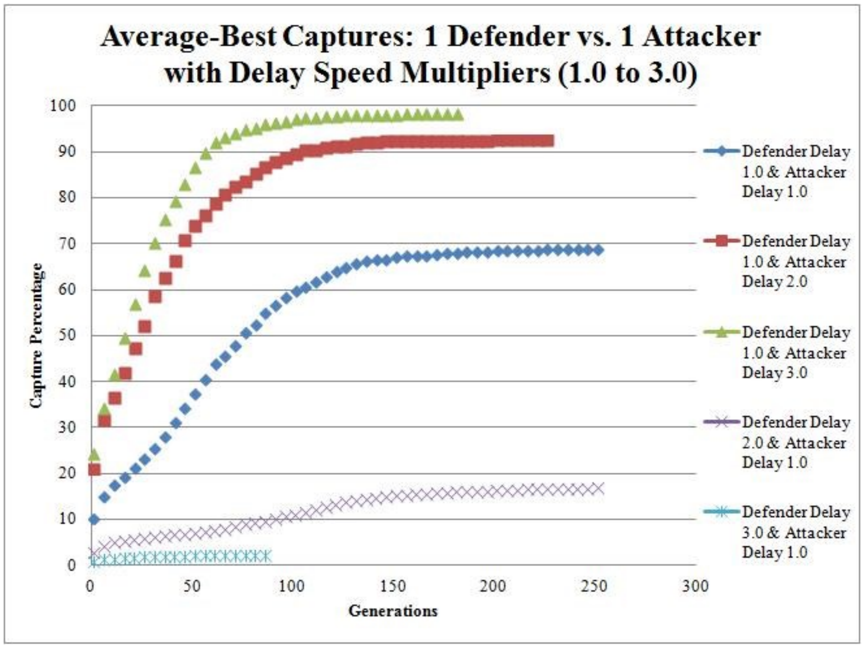

Figure 7: Max of best captures based on delay (Gutierrez, 2013).
Attacker is an Intruder, Defender is a Patroller.

Figures 6 and 7 are older results from an older version of GAMMASys' genetic algorithm. These results are relevant, because the movement speed of the agents does still affect capture rates in the current version of GAMMASys. Due to time limitations these tests were not reconstructed. These figures depict the maximum-best captures and average-best captures of the speed delay test. What this means is, of the elite strategies in the population of patroller strategies, the top strategy of every generation was recorded for Figure 6, and all of the elite strategies were averaged for every generation for Figure 7. These techniques can also be used for the average captures and best captures across the full population. However, this research project is focused on the best strategies that can be produced from simulation, and not on the full population of strategies. All of these test cases in the test scenarios are cut off before 300 generations. This is because GAMMASys has a feature that halts the running of additional simulations if the strategies are not improving in captures within a set number of generations. This has been done in order to save computational time. The results show that when the patroller is slowed by a multiplier of three compared to the intruder, then the patroller capture percentage

converges towards the optimal solution the fastest in Figure 6. When the intruder is slowed by a multiplier of two, the patroller capture percentage converges at a similar, but slower rate to the previous test case. When the patroller and intruder move at the same speed, the patroller capture percentage converges towards the optimal solution in around one hundred generations in the maximum-best captures graph. The worst case is when the patroller is slowed by a multiplier of three compared to the intruder, which results in a very low capture percentage and is cut off around one hundred generations.

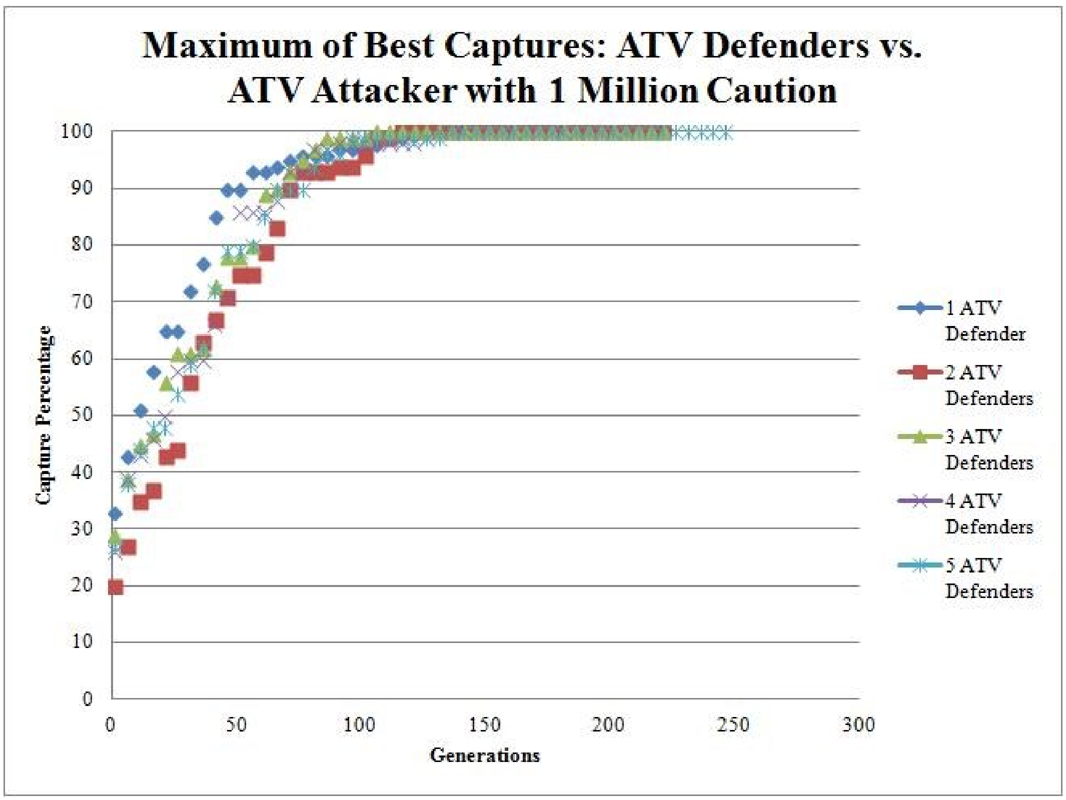

Figure 8: Max of best captures of multiple patrollers vs. intruder with caution (Gutierrez, 2013). Attacker is an Intruder, Defender is a Patroller.

In Figure 3, the game scenario is simplified by using a uniform terrain, limiting the number of adversaries to one patroller and one intruder, and limiting the number of source and

target nodes to one each. In Figure 5, this same scenario is applied, but to avoid running into the patroller's zone, caution is applied to alter the path of the intruder. If the border security official were to run a visual simulation of the game scenario, they would see an intruder cautious path similar to that of Figure 5, but how effective is caution in this game model? Would more patrollers increase the rate of convergence to that at which the capture rate reaches the optimal solution? According to Figure 8, the test cases using between one and five patrollers seem to converge towards the optimal solutions at approximately the same rate. This test scenario uses a caution value of one million, which is an acceptable value for the intruder to take evasive action of the patroller, in this version of the caution mechanic. A test scenario similar to that of Figure 8 was executed and contained a caution value of zero, which would set the intruder to just execute a shortest path route. This test scenario produced results similar to Figure 8. Due to these tests we can observe that, in this scenario, the number of patrollers did not make much of a difference with regards to the capture percentage of intruders, and the caution had little to no effect on intruder evasion. These results can be attributed to numerous factors, such as the terrain having no variation, giving the intruder a simple route alteration to the outside of the patroller's zone. Also, the simple nature of the scenario might not allow for significant improvement in the capture rates. In this sense, it would be more beneficial for the patrollers to have one fast travelling patroller instead of five patrollers that match the speed of the intruder. In Figures 6 and 7, it can be concluded that speed delay of agents can affect capture rate, if the patroller has a different delay than the intruder. If the patroller's delay is higher than the intruder's, then the capture rate decreases. If the patroller's delay is lower than the intruder's, then the capture rate increases.

It is important to ask questions about the game model when analyzing adversarial matchups such as the patroller/intruder game. The questions that have been asked in this section were:

- How does the speed delay of the agents affect the patroller capture rate?
- How does an increase in the number of patrollers affect the patroller capture rate?
- How does intruder caution affect the patroller capture rate?
- How does grid size affect the patroller capture rate?
- How does the inclusion of multiple source and target nodes affect the capture rate?

These questions revolve around patroller capture rates, which is important to the goal of this thesis. By ensuring that the patroller strategies can improve to an effective level when modifying the game model, the more resilient these strategies can be in a highly complex environment, such as the U.S. borders.

3.2. Improvement of the Caution Mechanic

In Figure 8, the caution mechanic that has been used for this test is an older mechanic that was used in previous intrusion tests. The cautious penalized the intruder and contained flaws. The reason the older version of cautious is explained here, is to show that a non-penalized cautious mechanic makes the intruder more formidable, which can improve patrolling strategies.

Caution in our system is a value that is multiplied by the probability that a patroller will be in a certain location that is being assessed when an intrusion route is being constructed (this is found using the Monte Carlo simulation in GAMMASys). Once the caution value is multiplied by the probability of that the intruder will be captured if they travel along the current path, the number of time steps that have occurred in the simulation is added. This value is the ***path cost*** (*path cost = path distance + [caution * probability of capture]*). The information of time steps and the probability of capture are given from neighbor assessment. Dijkstra's Shortest Path

Algorithm is used to find the travel distance from the starting location of a graph to all other locations on the graph. This is how we find out the shortest distance from the source and target locations on our grid. In order to find the distance of all locations on a graph, we must choose a location to assess, and then assess the neighbors of that node. When finding the shortest path using caution, we assess neighbor capture probabilities (*neighbor capture probability = node capture probability + [1 – node capture probability] * probability of a patroller occurring in this node*) , as well as neighbor distance (*current node distance + neighbor travel cost based on vehicle type*). Overall, the shortest path will be chosen based on path cost, which is a combination of path distance (number of time steps) and the probability of capture (weighed by caution amount).

In Figure 8, the caution that has been used for this has three flaws. First, cautious intruders move slower than non-cautious intruders as well as the patrollers. Second, these cautious intruders have a delay in order to observe the initial movements of the patroller, making the intruder start their intrusion route at the same time step for every simulation, Finally, this cautious intruder only takes into account the probability that a patroller will be in a certain node, not the probability that the intruder will be captured along the route taken. The cautious intruder used in the test cases shown in Figure 8 was of a previous intruder type that was used in GAMMASys. Currently GAMMASys uses an ***improved cautious intruder***, or an artificial agent within GAMMASys that uses an evasive caution mechanic that does not slow down the agent. This intruder chooses an intrusion route that is a ***cautious shortest path***. The intrusion path is selected using an altered form of Dijkstra’s Shortest Path algorithm. The difference between the improved cautious intruder, and the previous cautious intruder or the ***penalized cautious intruder***, an artificial agent in GAMMASys that uses an evasive caution mechanic that slows

down the movement of the agent, is that the improved cautious intruder bases its shortest path on probability that the intruder will be captured by the patrollers, and the distance travelled based on the vehicle-terrain travel time, or the total travel distance of the path. In turn, the improved cautious intruder will choose the path that has the lowest capture probability and the shortest distance travelled, based on the caution weight. If the user tests caution ranging from 0 to 100, then the official will observe the intruder being non-cautious at a 0 caution value, where the intruder will place all of the weight towards shortest distance travelled and will have no concern about the probability that it will be captured along its path. If the caution value is placed at 100, the intruder will try to take the shortest path, while avoiding capture. Future tests should observe the evasiveness of the intruder with cautious values between 0 and 100.

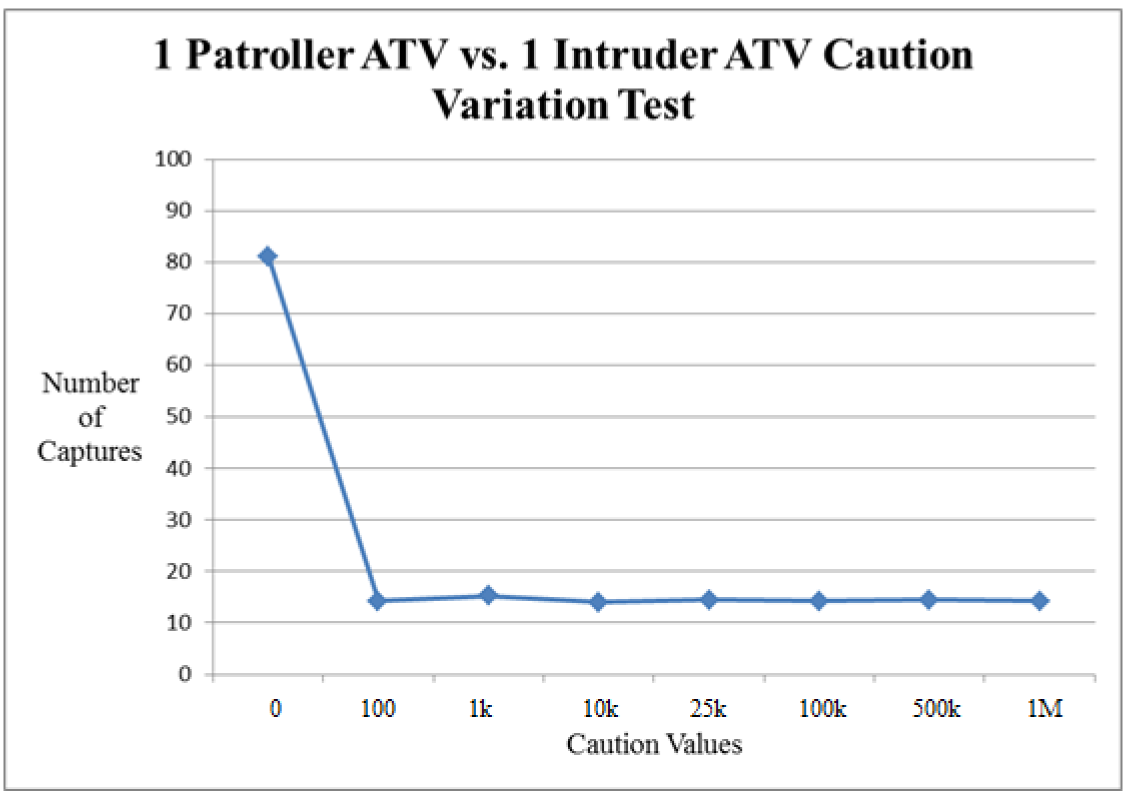

Figure 9: Maximum value of averaged-best captures of 1 ATV Patroller vs. 1 ATV Intruder with varying caution between 0 (no caution) to a 1 million caution value.

Figure 9 represents a test case where the improved cautious intruder using an ATV for travel means is attempting to intrude towards a target node through uniform desert terrain with a 15x15 area space within the game model. The genetic algorithm settings for this test were: 100 trials of tests; 500 generations per trial. Trials were cutoff after 30 generations of no improvement. We used a population size of 100 multi-strategies. Strategy distribution was: 25% elite strategies; 25% mutated strategies; 25% crossover strategies; 25% averaged strategies. Node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The source and target node selections were not random, and were chosen by the intruder. The intruder started at a random time step between 0 time steps and 150 time steps. This time step starting point, that is chosen at random, is meant to simulate an intruder beginning their route in the middle of a patrolling route. This is important because the patroller could exploit the positioning of the intruder for every simulation, because the intruder and patroller movements could become synchronized during the simulations.

Table 1: Maximum value of averaged-best captures of 1 ATV patroller vs. 1 ATV intruder with varying caution from a 0 caution value to a 1 million caution value, in non-consistent intervals. The underlined value is the most evasive intruder against its comparisons.

Caution Variation Test: 1 ATV Patroller vs. 1 ATV Intruder (0 Caution – 1 Million Caution)								
	Caution Values							
	0C	*100C*	*1kC*	*10kC*	*25kC*	*100kC*	*500kC*	*1MC*
Number of Captures	81.02	14.27	15.28	<u>14.03</u>	14.49	14.24	14.51	14.19

Based on the graph, it can be concluded that after a caution value of 100 there is not much change in the capture rates on the succeeding caution values. . The maximum values of averaged best captures in the caution variation test are presented in Table 1. The purpose of caution is to allow the intruder to execute routes that avoid capture by their adversaries. If an intruder's route is effective against the patroller, one would expect to see low capture rates. In this caution variance test, all result past 0 caution are relatively low compared to the 81.02% patroller capture rate against the non-cautious intruder. As demonstrated by Table 1, the intruder with the lowest capture rate is the intruder operating with a 10,000 caution value. It can be argued that a 10,000 caution value is the best choice for caution values in a 15x15 uniform desert terrain map, with a similar adversarial matchup, but the capture rate difference between the 10,000 caution intruder and the other caution intruders is miniscule.

There are other factors that could cause the capture rates to fluctuate, such as the patroller choosing its path based on the random probability distributions in the patroller. The intruder starts at a random time step between 0 and 150 time steps for each simulation. This means that for each patroller-intruder matchup, the patroller has the chance of being at a different location at the beginning of every simulation when the intruder begins its intrusion route. Finally, the Monte Carlo simulation may not allow the intruder to observe the route that will be executed by the patroller in the real simulation. For example, in a real-world scenario, the Monte Carlo simulation would be similar to an illegal intruder observing patroller routes a day before they actually execute their intrusion strategy. In this sense, the illegal intruder may plan their intrusion plan with regards to the patrol route they observed on the previous day, but this patrol route is subject to change when the intruder officially executes its intrusion strategy.

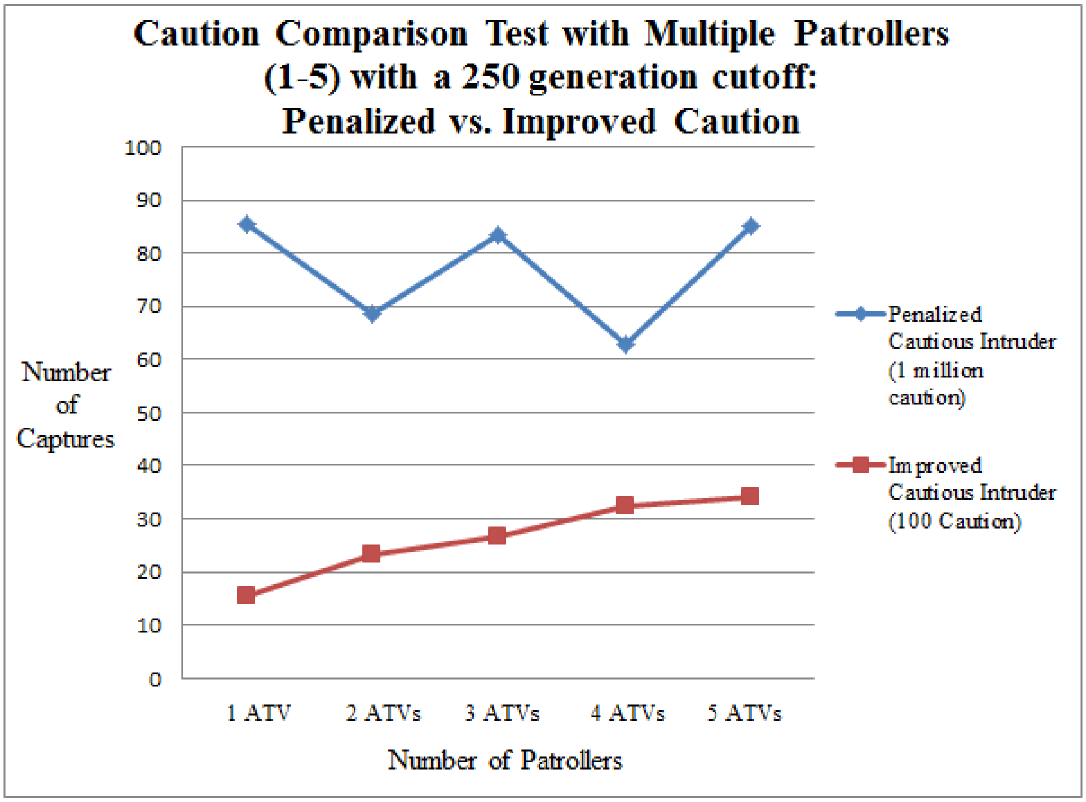

Figure 10: Maximum value of averaged-best captures between multiple patroller test (1-5) with a 250 generation cutoff. Penalized Cautious Intruder vs. Improved Cautious Intruder.

Figure 10 represents a comparison of test cases involving the penalized cautious intruder that was used in the test case in Figure 8. In this comparison, a test case using the penalized cautious intruder was plotted with the maximum of the averaged-best captures between patrollers using non-restrictive Markov strategies and the penalized cautious intruder. The second test case plotted for comparison is the test case between patrollers using non-restrictive Markov strategies and an improved cautious intruder.

The outcomes of the first test case, in Figure 10, are represented by the blue-diamond plots, where the penalized cautious intruder, equipped with an ATV, is competing against 1-5 ATV-using patrollers employing non-restrictive Markov strategies. This test case involves the intruder trying to reach its target destination on a uniform desert terrain with a 15x15 area space

within the game model. The genetic algorithm settings for this test were: 100 trials of tests, 250 generations per trial, and trials were cutoff after 20 generations of no improvement. We used a population size of 100 multi-strategies. The population distribution was: 25% elite strategies, 25% mutated strategies, 25% crossover strategies, and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The source and target node selections were not random, and were chosen by the intruder; and, the intruder does not start at a random time step throughout each simulation.

The second test case depicted in Figure 10, is represented by the red-squared plots where the improved cautious intruder equipped with an ATV is competing against 1-5 ATV patrollers using non-restrictive Markov strategies. This test case involves the intruder trying to reach its target destination on the same uniform desert terrain with 15x15 area space within the game model. The genetic algorithm settings for this test were: 100 trials of tests, 500 generations per trial, and trials were cutoff after 30 generations of no improvement. The population size was 100 multi-strategies. The population distribution was: 25% elite strategies, 25% mutated strategies, 25% crossover strategies, and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The source and target node selections were not random, and were chosen by the intruder; and the intruder starts at a random time step between 0 time steps and 150 time steps.

It is important to note that there are differences between the two tests outlined above in Figure 10. Initially, the first test (blue-diamond plots case) case is not using the current caution mechanics that are used by GAMMASys, while the second test case (red-squared plots) are using the current intruder caution mechanics employed by GAMMASys, which incorporates the improved caution mechanics. Second, the number of generations used by GAMMASys varied

for these test cases. The test case involving the penalized caution mechanic halts at 250 generations, while the test case involving the improved caution mechanics halts at 500 generations. There was a difference in generation improvement cutoff as well. Where the first test case cut off after 20 generations of no improvement, the second test case cut off at 30 generations of no improvement. The last difference in the test cases is that, while the first test case does not allow for the intruder to execute its intrusion strategy at a random time step for each simulation, the second test case does allow the inclusion of this metric.

The only difference in the test setups that cause the capture rates between the two test cases to vary significantly is the differences in the caution mechanics. The first test case uses caution mechanics that cause the intruder to move slower, delay their start time to observe the patrollers' initial moves, and to measure the probability that the patroller will be in their given location when constructing an intrusion route. The second test case uses caution mechanics that allow the intruder to move at the same speed as the patrollers (if they are of the same vehicle type) executes an intrusion route at a less predictable time step throughout each simulation, and constructs an intrusion route that more accurately measures the probability of capture, as opposed to the less accurate probability that a patroller will be in a certain node at the time of construction. The other differences do not cause significant variation. The second test case in Figure 10 was recorded using a 250 generations limit, ignoring the full set of results leading up to the 500 generations limit. In order to match the first test case, the generation improvement cutoff variables did not matter, because the second test case was altered to match the cutoff of the first test case. Also, the random execution time for the intruder did not make much of a difference, because, for the first test case, the intruder might be more predictable starting at the same time step for every simulation, but the intruder is also given the probabilities that the

patroller will be in a given location. Finally, the patroller is given no information about the intruder unless a capture occurs. For the second case, the intruder observes previous patrol routes from the patroller, not the initial moves of the patrol route. Therefore, the decisions made by the patroller during the simulation might vary from those made during the Monte Carlo simulation, putting the intruder at a slight disadvantage.

Table 2: Maximum value of averaged-best captures between multiple patroller test (1-5) with a 250 generation cutoff. Penalized Cautious Intruder vs. Improved Cautious Intruder. Underlined values are more evasive intruders.

<table>
<tr><td colspan="7">Caution Comparison Test: Multiple Patrollers (1-5 ATVs) vs. 1 ATV Intruder with a 250 generation cutoff. Penalized Cautious Intruder vs. Improved Cautious Intruder</td></tr>
<tr><td colspan="2" rowspan="2"></td><td colspan="5">Patroller Teams</td></tr>
<tr><td>1 ATV Patroller</td><td>2 ATV Patrollers</td><td>3 ATV Patrollers</td><td>4 ATV Patrollers</td><td>5 ATV Patrollers</td></tr>
<tr><td rowspan="2">Number of Captures</td><td>Penalized Cautious Intruder</td><td>85.52</td><td>68.54</td><td>83.42</td><td>62.94</td><td>85.2</td></tr>
<tr><td>Improved Cautious Intruder</td><td><u>15.33</u></td><td><u>23.2</u></td><td><u>26.77</u></td><td><u>32.5</u></td><td><u>34.25</u></td></tr>
</table>

According to the results given in Table 2, the improved cautious intruder is more evasive than the penalized cautious intruder against all five teams of ATV patrollers. Also, the capture rates of the penalized cautious intruder fluctuate nonsensically. Ideally, one would expect that a cautious intruder would be less successful as the number of patrolling agents increases in a 15x15 patrolling zone. If the intruder is less successful, then the capture rate will increase. This is the case, where the improved cautious intruder's capture rate increases at the same time as the

number of patrollers increase. In this sense, the improved cautious intruder is more evasive than the penalized cautious intruder. To test the research questions presented in this paper, the improved cautious intruder will be used, because this intruder is more reliable when assessing coverage of a patrolling zone. The use of an effective intruder results in a more accurate assessment of border coverage. An improved cautious intruder will be designated as an intruder with a caution value of 100. Figure 9 and Table 1 show that after 100 caution, the fluctuation of capture rates is miniscule, or at least within a positive or negative 2% range. With this in mind, there is no reason to range caution outside of a value of 100. In future tests regarding this research, caution values between 0 and 100 will be analyzed.

3.3. Problems to Address Through Research

This research focuses on several experiments that demonstrate the usage of the game theory model as it relates to this project. These tests focus on the capabilities of the computational model, the genetic algorithm, and the behavior of the agents. Some tests use several scenarios, in order to test for different types of cases that serve as indicators for the impact of the result.

3.3.1. Research Question One Analysis

The first research question is: what effects does facing a more intelligent intruder have on the effectiveness of a patrolling strategy? It is important to test whether or not actions taken by the intruder will cause the capture rate of the patrolling agent to be significantly affected. A non-cautious intruder will take the shortest path for every simulation with no regard for the probability that this agent will be captured en route. A cautious intruder will take the shortest intrusion route possible while attempting to avoid capture. If the intruder is trying to

avoid capture, the patroller might alter its patrolling strategy in order to maximize the capture rate against this cautious intruder. This can be done by altering the probability distributions of the patrolling strategies for each hexagonal node in a sophisticated fashion, such as modifying the patroller strategies with a genetic algorithm. The concern is that, if the patroller is using a baseline strategy, it cannot improve the patrolling strategies in a sophisticated fashion that would maximize coverage of a border zone. If an intruder uses different intrusion strategies that are more sophisticated, it can be expected that the intruder will be more evasive against a baseline patrolling strategy.

To test whether intelligent intruders affect the capture rates significantly, sophisticated intruder strategies must be constructed, and a baseline patrolling strategy must be used against varying types of intruders. There are four intruder strategies that will be tested against a baseline patrolling strategy. The first intruder strategy is the ***non-cautious—non-random intruder strategy***. In this strategy, the intruder takes the shortest path, is given a caution value of 0, and tries to reach the target node from the source node in the shortest amount of time steps while choosing the best source-target node pairs to accomplish the agent's objective. The second intruder strategy is the ***cautious—non-random intruder strategy***. In this strategy, the intruder uses improved caution mechanics, is given a caution value of 100, attempts to reach the target node from the source node in the shortest amount of time steps while avoiding capture, and the intruder chooses the best source-target node pairs to accomplish its objective. The third intruder strategy is the ***non-cautious—random intruder strategy***. In this strategy, the intruder takes the shortest path, is given a caution value of 0, and tries to reach a random target node from a random source node in the shortest amount of time steps, where source and target node pairs are chosen from a ***node set*** provided by the user. This node set is a list of source and target nodes

that the intruder can use as starting and intrusion points during simulation, respectively. The fourth intruder strategy is the ***cautious—random intruder strategy***. In this strategy, the intruder uses improved caution mechanics, is given a caution value of 100, and attempts to reach a random target node from a random source node in the shortest amount of time steps while avoiding capture, where the source and target node pairs are chosen from a node set provided by the user.

Finally, the baseline patrolling strategy will be the ***restrictive consistent distribution patrolling strategy***, where GAMMASys keeps track of a reference hexagon which is given a random probability distribution. From here, the patrolling strategy for every hexagon in the grid will use this same probability distribution. This distribution will not be uniform for two reasons. First, the outer edges of the grid do not have the same number of legal moves as inner-grid hexagons, so their probabilities will be renormalized for a proper distribution. Second, some of the terrain in the grid may not allow certain agents to travel through these terrain mapped hexagons, based on their vehicle types, so there will be less legal moves appointed to these hexagons than the seven allotted moves in a non-restricted hexagon, meaning that the probabilities in this restricted hexagon will have to be renormalized. This strategy is suitable for a baseline patroller strategy, because all hexagons in the grid derive from the same probability distribution. These probabilities are altered only to ensure that all legal moves are accounted for, and patrollers and intruders cannot venture outside the action space, or travel through terrain in which their vehicle is not properly equipped. An example of an illegal move based on improper vehicle movements on restricted terrain is that of a boat-based agent moving on to a hexagon mapped with mountainous terrain within the terrain grid. This strategy is also suitable for determining a baseline patrolling strategy, because the only improvements made in the patrolling

strategy population are that a percentage of the population is renormalized for strategy exploration purposes. There is no guarantee that this alteration will result in significant improvement to the patroller's capture score.

To test the adversarial interaction between intelligent intruder strategies and the baseline patrolling strategy, a border patrol scenario must be constructed. The border region can be modeled with a uniform terrain distribution where the adversaries have similar travel means such as a uniform desert terrain where the intruders and patrollers are all using ATVs to strive towards completing their objectives. We expect that the most evasive intruder strategies (lowest capture rates) to the least evasive intruder strategies (highest capture rates) competing against the baseline patrolling strategy will be: the cautious—random intruder, the cautious—non-random intruder, the non-cautious—random intruder, and the non-cautious—non-random intruder.

If intruder strategies become significantly more evasive against a baseline patrolling strategy, then one can conclude that improved intrusion strategies affect baseline patrolling strategies significantly. The reason the cautious—random intruder was selected for this test is because this type of intruder not only tries to avoid capture while taking the shortest intrusion route based on its source/target nodes pair, but it can also take a path that is less predictable. The cautious—non-random intruder was selected for this test is because this agent will try to avoid capture while choosing the best path to complete its objective. The non-cautious—random intruder was selected for this test, because this intruder executes a less predictable route (since the source/target nodes pair changes), and will try to arrive at the attack location as quickly as possible without focusing on capture evasion. The least intelligent intrusion strategy, the non-cautious—non-random strategy is still important for this test, because this intrusion strategy focuses on the absolute shortest path between the source/target nodes pairs provided, and the

intruder tries to reach the target location as quickly as possible without focusing on capture evasion. This intrusion strategy is considered the least intelligent strategy, because it does not try to avoid capture and only tries to reach its target location, regardless of the patroller's decisions. The test results of the adversarial interaction between a baseline patrolling strategy and varying intruder strategies are represented in Figure 11.

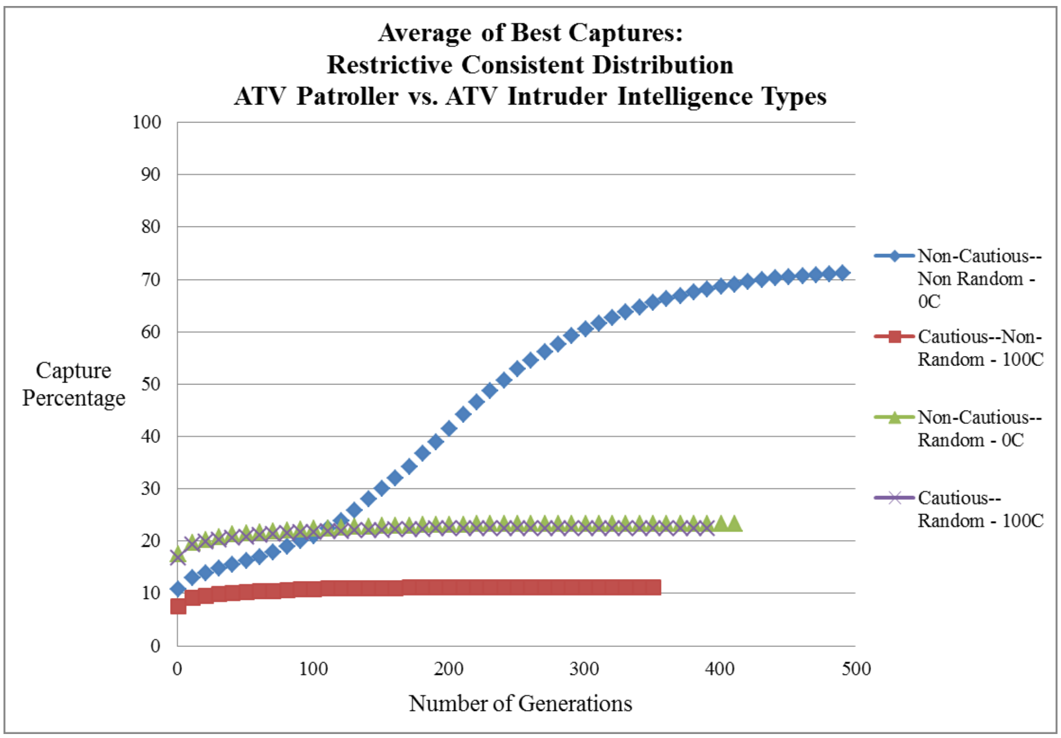

Figure 11: Averaged-best captures between 1 ATV patroller using a restrictive consistent distribution strategy, and 1 ATV intruder with 4 different intrusion strategies.

The outcomes of this test case, shown in Figure 11, are represented by lines that signify the average of the best capture rates against intruder strategies. When GAMMASys runs the genetic algorithm, multiple trials containing even more generations are run through the system. The system then records all the best capture rates from each trial, and then averages them. This is what is meant by the average of best capture rates. This test case involves four separate

scenarios, where a baseline patroller strategy, the restrictive consistent distribution patrolling strategy, competes against a non-cautious—non-random intruder (blue diamond plots), a cautious—non-random intruder (red square plots), a non-cautious—random intruder (green triangle plots), and a cautious—random intruder (purple cross plots). The goal for the adversaries is to reach its target destination on a uniform desert terrain with a 15x15 area within the game model, and for the patroller to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials of tests, 500 generations per trial, and trials were cutoff after 100 generations of no improvement (to ensure the trials cutoff only when there is very little potential for improvement). We used a population size of 100 multi-strategies. The population distribution of the population was: 25% elite strategies, 25% mutated strategies, 25% crossover strategies, and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid, and the starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random and the cautious—non-random intruders, the selection of the source and target nodes was made by the intruder. For the non-cautious—random and the cautious—random intruders, the source-target pair nodes were selected randomly from the 3 source node set and the 3 target node set; intruder starts at a random time step between 0 time steps and 150 time steps.

Table 3: Averaged-best captures between 1 ATV patroller and 1 ATV intruder with 4 different intrusion strategies, including generation cutoff points.

Adversarial Interaction Test: Intelligent Intruder Strategies in ATVs vs. a Restrictive Consistent Distribution Patroller in an ATV				
	Intruder Strategies			
	Non-Cautious—Non-Random	*Cautious—Non-Random*	*Non-Cautious—Random*	*Cautious—Random*
Max Capture Percentage	73.26%	11.16%	23.36%	22.43%
Generation Cutoff Point	500	351	411	391

After the average of best capture rates are graphed and recorded, the maximum of the average of best capture rates are recorded. Essentially, these values are the expected capture rates of the patrollers for each trial, after they have completed. This value represents the absolute best capture amount by the patroller strategy against a particular intruder strategy. This value is generally recorded before the generation cutoff on the graph. According to the results outlined in Table 3, the most evasive intruder is the agent using a cautious—non-random strategy ending up with a maximum capture percentage at 11.16% with the generation cutoff at 351, meaning that improvement in captures stopped around 250 generations. The second most evasive strategy was the cautious—random intruder strategy ending up with a maximum averaged-best capture percentage of 22.43%, with the generation cutoff at 391 generations, meaning the improvement in captures stopped around 290 generations. The next strategy was the non-cautious—random

intruder strategy, which had a similar capture rate to the cautious—random intruder strategy which had a maximum averaged-best capture percentage of 23.36 % and cutoff at 411 generations, meaning that improvement stopped around 311 generations within the trial. The capture rate difference between these two intruder strategies is miniscule, and can be argued that there is not much difference in adversarial interaction between a restrictive consistent distribution patroller and intruders using non-cautious—random and cautious—random strategies. Finally, the least evasive intruder strategy is the non-cautious—non-random strategy, which has a maximum averaged-best capture percentage of 73.26%, and improves throughout the trials.

It is important to note that there are increased captures against this non-cautious, shortest path, intrusion strategy, but these increases can be from two sources. First, the genetic algorithm keeps track of improved capture rates, whether the strategy itself is improving, or if there are just improvements in capture rates due to coincidental captures (like a patroller using a random patrolling strategy, and capturing a non-cautious intruder). Another source could be from a percentage of the patroller population being renormalized for exploration purposes. Notice, however, that there is very slight improvement against the more sophisticated strategies.

From these results, one can conclude that intelligent intruder strategies do affect patroller strategies significantly, because the difference of the non-cautious—non-random and cautious—non-random intruder strategies is substantial (62.1% difference) and the differences between cautious—non-random and the random intruder strategies (non-cautious and cautious) are not dismissible (12.2% and 11.27%, respectively). The prediction that the cautious—random strategy would be the most intelligent was not correct, according to the results. The most intelligent intruder was the cautious—non-random intruder, because this agent was given the

option to choose its source/target nodes pair. That does not mean that the cautious—random strategy is not formidable, because it still performed remarkably well against the non-cautious—non-random shortest path strategy. Certain alterations could be made to the cautious—random and non-cautious—random strategies which would improve their evasiveness - such as removing bad source/target nodes pairs, such as those that require a high number of time steps. While different strategies affected the capture rates of the test case represented in Figure 11 and Table 3, the results of the two random intruder strategies were similar. This could be due to the amount of nodes allowed in the source and target node sets. In this test the source nodes were [14,0], [14,7], [14,14], and the target nodes were [0,0], [0,7], [0,14].

In future tests, it is recommended that intruders using random source/target selection methods should plan intrusion routes using a larger source/target set. For instance the source set for a 15x15 desert grid could include nodes [14,0] to [14,14], and the target node set would include nodes [0,0] to [0,14]. With these improvements, intruders using random source/target selection methods would receive a wider variety of source/target pairs that would allow for travel paths that require fewer time steps travelled. With the test case presented in Figure 11, two source/target pairs force the intruder to create a shortest path with more time steps travelled than other pairs. Node pairs [14,0] (bottom left of grid) to [0,14] (top right of grid) and [14,14] (bottom right of grid) to [0,0] (top left of grid) require more time steps than node pairs [14,0] (bottom left of grid) to [0,0] (top left of grid) or [14,14] (bottom right of grid) to [0,14] (top right of grid).

During a simulation an ATV takes 6 time steps to travel through one desert-mapped hexagon. If that ATV travels from [14,0] (bottom left of grid) to [0,0] (top left of grid), the total distance travelled is 90 time steps, which is 15 hexagons multiplied by 6 time steps. To travel

from [14,0] (bottom left of grid) to [0,14] (top right of grid) it takes 132 time steps while using an ATV on desert terrain, because the total travel distance is 22 hexagons, as opposed to 15 hexagons. With a longer path, the intruder is travelling an extra 42 time steps, which is nearly half the amount of time as a 15-hexagon intrusion route. Also, given that cautious paths could consist of more hexagons traveled, this would result in a higher time step count.

Some source/target node pairs are more costly to travel. This is why the cautious—non-random intruder is most effective, because the intruder chooses from the source/target node sets, and then the best path, instead of a source/target pair that is randomly chosen for the intruder planning the path. With intruders travelling these longer distances, the game continues for more time steps, and as long as the patroller is travelling, there is the potential for capturing the intruding agent. The patroller benefits from reaching the target location as quickly as possible while avoiding capture.

Research Question 1 Conclusion: The more intelligent the attack strategy used by the intruder, the more evasive the intruder will be against a base line patrolling strategy, and the capture rates will be lower. The least evasive intruder is the agent using the non-cautious—non-random intrusion strategy, while the most evasive intruder is the agent using the cautious—non-random intrusion strategy. The random intrusion strategies have similar capture rates, but creating a larger set of nodes for source/target pair selection could increase the evasiveness of these intruder strategies, because there are only nine possible routes that these sets allow. Similar sets of tests could be constructed for the purpose of analyzing different patroller strategies, as well as different intrusion strategies, and how these matchups affect capture rates. If tests can be constructed that observe interaction between different patroller and intruder strategies, then it can be concluded that adversarial interaction is occurring within the system,

and similar strategies can be practiced in real-world simulations before use in the field. Also, the cautious—non-random intruder is evasive and effective against the baseline patroller. By identifying an evasive intruder, we can identify sophisticated patroller strategies that can be more effective against evasive intruders.

3.3.2. Research Question Two Analysis

The second research question is: can more intelligent patrolling strategies increase capture rates against intelligent intruders? The effectiveness of intruder strategies has been analyzed, but now, more intelligent patroller strategies can be analyzed as well. It can be expected that capture rates should increase as intelligent patroller strategies are introduced. In order to test whether sophisticated patrolling strategies perform well against varying intruder strategies, we construct tests that allow these adversaries to compete for their objectives. The types of intruder strategies used to test the effectiveness of varying patroller strategies are those outlined in the discussion of the first research question. The intruder strategies include: the non-cautious—non-random, cautious—non-random, non-cautious—random, and the cautious—random intruder strategies.

The patrolling strategies that will be introduced are not be specifically designed to exploit these intruder strategies, but they are designed to assign probabilities for travel in different fashions, and will be processed through the genetic algorithm where improvements on the capture rates will, or will not occur. The total average of captures will be taken for each patroller type to determine which patroller strategy was most affective against the varying intruder strategies.

The first patroller strategy that will be used in this test is a ***random distribution patroller strategy***, where the population of patroller strategies will be randomly generated, and will not be altered or improved when run through the genetic algorithm. The reason that improvement will not be allowed for this strategy, is because it is meant to be of low sophistication. This means that the population of a patroller using this type of strategy will not be altered with mutation, crossover, or averaged strategies. If this strategy cannot improve, a low initial capture score is expected to stay low, and only slight increases in capture rates are expected. This is due to the fact that GAMMASys keeps track of best capture scores. Hence, an unchanging strategy can have an improved capture rate through capture successes, without altering the structure of the strategies. In this test, it was expected that this would be the least intelligent patroller strategy.

The second patroller strategy that will be used in this test is a ***restrictive consistent distribution patroller strategy***, which was the baseline patroller strategy used in the test regarding the first research question. The purpose of using this patroller strategy is that exploration for more effective patroller strategies is allowed, but only with the use of averaged strategies. This causes the probability distribution sets in each hexagonal reference node to be multiplied by random numbers, and the probability distribution will be renormalized. In this test, only 25% of the strategy population will be altered by renormalization. This strategy is meant for consistency of movement, meaning that before the patroller population is generated, the probability distribution for the whole grid is already determined. If some nodes on the grid have less than seven legal moves, they will be altered based on the referenced probability distribution so that they remain consistent. The population should allow for similar hexagons to have similar probability distributions, based on the number of access points in the node, number of legal moves allowed for exiting the node, and the type of terrain. It can be expected that this patroller

strategy might be effective against a non-cautious—non-random intruder, but less effective against the other intruder strategies. This patroller strategy can be designated as more intelligent than the random probability distribution patroller strategy.

The third patroller strategy that will be used in this test is the ***deterministic probability distribution patroller strategy***, which creates deterministic routes that are less flexible for change, but allows the patroller to be more mobile. This strategy only allows for patrollers to move in the six allotted directions on the hexagonal grid. Unlike the other patroller strategies, this patroller policy does not allow the patroller to stay in the hexagon for a certain number of time steps, which would be similar to a border patroller waiting in a specific location for a certain amount of time in a border zone. This patroller strategy allows the patroller to be mobile, but this also allows the intruder to predict the location of the patroller at any given time step, because the decisions of the patroller are predetermined. For instance, if a probability distribution for one hexagon lists six possible travel locations for the patroller, listing: travel north; travel northeast; travel southeast; travel south; travel southwest; and travel northwest; with no option to stay still; the system must choose one travel direction to be executed every time. If the decision "travel northeast" was selected, then every time the patroller reaches this node, it must travel northeast. An intruder that observes this type of behavior during the Monte Carlo simulation can learn to completely avoid the patroller if this strategy is executed. This is the case with the Monte Carlo simulation that is executed by GAMMASys, whereby the intruder gets to preview one full simulation of the patroller strategy before it executes its intrusion route. Arguably, this is more effective than the intruder halting for a preset amount of time and observing the patroller, because the intruder executes their path at a random time step, and there can be inconsistencies with how cautious an intruder can be. An example of this would be an

intruder using the improved caution mechanic, who must execute the intrusion strategy at the initial time step, where the intruder was unable to observe the patroller's initial movements; therefore the intrusion attempt lacked cautious planning.

This patroller strategy can be very effective in creating patrol routes that can be simulated in real world scenarios, because there is no option for the patroller to stop, and by using the genetic algorithm, this patroller can learn to create looped routes that cover more area of the border zone. GAMMASys uses a different genetic algorithm that alters this patroller strategy. In the algorithm that alters the deterministic distribution strategy, every non-elite strategy is altered using crossovers, where probability distributions are imported from parent strategies that are also elite strategies. A certain number of strategies from the patroller population were mutated, and when this occurred, certain probability distributions are discarded and replaced with random distributions. It is expected that this patroller strategy will be effective against the varying intruder strategies except for the cautious—non-random intruder strategy. The same could be said for the cautious—random intruder strategy, but because this patroller strategy is so mobile, there is the possibility that patrol routes could cover a large area in the game space to limit intruder exploitation.

The fourth patroller strategy that will be used in this test is the ***non-restrictive Markov distribution patroller strategy***, which is essentially the random probability distribution patroller strategy, altered by GAMMASys' genetic algorithm purposed for non-restricted probability distributions. In this model, the patroller initially travels along the game space according to decisions made from random probability distributions, and as capture rates increase with certain strategies in the patroller population, those strategies gain rank, and the top percentage of strategies with the highest capture rates become elite. The non-elite strategies will be altered

after each generation by means of crossover, mutation, or averaging techniques. A crossover strategy in GAMMASys is one in which the probability distributions for each referenced node will be replaced by one of two parent nodes for each strategy. A mutated strategy in GAMMASys is one in which some referenced nodes in the hexagon are altered through renormalization. An averaged strategy in GAMMASys is one in which all the referenced nodes in the strategy will be renormalized. The setup of a population that will be used by a patroller planning their routes with a non-restrictive Markov probability distribution is that the first 25% of strategies in the population are the elite strategies, the second 25% of strategies in the population are the crossover strategies, the third 25% of strategies in the population are mutated strategies, and the last of the strategies are averaged strategies. These alterations occur in every generation, and hundreds of generations could be executed within each trial.

This patroller strategy can be effective against varying intruder strategies, because non-restricted Markov probability distribution policies allow the patroller to explore for effective strategies. However, since this type of strategy allows the patroller to choose to stay put in a hexagon for a certain amount of time, this patroller may be less mobile than the deterministic probability distribution. We predict that this patroller strategy will be more effective against the more intelligent intruder strategy (the cautious—non-random approach) but less effective against the cautious—random and cautious—non-random approaches, because the lack of mobility causes the patroller using non-restrictive Markov policies to have less coverage than the deterministic probability distribution. Due to the drawbacks of the deterministic probability distribution and the non-restrictive Markov distribution, these two strategies can be designated as more intelligent than the restrictive consistent distribution and the random probability distribution strategies.

To answer this research question, we conduct multiple test cases similar to the test case that was used to answer the first research question. In this new test case we design a 15x15 desert terrain game space can be constructed, and patrollers and intruders can compete using differing strategies to achieve their objectives. Analysis of the test cases shows which patroller strategies are the most effective against which intruder strategies. Also, an overall score can be given to each patroller strategy to determine which strategy is most effective against the varying intruder strategies combined. In order to do this, the averaged -best capture rates across 500 generations in 100 trials are recorded and graphed. Then, the maximum of these scores is recorded and averaged, giving each patroller an overall effectiveness score. The results of these test cases can be seen in Figures 11 (in the previous section), 12, 13, and 14. The effectiveness score comparison is shown in Figure 15.

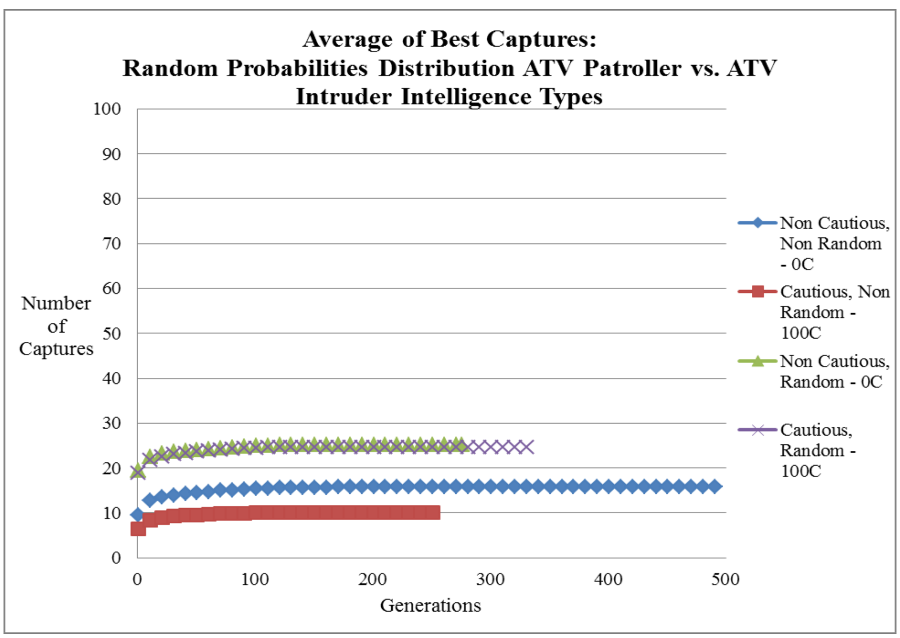

Figure 12: Averaged-best captures between 1 ATV patroller using a random probability distribution strategy, and 1 ATV intruder with 4 different intrusion strategies.

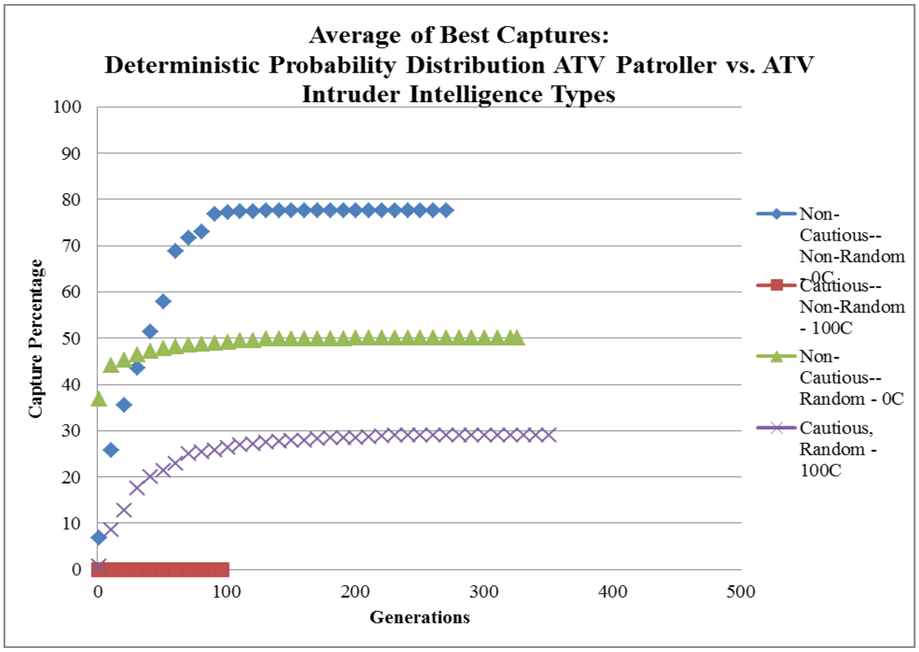

Figure 13: Averaged-best captures between 1 ATV patroller using a random deterministic probability distribution strategy, and 1 ATV intruder with 4 different intrusion strategies.

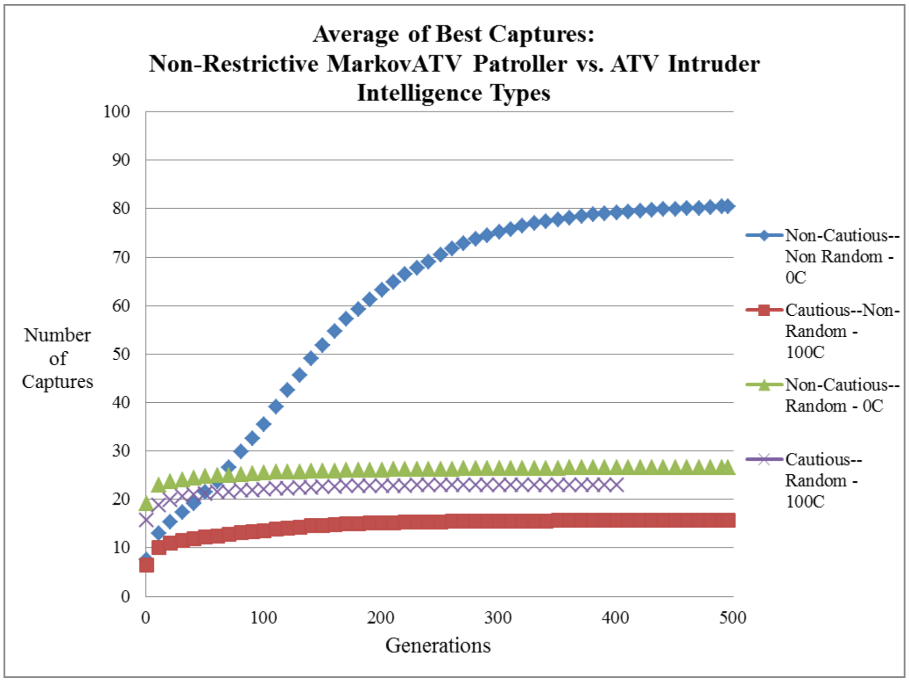

Figure 14: Averaged-best captures between 1 ATV patroller using a non-restrictive Markov strategy, and 1 ATV intruder with 4 different intrusion strategies.

The outcomes of these test cases, in Figures 11, 12, 13, and 14, are represented by four lines which signify the average of the best capture rates against intruder strategies. These test cases involves four separate scenarios, where each patroller strategy competes against a non-cautious—non-random intruder (blue diamond plots), a cautious—non-random intruder (red square plots), a non-cautious—random intruder (green triangle plots), and a cautious—random intruder (purple cross plots). The goal for the adversaries is for the intruder to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and for the patroller to capture the intruder before it reaches its destination.

The genetic algorithm settings for these tests were: 100 trials of tests, 500 generations per trial, trials were cutoff after 100 generations of no improvement (to ensure the trials cutoff only when there is very little potential for improvement). We used a population size of 100 multi-

strategies. The population distribution of strategies was: 25% elite strategies; 25% mutated strategies; 25% crossover strategies; and 25% averaged strategies, where some of the patroller strategies witnessed all, some, or none of these alterations. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on these test cases was at [0,7]. For the non-cautious—non-random and the cautious—non-random intruders, the selection of the source and target nodes was made by the intruder. For the non-cautious—random and the cautious—random intruders, the source-target pair nodes were selected randomly from the 3 source node set and the 3 target node set. The intruders start at a random time step between 0 time steps and 150 time steps.

Based on the results in Figure 11, the intruder strategies can be ranked from most evasive (lowest capture rates) to least evasive (highest capture rates). These intruder strategies were ordered: 1. cautious—non-random; 2. cautious—random; 3. non-cautious—random; 4. non-cautious—non-random. In this test case, cautious—random and non-cautious random had very similar capture rates. The results in Figure 12 were rather different, where the intruder strategies were ordered: 1. cautious—non-random; 2. non-cautious—non-random; 3. cautious—random; 4.non-cautious—random. In this test case, cautious—random and non-cautious—random had very similar capture rates as well. Also, this test case shows that the non-cautious—non-random intruder strategy was fairly evasive, most likely due to the speed in which the intrusion strategy can be executed due to a lower amount of nodes that must be travelled. The results in Figure 13 have different capture rates, but the ordering is similar to the test results of Figure 11. The strategies were ordered: 1. cautious—non-random; 2. cautious—random; 3. non-cautious—random; 4. non-cautious—non-random. This is the only matchup, regarding this research

question, which the capture rates against the random intruder strategies differed significantly. The results in Figure 14 are fairly similar to the results in Figure 11, but with higher capture percentages and lower generation cutoffs. The ordering of the strategies is: 1. cautious—non-random; 2. cautious—random; 3. non-cautious—random; 4. non-cautious—non-random. The random intruder strategies did not differ by more than a 5% capture rate. Besides the matchups between the varying intruder strategies and the random probability distribution patroller strategy, the intruder strategies seem to behave consistently when it comes to evasiveness against a sophisticated patroller.

Table 4: Averaged-best captures between 1 ATV patroller using varying patroller strategies and 1 ATV intruder using varying intrusion strategies.

<table>
<tr><td colspan="6">Sophisticated Adversaries Test: Patrollers using varying patrolling strategies in ATVs vs. intruders using varying intrusion strategies in ATVs.</td></tr>
<tr><td colspan="2" rowspan="2"></td><td colspan="4">Intruder Strategies</td></tr>
<tr><td>Non-Cautious—Non-Random</td><td>Cautious—Non-Random</td><td>Non-Cautious—Random</td><td>Cautious—Random</td></tr>
<tr><td rowspan="4">Maximum of Averaged-Best Captures of Patrolling Strategies</td><td>Restricted Consistent Distribution</td><td>73.26%</td><td>11.16%</td><td>23.36%</td><td>22.43%</td></tr>
<tr><td>Random Probabilities Distribution</td><td>15.89%</td><td>10.15%</td><td>25.22%</td><td>24.71%</td></tr>
<tr><td>Deterministic Probability Distribution</td><td>77.67%</td><td>0%</td><td>50.06%</td><td>29.03%</td></tr>
<tr><td>Non-Restricted Markov Distribution</td><td>80.42%</td><td>15.75%</td><td>26.65%</td><td>22.92%</td></tr>
</table>

In Table 4, which are the maximum of averaged-best capture rates from Figures 11-14, we observe that the non-restricted Markov distribution patroller strategy outperforms the rest of the patrolling strategies with the highest capture rate. The non-restricted Markov distribution patroller strategy outperforms the restricted consistent distribution strategy by a 7.16% difference, outperforms the random probabilities distribution by a 64.53% difference, and outperforms the deterministic probability distribution strategy by a 2.75% difference. Notice that the deterministic probability distribution and non-restricted Markov distribution strategies are within a 3% difference range. Given the different strategies in the patroller populations, the deterministic probability distribution patroller strategy could possibly dominate the non-restricted Markov distribution patroller strategy against a non-cautious—non-random intruder in other test cases.

The non-restricted Markov distribution strategy also outperforms the other patroller strategies against the most evasive intruder, the cautious—non-random intruder strategy. In this matchup, the non-restricted Markov distribution strategy outperforms the restricted consistent distribution strategy by a 4.59% capture rate difference, dominates the random probability distribution strategy by a 5.6% capture rate difference, and completely outperforms the deterministic probability distribution by a 15.75% capture rate difference, where the deterministic strategy received a 0% overall capture rate against the cautious—non-random intruder. However, the deterministic probability distribution patroller strategy was the most effective against the non-cautious—random intruder strategy. This patroller strategy outperforms the restricted consistent patroller strategy by a 27.24% difference, outperformed the

random probabilities distribution strategy by a 24.84% difference, and outperformed the non-restricted Markov distribution strategy by a 23.41% difference. The deterministic strategy was also the most effective patroller strategy against the cautious—random intruder strategy. The deterministic patroller strategy dominated the restricted consistent distribution strategy by a 6.6% difference, dominated the random probability distribution strategy by a 4.32% difference, and dominated the non-restricted Markov distribution strategy by a 6.11% difference. Therefore, we expect that in future matchups with these varying types of adversaries, the non-restricted Markov distribution strategy should be most effective against non-cautious—non-random and cautious—non-random intruders; and the deterministic probability distribution should be most effective against the non-cautious—random and cautious—random intruder strategies. It should also be noted that the random probability distribution strategy performed slightly better than the restricted consistent distribution strategy against the non-cautious—random intruder strategy, and performed slightly better than the restricted consistent distribution strategy and the non-restricted Markov distribution strategy against the cautious—random intruder strategy.

We observe that there are small increases in capture rates against these types of intruders, as seen in Figures 11, 12, and 14, and because all of these patroller populations are generated with random probabilities, it is likely that these patrollers are all operating similarly to a patroller using a random probability distribution strategy, and the inability to be more mobile across the border zone is inhibiting the patroller from creating stronger strategies that cover more area in the zone. In the test cases above, it is likely that the probabilities of the patroller using a random probability distribution strategy allowed for slightly better coverage than the other patrollers using the varying patroller strategies. And in GAMMASys, as well as real-world border regions, there will be limited patroller resources to cover the area.

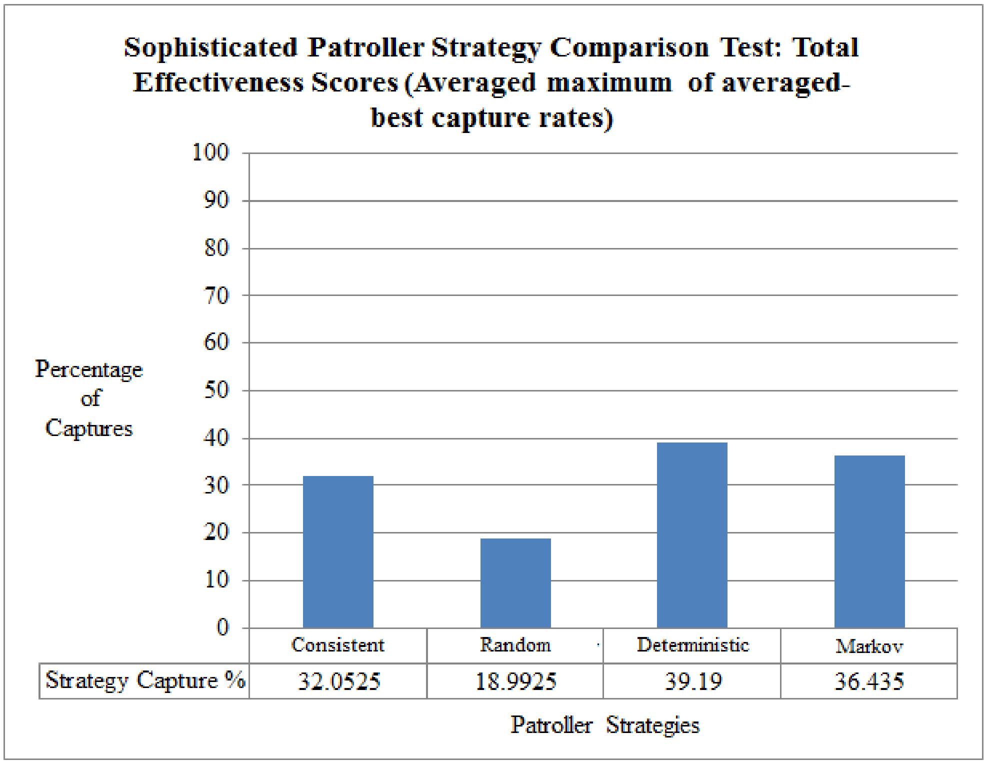

Figure 15: Sophisticated Patroller Strategy Comparison Test with total effectiveness scores: Patrollers using varying strategies on ATVs, and intruders with varying strategies on ATVs.

Figure 15 displays the total effectiveness scores of each patrolling strategy. This score is calculated by averaging all four maximum-of-averaged-best scores across all trials in the test cases. This score represents the expected capture rate a patroller would probably achieve if any intruder strategy were to occur, if the patroller did not know the source/target locations, as well as the intrusion strategy that would be selected by the adversary. According to these results, the most effective patroller strategy is the deterministic probability distribution patroller strategy, with a 39.19% capture effectiveness rating. The least effective patroller strategy is the random probability distribution, with a capture effectiveness rating of 18.99%. It could be proposed that, in order to cover a border zone with the most effectiveness, a border patroller should use a deterministic probability distribution, but it is important to observe that this strategy was the least

effective strategy against the most evasive intruder, the cautious—non-random intruder strategy. Ideally, it would be better for a border patroller to use a non-restricted Markov distribution strategy, where the border patroller would expect 2.76% less captures overall. By using this strategy, the intruder would not be able to exploit a patroller that obtains a 0% capture rate against the cautious—non-random intruder. To improve the non-restricted Markov distribution strategy, the ability to stay still should either be removed, or GAMMASys should force this option to be a very low percentage, so this type of patroller can be more mobile. If this occurs, this patroller strategy has the potential to be dominant overall.

Research Question 2 conclusion: the effectiveness of patrolling strategies varies based on the intrusion strategies their adversaries are using, but the overall effectiveness of the patroller strategy increases as patroller strategies become more intelligent. The non-restricted Markov distribution patroller strategy dominated the non-random intruder strategies, while the deterministic probability distribution strategy dominated the random intruder strategies. Also, the random distribution strategy was not the least effective patroller strategy against some of the varying intruder strategies. We conclude that certain patroller strategies are effective against certain intruder strategies, but no single patroller strategy was completely dominant against all intruder strategies. Overall, the deterministic distribution strategy was the most effective patroller strategy, but its flaw can be seen in that, if an intruder were to use a cautious—non-random intrusion strategy, then it is expected that this patroller will never capture the intruder, if the intruder can observe the patroller strategy during a Monte Carlo simulation. Therefore, in the remaining tests conducted for this research, the non-restricted Markov distribution strategy will be used to test patroller effectiveness. As mentioned previously, the best way to make the non-restricted Markov distribution strategy overly dominant is to limit or

remove the ability for the patroller to wait in a node for a certain amount of time. Even though the random probability distribution strategy was not the least effective strategy against certain intruders, this strategy's effectiveness rating was the lowest, and was nearly doubled by the restricted consistent distribution strategy and the non-restricted Markov distribution strategy, and was doubled by the deterministic probability distribution strategy.

3.3.3. Research Question Three Analysis

The third research question is: can the computational model be used to evaluate and optimize resource allocations for a given zone? In a real world scenario, CBP might not have the resources or funds to maximally cover all borders and coasts of the U.S. in order to stop incoming intrusions. Focusing on resource allocation within border zones is a cost-effective way to cover areas with mobile patrols. Theoretically, the more patrollers that are covering a border zone at a specific time, the higher the capture rate of the patrollers in that given zone over time. It is apparent that a team of patroller types employed for tackling specific terrain should be used in appropriate border zones. For instance, if three patrollers were given plans to cover a specific border zone that consisted of only flat-desert terrain, it would be expected that using three patrollers that have access to ATVs would accumulate the greatest amount of captures. If optimal allocation solutions can be found using the GAMMASys tool, then this tool can be utilized by the CBP and other government organizations for patrol-route planning in a cost-effective manner.

We conducted two tests to indicate whether GAMMASys has the capability of finding optimal patrolling solutions for U.S. border zones. The first test compares the effectiveness of multiple agents being used in a border zone. For instance, patrollers using non-restricted Markov

distribution strategies can be placed against varying intrusion strategies on a uniform terrain, similar to the intelligent-agents matchup tests regarding the second research question. For each matchup, the number of patrollers will be increased by one, for five total matchups. For example, the first matchup will involve a patroller equipped with an ATV, while using a restricted Markov distribution strategy and will matchup against intruders using the random and non-random intrusion strategies. There will be five total matchups, and for every matchup, the number of patrollers will increase by one. It is expected that the outcome of this test will show that: as the number of patrollers increases, so do the capture rates, if the patrollers and intruders are using similar travel means. The individual graphs of the matchups between the patrollers and the varying intruder strategies are shown, but the results regarding the total effectiveness rating of the teams will be presented, along with the results of the maximum of averaged-best captures regarding the patroller versus the varying intelligent intruder strategies matchups.

The test we have constructed involves five teams of patrollers. The first team contains one patroller; the second team contains two patrollers, etc. This is continued with five total teams, and these teams are matched up against the four varying intelligent intruders from the first two research questions. The test involves four separate scenarios. These scenarios include patrollers utilizing non-restrictive Markov distribution strategies which will compete against non-cautious—non-random-, cautious—non-random-, non-cautious—random-, and cautious—random intruders. These matchups reoccur with an increment of one patroller on each team for five total test matchups. The goal for the adversaries is for the intruder to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and while the patrollers' objective is to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials of tests, 500 generations per trial, trials were cutoff after 100 generations of no improvement (to ensure the trials cutoff only when there is very little potential for improvement). We used a population size of 100 multi-strategies. The population distribution of strategies was: 25% elite strategies; 25% mutated strategies; 25% crossover strategies; and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random and the cautious—non-random intruders, the selection of the source and target nodes was made by the intruder. For the non-cautious—random and the cautious—random intruders, the source-target pair nodes was selected randomly from the 3 source node set and the 3 target node set. The intruder starts at a random time step between 0 time steps and 150 time steps. After the tests are complete, the maximum of averaged-best capture rates from each matchup are taken, and a total effectiveness rating is given to each patroller team setup. Figure 16 shows the total effectiveness scores of each patrolling team.

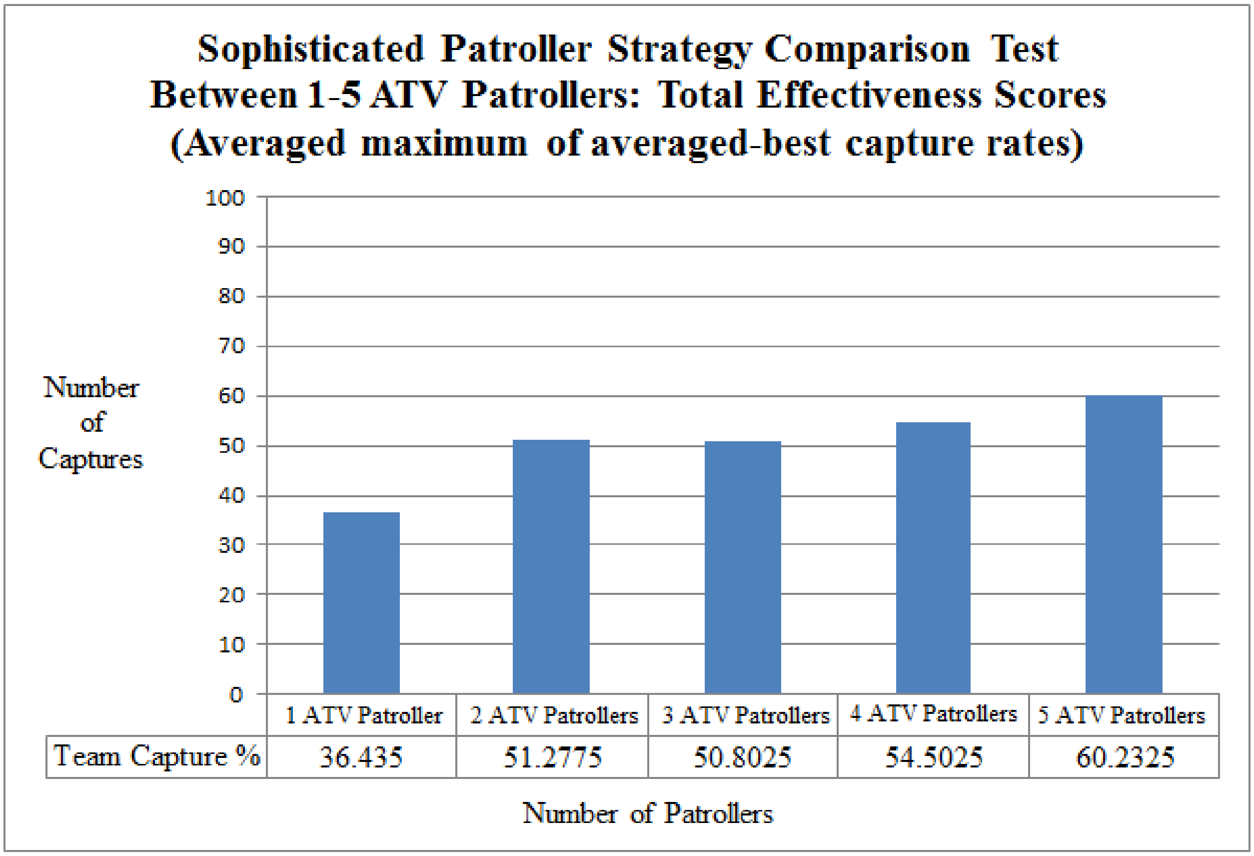

Figure 16: Sophisticated Patroller Strategy Comparison Test between 1-5 ATV patrollers with total effectiveness scores: Patrollers teams on ATVs, and intruders with varying strategies on ATVs.

Overall, the trend of the results is what we expect to see. As the number of patrollers rise, so do the captures. What is important to take away from this is that 2-3 patrollers is only about 10% less effective than 5 patrollers. In real-world situations, that could mean using 2-3 less patrollers at any given time, which could save money, especially because some border regions might see few intrusion attempts. In test cases involving much more patrollers, we still expect the capture rates to increase, but we might see diminishing results. This was not the case in this test, because a lower amount of patrollers were used and some cautious intruders were being used in some of the tests.

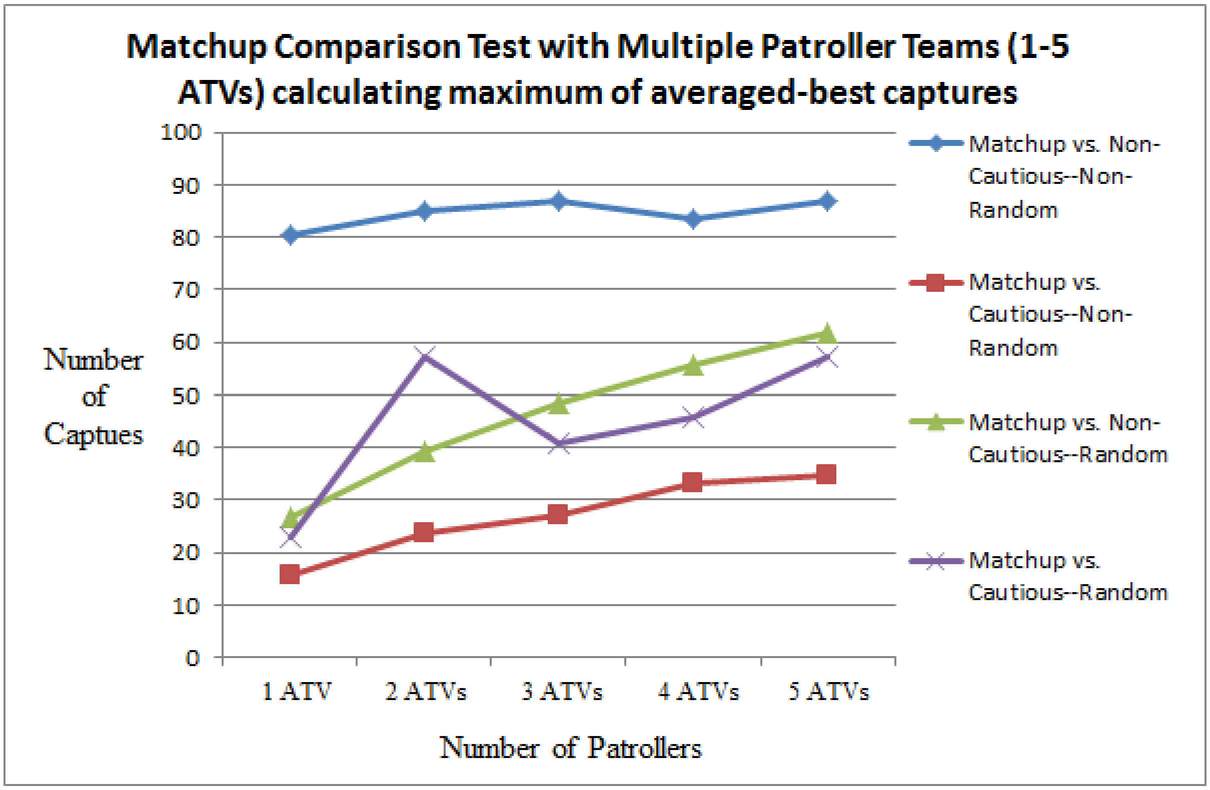

Figure 17: Matchup Comparison Test with Multiple Patroller Teams (1-5 ATVs): Patrollers on ATVs, and intruders with varying strategies on ATVs.

Figure 17 is split up in four categories: how the patroller teams performed against the non-cautious—non-random intruders (blue-diamond plots); how the patroller teams performed against the cautious—non-random intruders (red-square plots); how the patroller teams performed against the non-cautious—random intruders (green-triangle plots); and how the patroller teams performed against the cautious—random intruders (purple-cross plots). It is expected that as the amount of patrollers increase, the effectiveness of the patrolling team should increase as well, resulting in higher capture rates. In Figure 17, we see that the overall trend of the results follows: as the number of ATVs increases, so do the captures.

Future versions of GAMMASys could include features that allow patrollers to communicate and coordinate, as well as a feature that informs the user of patroller effectiveness.

For instance, in the case of the non-cautious—non-random intruder versus the team with four patrollers, the user can be informed of which patrollers are not receiving enough captures. If the first and second patrollers receive fifty percent of the captures, each, the third and fourth patroller strategies can be regenerated and tested to see if their involvement in coverage becomes more active. This can cause issues, however, if these patrollers are competing against more intelligent intruders, because if an intruder observes that all patrollers are executing pursuit routes on their adversaries, the intruder can plan routes to exploit this, and attack on the opposite side of the grid. In Table 5, we see that the team with five ATV patrollers is nearly dominant across all matchups, but in the matchup against the cautious—random intruder, the capture rate is tied with the team with only two patrollers.

Table 5: Averaged-best captures between multiple patrollers (1-5 in ATVs) using non-restricted Markov distribution strategies and 1 ATV intruder using varying intrusion strategies.

Matchup Comparison Test: Multiple Patrollers (1-5 ATVs) vs. varying intruder intelligent strategies in ATVs						
		Number of Patrollers				
		1 ATV Patroller	*2 ATV Patrollers*	*3 ATV Patrollers*	*4 ATV Patrollers*	*5 ATV Patrollers*
Maximum of Average-Best Captures	*Non-Cautious—Non-Random*	80.42	84.84	86.74	83.55	86.89
	Cautious—Non-Random	15.75	23.63	27.26	33.17	34.84
	Non-Cautious—Random	26.65	39.29	48.42	55.58	61.85
	Cautious—Random	22.92	57.35	40.79	45.71	57.35

The second test involves numerous matchups between three-patroller teams which compete on uniform desert terrain against two types of intruders, the most evasive and least evasive intruders. These kinds of matchups make sense since this test does not need to involve all types of intruders, given that interest should be placed on how these patroller teams perform against effective and ineffective patrollers. The types of travel means that will be analyzed in this test are: ATV, car, and foot patrols. So, since all teams will have three patrollers, all combinations of these travel means will be used:

- *Team 1:* 1 ATV, 1 Foot, 1 Car patrols
- *Team 2*: 2 ATV, 1 Car patrols
- *Team 3:* 2 ATV, 1 Foot patrols
- *Team 4:* 2 Car, 1 ATV patrols
- *Team 5*: 2 Car, 1 Foot patrols
- *Team 6:* 2 Foot, 1 ATV patrols
- *Team 7:* 2 Foot, 1 Car patrols
- *Team 8*: 3 ATV patrols
- *Team 9*: 3 Car patrols
- *Team 10:* 3 Foot patrols

All of these teams compete in matchups against non-random intruders, both cautious, and non-cautious, and the maximum of averaged-best captures is recorded. Due to there being ten teams, total effectiveness will not be calculated, because these teams will not be compared against the random intruder strategies. It is expected that since this test will take place on a large, uniform desert terrain, Team 8 will dominate since this team is completely composed of ATVs, which are the most mobile travel units on desert terrain. It is important to mention that this team might not become the most dominant, due to the alterations made by GAMMASys. Since a significant portion of alteration decisions are based on randomization methods, in order for patrol routes to be less predictable to intruders, some teams may have highly effective patrollers on their team, or non-effective patrollers on their team. For instance, a team could score a high amount of capture rates, where only one or two patrollers are making the majority of the captures, while the remaining patrollers are ineffective. Figure 18 presents the results from the ten competing patroller teams matching them up against an intruder using a non-cautious—non-random strategy, and Figure 19 presents the results from these teams matching up against an intruder using a cautious—non-random intruder strategy. These results are calculated as the maximum of averaged-best capture rates.

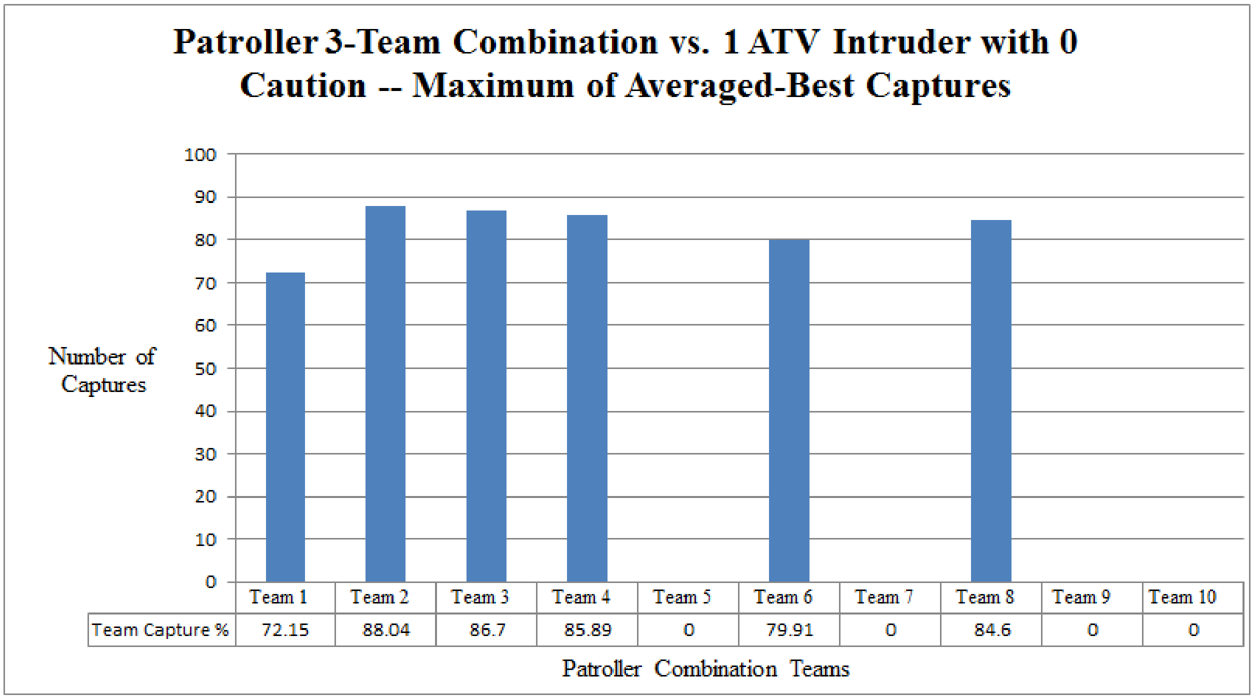

Figure 18: Combination Comparison Test with Multiple Patroller Teams (3-Team Combinations): Patrollers teams on ATVs vs. non-cautious—non-random intruders.

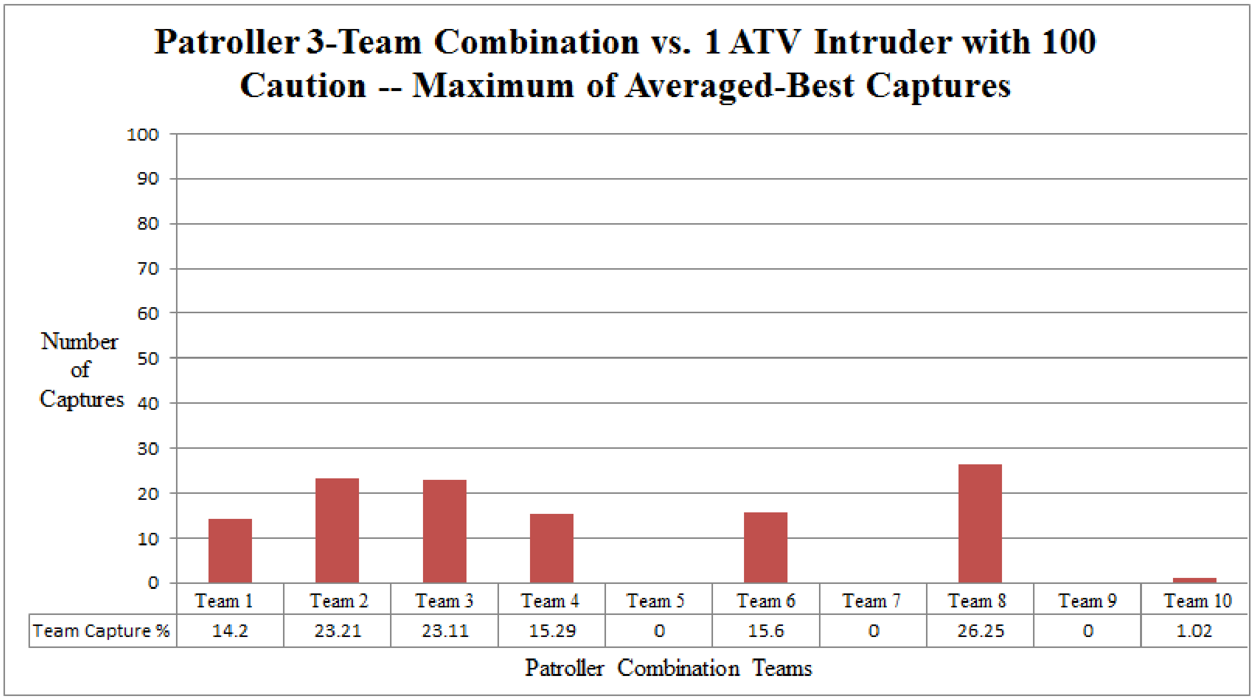

Figure 19: Combination Comparison Test with Multiple Patroller Teams (3-Team Combinations): Patrollers teams on ATVs vs. cautious—non-random intruders.

The combination comparison tests between ten teams of patrollers, included combinations of ATV, car, and foot patrols in a three team setup versus intruders using evasive and non-evasive strategies. The patrolling teams used non-restricted Markov distribution strategies. The goal for the adversaries are for the intruder to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and for the patrollers to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials of tests, 500 generations per trial, trials were cutoff after 100 generations of no improvement (to ensure the trials cutoff only when there is very little potential for improvement). We used a population size of 100 multi-strategies. The population distribution of strategies was: 25% elite strategies; 25% mutated strategies; 25% crossover strategies; and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random and the cautious—non-random intruders, the selection of the source and target nodes was made by the intruder. Intruders start at a random time step between 0 time steps and 150 time steps. After the tests are complete, the maximum of averaged-best capture rates from each matchup are taken.

According to Figures 18 and 19, the most effective teams are Team 2 (2 ATV and 1 Car patrols) and Team 8 (3 ATV patrols), respectively. Team 8 is rather effective against a non-cautious—non-random intruder, but Team 2 might have more effective patrollers, where a third ATV patroller in Team 8 may not be as effective as the ATV patrollers in Team 2. One could argue that the car patroller could be beneficial in the second team, allowing for such a high amount of captures, but on both figures, one can clearly observe that Teams 5, 7, and 9, all

contain cars, and on both test cases, these teams receive no captures. With this information, it can be argued that the ATVs in Team 2 were responsible for most, if not all of the captures in these matchups. Team 10 contains foot patrols, and interestingly, received a 1.02% capture rating against cautious intruders, but a 0% capture rating against non-cautious intruders. As mentioned in the previous test regarding the third research question, future tests should track which patrollers were the most effective team players, and which were not. One way to calculate this is to provide an overall capture percentage of that (individual) patroller. If a team member achieved half of all the captures, their effectiveness rating (the patroller itself) would be a 50% rating.

Research Question 3 Conclusion: GAMMASys does allow for obtaining optimal solutions for effective coverage of border zones, but the results will not always be optimal, and alteration of patrolling strategies might need to occur. By running tests, the user can construct a grid that represents the terrain they are trying to protect from intrusion. This user can set all parameters in GAMMASys that allow the test to represent the scenario they wish to analyze. From here, the user can find out how many patrollers to use, and what type of travel means these patrollers should assume. However, there is no clear cut answer to how many patrollers this user should use, or what type, because there are other factors involved in determining patrolling effectiveness as well. A user can receive a quick solution, but the user must keep in mind that some patroller strategies may not be effective, and some of these strategies must be re-generated. As seen in Figure 17 and Table 5, the team with two patrollers performed fairly well against an intruder using a cautious—random strategy. Also, in Figure 18, the team with two ATVs and one car performed best, even against a team with three ATVs. In some cases, some agents’ strategies may not be well generated, and regenerating these strategies

for this patroller might be beneficial. Saving strong patroller strategies is also recommended. The problem with this practice, however, is that if patrolling teams are molded to effectively cover against a certain intruder strategy, these teams might be vulnerable to exploitation against other intruder strategies. Even though a team with two patrollers rivaled a team with five patrollers, versus an intruder with a cautious—random intrusion strategy, the overall effectiveness score of the team with five ATVs was the highest.

3.3.4. Research Question Four Analysis

The fourth research question is: can the genetic algorithm find effective patrolling strategies fast enough for practical use? Solution quality is important, because border zones need to be effectively covered in order to mitigate exploitative intrusion strategies. However, strong patrolling solutions must be found quickly in order to use in real-world applications. For instance, the five-patroller test seen in Figure 16, versus the non-cautious—non-random intruder took 120 hours to complete (five days). Retrieving a solution in this time period might be unacceptable, because the development of patrolling strategies might need to be reactive based on new intelligence involving intrusion attempts across U.S. borders. GAMMASys operates with a genetic algorithm, and this algorithm has parameters that can be changed, in order for this system to generate solution strategies for several types of zones. Certain parameters could be adjusted in order to find strong patrolling solutions faster. There are four tests that we conducted to determine the right parameters to use for generating a fast and effective patrolling solution. These test cases will be analyzed by solution quality, and time these tests took to reach a certain success threshold (75%, 85%, 95%, and 100% captures) as well as their total time to complete.

The first test involves population size. The more strategies that are being used for the genetic algorithm, the more alterations that might need to occur, such as an alteration distribution is uniform (25% elite, 25% mutation, 25% crossover, 25% averaged). The results of this test can be seen on Figure 20 below.

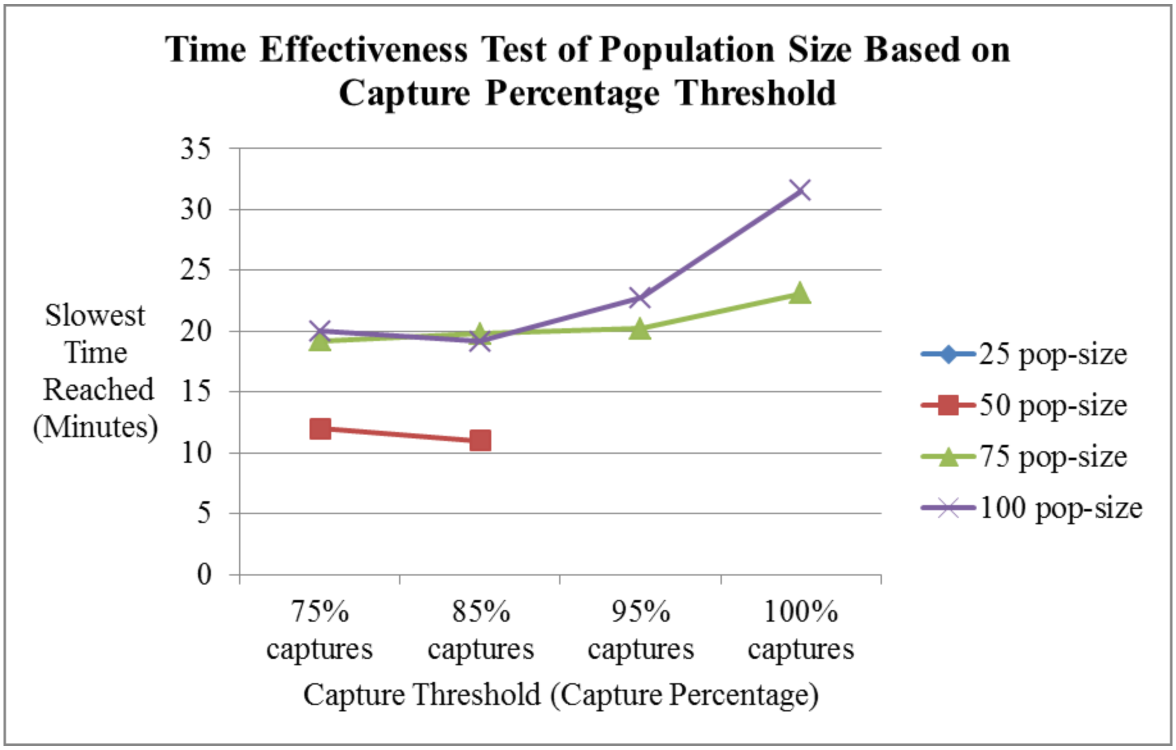

Figure 20: Time Effectiveness Test of Population Size: Patrollers teams with different population sizes on ATVs vs. non-cautious—non-random intruders.

The time effectiveness test of patroller population sizes compares captures thresholds (75%, 85%, 95%, and 100% captures) of patrollers with certain population sizes, (25, 50, 75, 100 strategies) and the slowest times that patroller reaches their thresholds are recorded. Note that in some trials of this test, the patroller may not have reached a threshold. Within each trial, the maximum time that it took the patroller to reach a capture threshold was recorded. If the threshold was never reached in the 100 trial limit, then that plot remained blank. In this test, the

patroller used a non-restricted Markov distribution strategy, and the intruder used a non-cautious—non-random intrusion strategy. The goal for the intruder is to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and the patrollers' objective is to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials of tests, 500 generations per trial, trials were cutoff after 25 generations of no improvement, and contained varying multi-strategy population sizes (25, 50, 75, and 100 strategies). For the population with 25 multi-strategies there was a distribution of 7 elite, 6 mutated, 6 crossover, and 6 averaged strategies which add up to 25 total strategies. For the population with 50 multi strategies there was a distribution of 13 elite, 13 mutated, 12 crossover, and 12 averaged strategies. For the population with 75 multi strategies there was a distribution of 19 elite, 19 mutated, 19 crossover, and 18 averaged strategies. For the population with 100 multi strategies there was a distribution of 25 elite, 25 mutated, 25 crossover, and 25 averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random intruder, the selection of the source and target nodes was made by the intruder. Intruders start at a random time step between 0 time steps and 150 time steps. After the tests are complete, the slowest time stamps of when the capture thresholds were reached were recorded, as was total time of the test.

Based on the results of the test presented by Figure 20, it is best to run a patroller with a population size of 75 patroller multi strategies. Running with a population of 25 multi-strategies took 4.9 hours total, but never reached the 75% capture threshold, let alone an optimal capture percentage. A population with 50 multi strategies performed fairly well, but only reached a

threshold of 85% captures. However, the slowest rate at which a patroller reached the 85% capture threshold was at 10 minutes (based only on the trials that reached this threshold). Now, the patrollers with 75 and 100 multi strategy populations both reached an optimal patrol route (at least one), but the patroller with a population of 75 multi strategies did so at around 24 minutes at its slowest, and the total runtime of the test was 21.10 hours. The patroller with a population of 100 multi strategies reached the optimal solution at around 32 minutes (at its slowest), but the total testing time was 29.51 hours. It makes more sense to run patrollers with a population of 75 multi strategies because this setup reaches the optimal solution faster, and the total testing time is much less.

The second test focuses on the number of trials that should be run per test. The reason for using trials is for a more significant number of comparisons of patroller effectiveness. If the parameter containing the number of trials is increased in the genetic algorithm, more matchups will occur in the test (and more generations that are executed). This allows for better observation of adversarial interaction in the game model. The issue with running too many trials is that they can become time consuming. Ideally, only one trial needs to run, which will contain a certain number of generations, for instance, most tests in this research contain 500 generations. Allowing multiple trials reduces the chance that the potential effectiveness of a patroller will be miscalculated. These test cases will be analyzed by solution quality, and the amount of time these tests took to reach a certain success threshold (75%, 85%, 95%, and 100% captures) as well as their total time to complete. The separation of test cases will include: patrollers estimated with 25 trials, 50 trials, 75 trials, and 100 trials. It is expected that the patroller executing strategies within 25 trials will have the lowest total runtime, but might not reach the optimal solution threshold. Figure 21 contains the results of the time effectiveness test of trial

amounts.

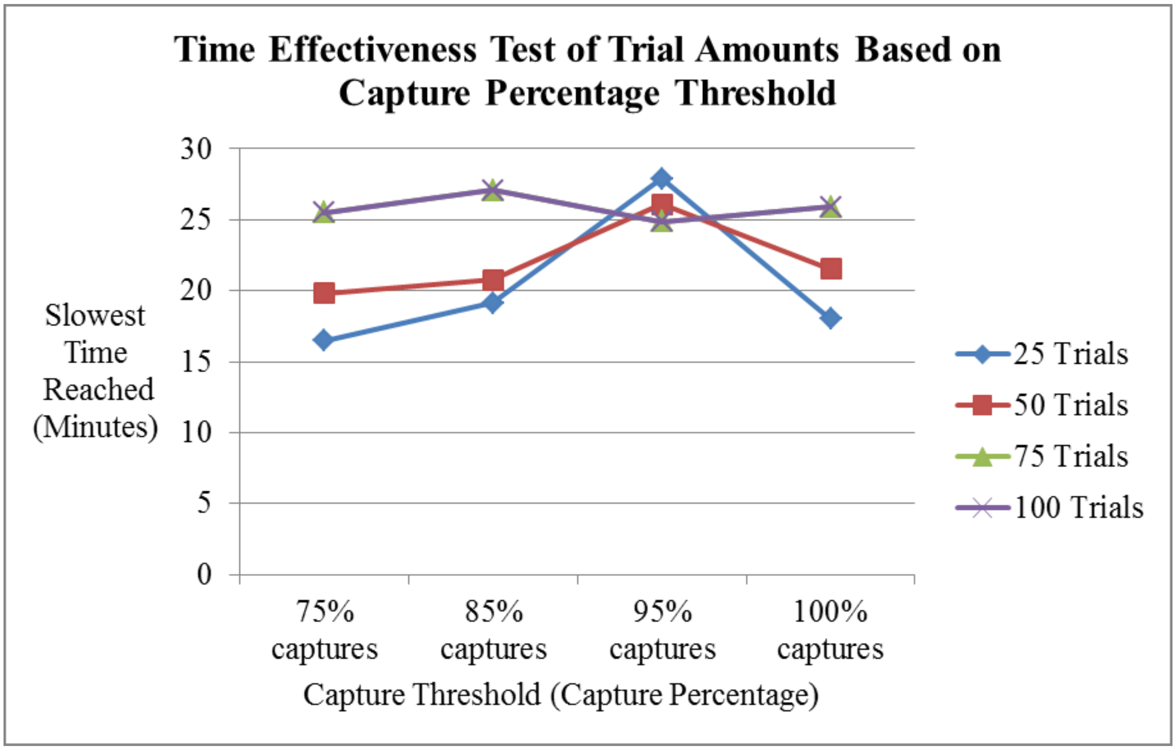

Figure 21: Time Effectiveness Test of Trial Amounts: Patroller teams with different trial amounts on ATVs vs. non-cautious—non-random intruders.

The time effectiveness test of patroller trial amounts compares captures thresholds (75%, 85%, 95%, and 100% captures) of patrollers with certain trial numbers (25, 50, 75, 100 trials executed) with the slowest times that patroller reaches these thresholds. The total time of these tests are then recorded. Within each trial, the maximum time that it took the patroller to reach a capture threshold was recorded. In this test, the patroller used a non-restricted Markov distribution strategy, and the intruder used a non-cautious—non-random intrusion strategy. The goal for the adversaries are for the intruder to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and the patrollers' objective is to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 25 - 100 trials of tests in 25 trial increments, 500 generations per trial, and trials were cutoff after 25 generations of no improvement. The total population of the patroller was 100 multi strategies. The population distribution of strategies was: 25% elite strategies; 25% mutated strategies; 25% crossover strategies; 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random intruder, the selection of the source and target nodes were made by the intruder; intruders start at a random time step between 0 time steps and 150 time steps. After the tests are complete, the slowest time stamps of when the capture thresholds were reached were recorded, as well as the total time of the test.

Figure 21 shows that the best setup for patrollers to use, based on trial amount, should be 25 trials. As expected, the setup with 25 trials was the fastest running matchup, and it did reach the optimal solution threshold. Although the slowest threshold time for a 95% capture rate was the highest, at nearly 27 minutes, the total time of running 25 trials was 8.97 hours, while 50 trials was at 14.45 hours, 75 trials was at 21.56 hours, and 100 trials was at 28.51 hours. We also note that the slowest time for reaching the optimal score of 100% captures was the lowest with the 25 trial setup. It should be reiterated that not all trials reach an optimal score, and in some cases, only one trial might reach this score.

The third test to be constructed is the time effectiveness test of generation cutoff points. The reason for generation cutoff is that in some cases, a patroller may no longer be improving against a certain type of intruder, and if the genetic algorithm is continuing to run, time that could be used for running other tests or for regenerating patrolling strategies is wasted. For this test, there will be five patroller matchups, where the patroller will be cutoff at 25, 50, 75, 100,

and 500 generations (no cutoff) based on non-improvement of capture rates. It is expected that the patroller with a 25-generation cutoff will be the fastest matchup, and will reach the optimal solution threshold. The reason for this is that the adversary for the patroller is the non-cautious—non-random patroller, and there should be plenty of improvement during this matchup. Figure 22 presents the results of these test cases.

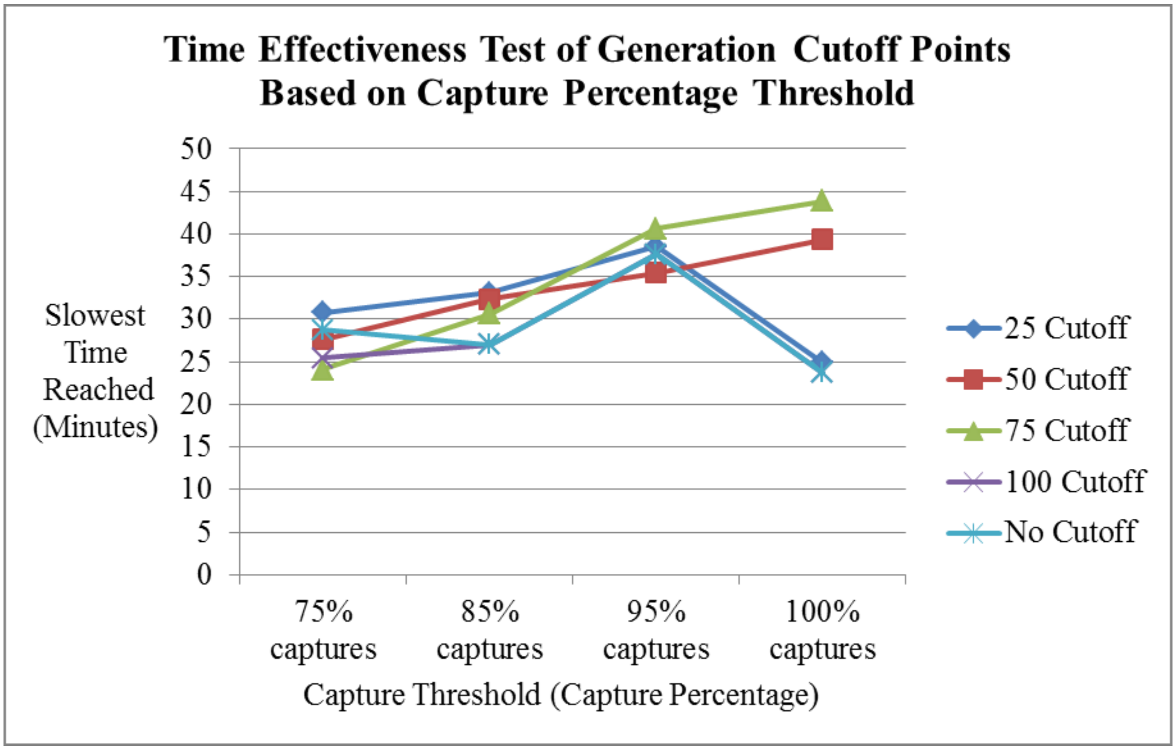

Figure 24: Time Effectiveness Test of Generation Cutoff Points: Patrollers teams with different generation cutoff points on ATVs vs. non-cautious—non-random intruders.

The time effectiveness test of patroller generation cutoff points compares the time-efficiency of patroller setups with different generation cutoff points, used by the genetic algorithm, and records the slowest times that patroller reaches their capture thresholds. In this test, the patroller used a non-restricted Markov distribution strategy, and the intruder used a non-cautious—non-random intrusion strategy. The goal for the adversaries are for the intruder to

reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and the patrollers' objective is to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials, 500 generations per trial, and trials were cutoff after varying generation cutoff points of no improvement (25, 50, 75, 100, 500). The total population of the patroller was 100 multi-strategies. The population distribution of strategies was: 25% elite strategies, 25% mutated strategies, 25% crossover strategies, and 25% averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]. For the non-cautious—non-random intruder, the selection of the source and target nodes was made by the intruder. Intruders start at a random time step between 0 time steps and 150 time steps. After the tests are complete, the slowest time stamps of when the capture thresholds were reached were recorded, as well as the total time of the test.

Based on the results presented in Figure 22, the most effective setup, surprisingly, is the setup with no generation cutoff. In this test, all thresholds were reached by this setup the fastest, although the spread between this setup and the 25 and 100 generation cutoff setups was rather slow (roughly 1 minute or less). However, it would be beneficial to use the 25 generation cutoff anyway, because the total time of the test using this setup was 32.16 hours long, as opposed to the total times of the 100 and 500 generation cutoff setups (35.88 and 35.89 hours, respectively). Considering that the total times will be much less if the patroller were combatting more intelligent intruders, there will be fewer capture improvements because the intruders will be more evasive.

The final test for this research question, the time effectiveness test of elite strategies kept, will compare patrollers that keep a certain amount of their elite population from being discarded (25, 50, 75, and 100 elites). Ideally, it would be important to not have too many elites, because the fewer elites that are used, the more exploratory the genetic algorithm can be to find new effective strategies. For instance, it is expected that the patroller keeping all strategies as elites will not reach the optimal solution. It is possible for this patroller setup to reach the optimal solution, but it is extremely unlikely. The way this type of patroller would work is that all multi strategies would be considered most effective, and there would be no exploration for stronger strategies to be developed. It can be expected that the patroller with 25 elite strategies will dominate in speed and will reach the optimal solution. The test results are presented in Figure 23.

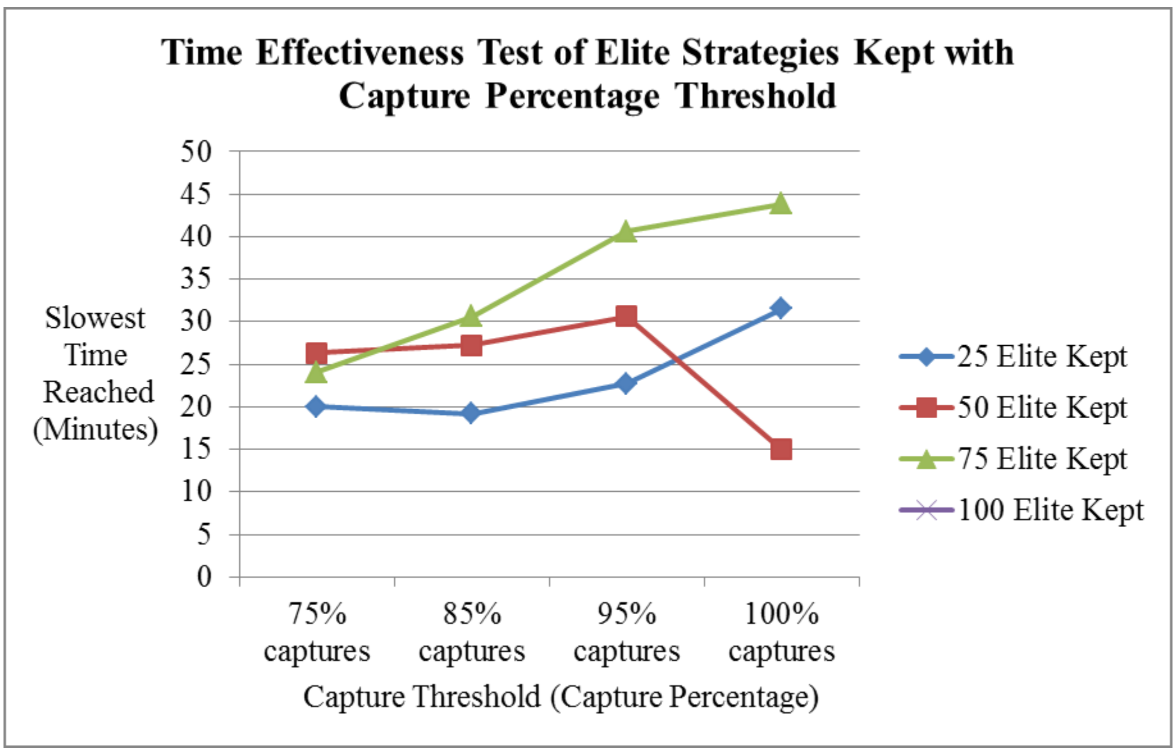

Figure 23: Time Effectiveness Test of Elite Strategies Kept: Patrollers teams with different elite strategy percentages on ATVs vs. non-cautious—non-random intruders.

The time effectiveness test of elite strategies kept compares the time-efficiency of patroller setups with different elite strategy percentages, used by the genetic algorithm, and records the slowest times that patroller reaches their thresholds. In this test, the patroller used a non-restricted Markov distribution strategy, and the intruder used a non-cautious—non-random intrusion strategy. It should be noted that the 100% elite strategy setup did not reach either of the four capture percentage thresholds. The goal for the adversaries are for the intruder to reach its target destination on a uniform desert terrain with 15x15 area space within the game model, and the patrollers' objective is to capture the intruder before it reaches its destination.

The genetic algorithm settings for this test were: 100 trials, 500 generations per trial, and trials were cutoff 25 generations of non-improvement. The total population of the patroller was 100 multi-strategies. The amount of elite strategies varied. For 25% elite strategies, there were 25% mutated strategies, 25% crossover strategies, and 25% averaged strategies; For 50% elite strategies, there were 17% mutated strategies, 17% crossover strategies, and 16% averaged strategies; For 75% elite strategies, there were 9% mutated strategies, 9% crossover strategies, and 9% averaged strategies. For 100% elite strategies, there were no mutated, crossover, or averaged strategies. The node sets were: 3 source nodes at [14,0], [14,7], and [14,14], and 3 target nodes at [0,0], [0,7], and [0,14] on the terrain grid. The starting location for the patroller on this test was at [0,7]; for the non-cautious—non-random intruder, the selection of the source and target nodes were made by the intruder; intruders start at a random time step between 0 time steps and 150 time steps. After the tests are complete, the slowest time stamps of when the capture thresholds were reached were recorded, as well as the total time of the test.

According to Figure 23, the setup that contained 25% elite strategies performs fairly well in achieving the capture percentage thresholds quickly. The 75%, 85%, and 95% thresholds are

reached under 25 minutes each, but the slowest time for reaching an optimal solution was just above 30 minutes. However, the patroller setup preserving 50% elite strategies only performed quickly with the optimal solution threshold. The total testing times for two tests were: 29.51 hours for the 25% elite preservation setup, and 25.14 hours for the 50% elite preservation setup. The problem with the latter is that there is not much room for altering strategies for solution exploration. A better solution would be to use an elite preservation percentage between 25% and 50%. Also, the 100% elite preservation setup did not reach any of the thresholds and took 2.73 hours to complete the entire test. The results would be similar to running 1 generation per trial, but this test takes much longer, and is not recommended.

Research Question 4 Conclusion: GAMMASys does find solutions fast and effectively if the parameters are set up correctly. Each of these tests only focused on one parameter to change, but all parameters can be altered for even faster results. For instance, test one focused on population size, test two focused on trial amounts, test three focused on generation cutoff points, and test four focused on elite strategy amount preservation. Another test could be constructed that focuses on the fast-but-effective setups from each test. However, while these results might not reach the optimal coverage solution, minor tweaks can allow for more efficient genetic algorithm setups. An example test, based on the results would involve the parameters set to include: a patroller using a population of 75 multi strategies; running with 25 trials in the genetic algorithm; a 25 generation cutoff; and using a range of 25% to 50% elite strategies. The mutated, crossover, and averaged strategies should be near equal to each other from here. Future tests could include varying the percentages on these strategy alterations so they are not uniform, in order to test the efficiency of the genetic algorithm setup.

3.4. Multi-Terrain Map Tests

The results of the fourth research question show that the genetic algorithm parameters can be changed in order to generate effective patrolling solutions quickly. We constructed tests, similar to those of research question three, and searched for effective patrolling strategies using a combination of the efficient genetic algorithm parameter settings we found in the fourth research question. We created a map that contains three chokepoint locations where mountainous terrain limits the amount of travel paths from the south to the north of the game-space. By using multi-terrain maps, we can evaluate the effectiveness of the adversaries on more realistic maps.

Figure 24: Chokepoint map which contains seven travel paths from the south to the north. There are two grass paths (green), three paved road paths (gray), and two unpaved road paths (orange). The brown terrain is mountainous and cars and ATVs cannot travel on this type of terrain. The yellow terrain is desert.

Figure 24 depicts the map that new tests will be conducted in. This map contains five terrain types, which is four more than the uniform desert terrain (excluding the red, impassible terrain). The map is more symmetrical, allowing the patroller and intruder to use the map to their advantages. For instance, patrollers can block intruder entry through one of the seven paths. If the intruder is traveling through the path that the patroller is blocking (e.g., the middle, paved-road path), then the patroller will most likely capture that intruder. However, if a patroller travels through one of the paths, the intruder can choose a different path, and the patroller will have to take extra path steps in order to capture the intruder.

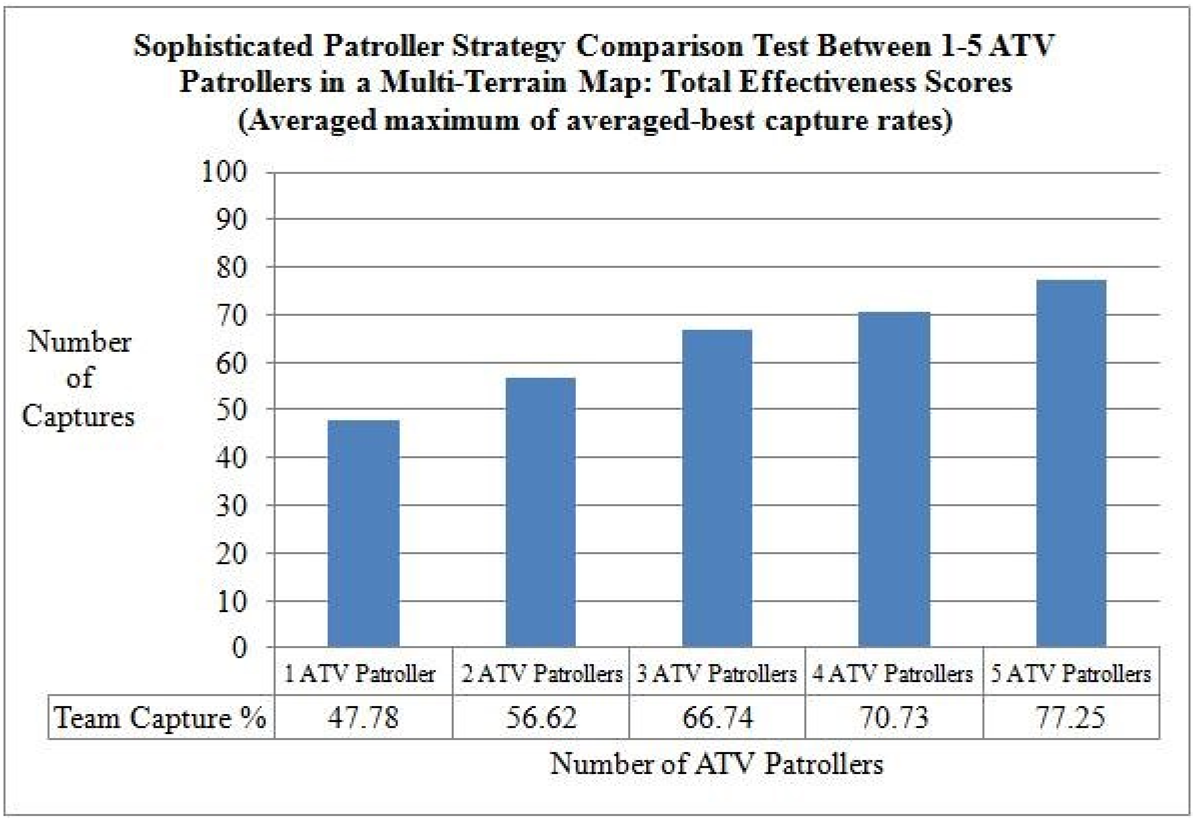

Figure 25: Sophisticated Patroller Strategy Comparison Test between 1-5 ATV patrollers with total effectiveness scores on a multi-terrain map: Patrollers teams on ATVs, and intruders with varying strategies on ATVs.

Figure 25 shows the total effectiveness scores of the five patroller teams using ATVs to patrol the chokepoint map. These patrollers were matched up against the four types of intruders that were introduced in the first research question. Similar to the first test case in the third research question, the trend of captures increases as the number of patroller increase. On the multi-terrain map, the overall number of captures has increased, as opposed to the results shown in Figure 16. We show that using five ATV patrollers to cover the chokepoint map only results in ten percent more captures than use three ATV patrollers. This is important for resource allocation purposes, if funding and resources are limited.

We changed the genetic algorithm parameters, due to the results we observed when analyzing the fourth research question, and found that the average total time of all tests involving 5 ATVs in research question three took 132 hours. The average total time of all tests involving 5 ATVs, where the genetic algorithm parameters have been changed to save computation time, was 13 hours. By changing the four parameters in the genetic algorithm (population size, trial numbers, generation cutoff point, and elite percentage), we were able to save 119 hours of computation time, nearly five days of searching for effective patrolling solutions. Similarly, if a decision-maker chose to use three ATV patrollers for coverage, then the average total computation time would have taken roughly 8 hours.

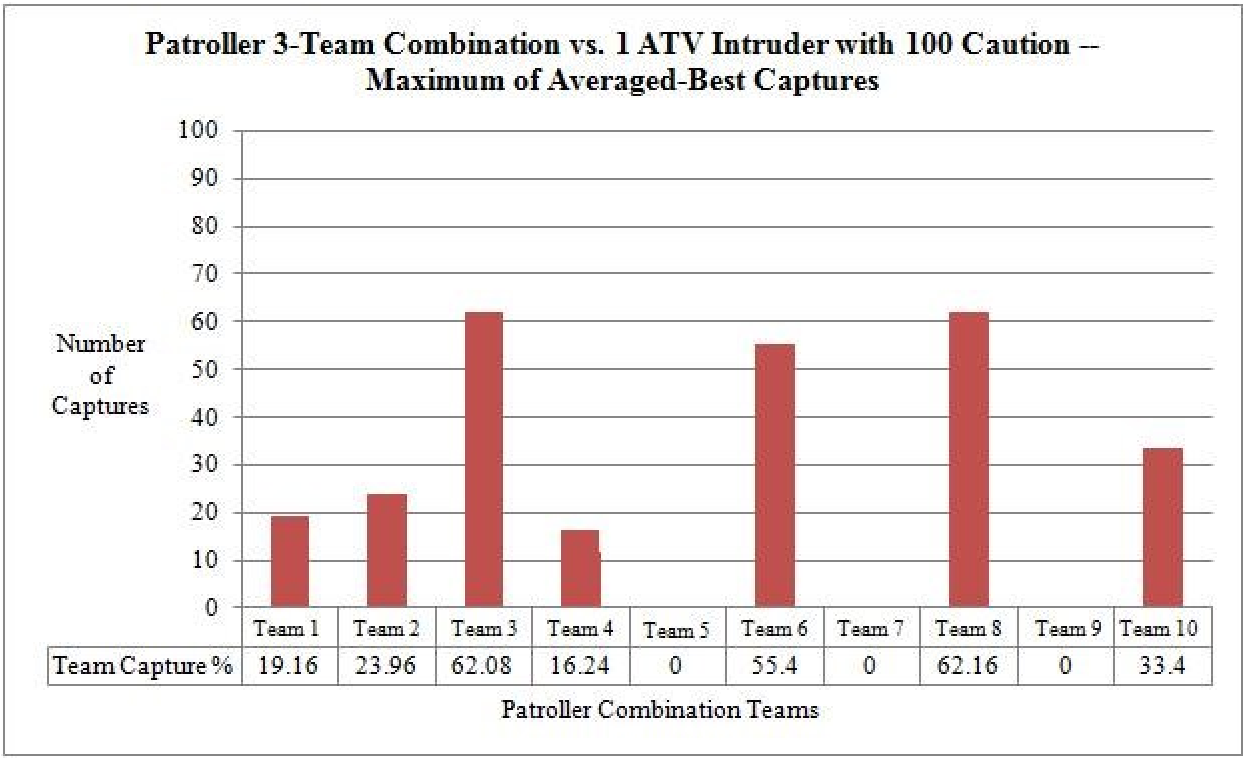

Figure 26: Combination Comparison Test with Multiple Patroller Teams (3-Team Combinations): Patrollers teams on ATVs vs. cautious—non-random intruders.

In Figure 26, we show team capture rates of patroller groups with three agents. These teams contain all combinations of ATV, car, and foot personnel. The teams are the same as those shown in Figures 18 and 19. We are only showing the results of the patroller combination teams against cautious—non-random intruders (most evasive intruders), because we want to analyze the effectiveness of sophisticated adversaries only. Figure 26 shows that the highest performing patrolling team was Team 8, which contains three ATV patrollers. However, Teams 3 and 6 have high capture rates and contain only 2 ATVs and 1 ATV, respectively. Team 10's capture rate is half of Team 8, and contains all foot patrols. This is important because Team 10 contains no ATV patrollers and consists of only foot patrols. This is likely because foot patrols in GAMMASys are able to travel across mountainous terrain, while ATV and car patrols cannot. This is important for resource allocation, because in real-world scenarios, foot patrols may have

a considerably lower cost than ATV patrols. In future tests, GAMMASys might allow ATVs to travel across mountainous terrain. In these tests, we also changed the genetic algorithm parameters in order to save computation time. Since we only matched each team against a cautious—non-random intruder, and did not matchup the teams against the other three intruder strategies, we just took the total time of the most effective patrolling team (3 ATV patrollers). The total time of Team 8 was 7 hours. In the team combination test in research question three, the total time to complete evaluation of the 3 ATV patroller team took 53 hours, or 2.20 days.

We conclude that changing the genetic algorithm parameters, by lowering the population size, total trial numbers, and the generation cutoff point; and raising the amount of elite strategies in the population, the total computation time will lower. Also, by generating patrolling strategies for terrain with restricted travel can result in higher capture rates. **If travel along a game-space is more restricted, lower amounts of patrollers will need to be used to effectively cover the border region. If the game-space is less restricted, more patrollers will need to be used to effectively cover the border region.**

Chapter 4: Conclusion

4.1. Related Work

There have been several experiments and studies that focus on resource allocation, applying game-theoretical methods for patrolling, and using algorithms such as a genetic algorithm to use exhaustive measures for developing effective strategies for game players to execute during game simulation. This concluding section of this study outlines some of these projects.

In terms of automated patrolling using robots, Marino et al. ran an experiment where patrolling and intruding robots would be placed in a location filled with obstacles, and the robots would interact in an adversarial environment (Marino e.t. al., 2013).[11] The patrolling robots, similar to the patrolling agents in GAMMASys, were given the objective to capture the intruding robots by placing themselves within a set distance near its adversary, which is similar to another study on automated robotic patrollers conducted by Basilico et al. (Basilico et al., 2009).[12] The intruder robots' objective is to reach a targeted location without being detected or captured. The difference between the robot patrolling study and this research project is that the patrolling robots use a ***swarm*** approach, where the patrollers can communicate amongst themselves.[10,11] GAMMASys does not allow communication between patrollers, because the patrollers cannot detect an intruder if the intruding agent is at a different ***node***, a location on the game space that can be visited by an agent. Since randomized probabilities are used for the patroller's path to be created, the location of the intruder does not alter the probabilities of travel, therefore, the patroller does not need to communicate with other patrollers about the location of the intruder, unless the intruder is captured.

Basilico's research project allowed the intruding robot is able to observe the initial movements of the patrolling robots, much like the patrolling agents in GAMMASys. In one experiment, the system that is used to develop effective patrol strategies limits the amount of time to create these paths (Basilico et al, 2009b). The system is set to stop computations if the program has executed path generation for ten hours.[12] While GAMMASys is similar, instead, trials (multiple trials consisting of many generations and simulations) are halted if interdiction rates do not improve after a set number of generation.

There is another study that has been conducted that focus on patrolling strategies using game theory, but not using autonomous robots, such as the research project by Nicola Gatti from the Artificial Intelligence Laboratory in Milano, Italy (Gatti, 2008). This study used simulation tools similar to GAMMASys, and modeled their games using adversarial interactions like the patroller/intruder model. In this research project, the approach to setting up a game for simulation was to model it as a Bayesian Game, where players of the game can belong to one or more agent types (such as patroller and intruder), and the payoff of achieving that player's objective is based off that unit's agent type. The objective of the intruding agent in this model is to intrude in a targeted house on the game space, while the patrolling agent patrols a number of houses in order to prevent an intruder from smuggling goods from a targeted house.[17] This is similar to GAMMASys, because the intruder is given a source node, the location where the intruder starts at the beginning of the game, and a set of target nodes, where the intruder can choose between the set for the location it wants to target. In turn, the patroller scouts throughout the border zone in order to prevent the intruder from reaching its targeted node.

[17] Gatti, Nicola. "Game Theoretical Insights in Strategic Patrolling: Model and Algorithm in Normal-Form." *ECAI-08* (2008): 403-407.

Bassilico and Gatti worked together on a research project with Amigoni, where they used a simulation tool to search for an optimal patrolling solution that was computationally demanding. The experiments were conducted using a simulation system called USARSim and the patrolling solutions were generated by the BGA Model. Their conclusions were that the optimal patrolling solutions generated by the BGA Model outperformed other patrolling strategies that were not as computationally demanding (Amogini, 2010). This research project is similar to GAMMASys, because our system uses simulation tools that evaluate patroller strategies after they are generated, however, our simulation system generates patrolling strategies, and we do not use a separate model or system to do so. However, the simulation system was developed based our game theoretic model that we constructed. These researchers also published another research paper titled "Developing a Deterministic Patrolling Strategy for Security Agents," where they propose that artificial patrollers in a simulation system could use a deterministic patrolling strategy to conveniently detect intrusions (Basilico et al., 2009a). The results of this research show that patrollers should execute deterministic cyclic paths in order to effectively detect intrusions on a graph and ensuring intruder capture. This research is relevant to this thesis, because we have developed patroller strategies that use deterministic cyclic paths in order to capture intruders. We show that our deterministic patroller strategy was very effective against all types of intruders except for the cautious—non-random intruder.

Chevaleyre et al. analyze multi-agent patrolling strategies and proposed that groups of agents can be used to perform patrolling tasks in varying domains, such as computer networks or in sub-domains of computer war games. These researchers also claim that a strategy is not good, unless it minimizes the time to travel in between nodes on a patrol route in a simulation. The research project focuses on assessing whether algorithms that generate near-optimal strategies

are efficient. This research showed that as long as the graph contains small travel distances between nodes, cyclic patrol paths will outperform graph partitioned patrolling strategies. These researchers also tested the efficiency of the cyclic patrol paths against other patrolling strategies, and the cyclic routes were just as effective and efficient as the "state-of-the-art" patrolling strategies (Chevalyre et al., 2004). This research is relevant, because GAMMASys has the capability of using deterministic patroller strategies, which are cyclic by nature.

A research project constructed by Garnaev is similar to our research project, but Garnaev's model is related to terrorists infiltrating a guarded area (Garnaev, 1997). Garnaev constructed a two-player zero-sum game, a game where only one player can win, that takes place on a graph, which is similar, but less complex than the hexagonal grid on GAMMASys. There are starting and ending locations on the graph for the infiltrator, and the game must be completed within a certain number of time steps, similar to the players' objectives in GAMMASys' game theoretic model. Captures occur when the "guard" and "infiltrator" travel to the same location at the same time. In Garnaev's model, there are safe nodes, where infiltrators can travel to, without being captured. GAMMASys does not directly have a similar feature, except if the patrollers and intruders are using different travel methods, and the intruder travels on terrain that the patroller cannot reach, or travel on. In Garnaev's model, both players keep track of their past actions. This is not done in GAMMASys, because of the size of the game-space, and the amount of time steps that are allowed during a simulation.

Jakob et al. from the Czech Technical University created a research project that uses coordinating and planning techniques to suppress maritime piracy. An agent-based simulation of maritime traffic was used to evaluate scheduling and patrolling strategies. With the use of environmental data, vessel behavioral data, and vessel behavioral characteristics, plans can be

developed to either deter or evade pirate attacks (Jakob e. al., 2011). This system uses Google Earth as a user interface, which is similar to GAMMASys, because the hexagonal grid on our system is placed over Google Maps satellite pictures. This research project on transit games is relevant to our research, because evasive strategies are developed for intruders in GAMMASys, in order to generate formidable intrusion strategies. In turn, we can generate adaptive patrolling strategies.

Two other simulation tools have been created for police patrolling of "hot spots," locations that have a high frequency of crime in the past, to deter future crime. These experiments were carried out by Melo et al. in 2005 and Chawathe in 2007. One of the experiments uses geographic information systems to map criminal activities for the user, or police official, to observe and plan patrolling operations. Much like this patrolling experiment, GAMMASys uses satellite photos from Google Maps, to help the user or analyst plan patrolling operations at specific border zones. This experiment allows the patrolling agent to behave similar to the autonomous robots with the ability to communicate, or to be independent of their other patrolling agents, like those in GAMMASys. The former strategy in this experiment is called a ***shared strategy***, while the patrollers in GAMMASys use ***independent strategies***. Like GAMMASys, this simulation tool developed patrol routes with systematic randomness, in order to inject uncertainty in the intruder's decision process. This experiment also uses a type of route known as a critical route, which weighs importance of several locations on the map in order for the intruder to make sure they visit that node within their path to increase protection on vital locations (Melo et al., 2005).[18] The second experiment with police patrolling is similar to GAMMASys, in which it uses vehicle-based agents (in patrol cars) and is focused on road

[18] Melo, Adriano, Mairon Belchior, and Vasco Furtado. "Analyzing Police Patrol Routes with the Simulation of the Physical Reorganization of Agents." (2005).

networks of a police district sized domain. This experiment is different in the sense that the game space is set up as a graph with road networks, but topology is ignored. The visualization tool in GAMMASys contains topology and road networks (Chawathe, 2007).

Even though this research is focused on developing effective patrolling strategies to increase interdiction rates between POEs, resource allocation is also relevant to developing these strategies and is an objective of the USBP in their latest national strategy. RAND Corp (Research and Development Corporation) has conducted a study that focuses on resource allocation to better deploy personnel, equipment, and finances to strategic zones along the U.S. borders. This study concluded that the best way to allocate resources between border zones was to couple pattern/trend analysis with systematic randomness. What this means is that personnel, equipment, and finances will be deployed to a set of zones that are frequently crossed by intruders, but personnel can also be randomly deployed to any border zone to create uncertainty in the intruder crossing strategy (Predd et al., 2012). Similar to our research project, the RAND study focused on competing with expensive alternatives, and concluded that the interdiction rates of this type of approach could nearly match those strategies that used a higher amount of resources. This study on resource allocation suggests that relative measures, such as the coverage of the border are more important than the size of the border, or the amount of intruder that are attempting the cross into the U.S. GAMMASys simulations allow the intruder to observe a set number of moves that the patroller makes before the intruder can generate its path to its goal location. This allows the intruder to be cautious. The RAND study does not use this feature, and does not allow the intruder to observe the decisions of the patroller.[5] While the study conducted by RAND is not the main focus of this research, resource allocation plays an important role in developing effective patrolling strategies, because it allows the analyst that is

running simulations to determine which zone will be represented as the game space in GAMMASys (the location the game will be simulated), how many patrollers to create, what equipment will be used, and their means of travel (vehicle types).

Other systems have been developed with resource allocation in mind with a focus of national security that benefits the Transportation Security Administration (TSA) and U.S. Coast Guard. These systems are GUARDS (Pita et al., 2011), PROTECT (An et al., 2011), ARMOR (Jain et al., 2010) and IRIS (Tsai et al., 2009). GUARDS focuses on using resource allocation techniques to assist in the defense of over 400 airports in the U.S.[19] PROTECT focuses on maritime defense at ports, coasts, and inland waterways, using resource allocation and patrolling strategies to protect from influx of criminal activities and terrorism from waterside entry.[20] ARMOR focuses on using randomization to deploy personnel and canine units to different checkpoints and roads at airports for security purposes.[21] IRIS randomizes flight schedules for Federal Air Marshalls to board international flights to protect against aerial hijacking.[22]

Some research projects have developed genetic algorithms in order to develop solutions for complex problems. Ahn & Ramakrishna, members of IEEE, created a genetic algorithm for shortest path routing problems and sizing populations of strategies (Ahn & Ramakrishna, 2002).

---

[19] Pita, James, Milind Tambe, Chris Kiekintveld, Shane Cullen, and Erin Steigerwald. "Guards - Game Theoretic Security Allocation on a National Scale." In *Proc. of The 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, vol. 3. 2011.

[20] An, Bo, James Pita, Eric Shieh, Milind Tambe, Chris Kiekintveld, and Janusz Marecki. "Guards and Protect: Next Generation Applications of Security Games." *ACM SIGecom Exchanges* 10, no. 1 (2011): 31-34.

[21] Jain, Manish, Jason Tsai, James Pita, Christopher Kiekintveld, Shyamsunder Rathi, Milind Tambe, and Fernando Ordóñez. "Software Assistants for Randomized Patrol Planning for the LAX Airport Police and the Federal Air Marshal Service." *Interfaces* 40, no. 4 (2010): 267-290.

[22] Tsai, Jason, Christopher Kiekintveld, Fernando Ordonez, Milind Tambe, and Shyamsunder Rathi. "IRIS-a tool for strategic security allocation in transportation networks." (2009).

Instead of using patrolling strategies, like GAMMASys, these researchers used strings to represent chromosomes, and genetic algorithm parameters to represent genes. GAMMASys uses patrolling strategies to represent members of the population, and contains genetic algorithm parameters that can be altered to change the way the genetic algorithm operates. The genetic algorithm that was created by Ahn & Ramakrishna also uses crossover and mutation to modify the population in order to search for better paths. GAMMASys also uses mutation and crossover techniques for modifying the population of patroller strategies. Ahn & Ramakrishna's research is relevant to ours, because the use of a genetic algorithm as a search technique allows us to identify effective patrolling strategies, but one drawback to our research has been long computations times, partly due to large population sizes.

Tormos et al. also developed a genetic algorithm, in order to solve complex real-world issues. These researchers tasked themselves with using an evolutionary algorithm to solve the complex train timetabling problem, where they try to optimize train time tables on a single line track. It is difficult to search for good solutions for this real-world problem in a reasonable amount of time for scheduling purposes. The genetic algorithm that these researchers created generates a schedule for new trains. This research project found that the genetic algorithm used for the train timetabling problem outperformed other train scheduling techniques, and certain modifications to the algorithm can improve performance (Tormos et al., 2008). This research is relevant to GAMMASys, because future work will involve comparing our genetic algorithm with other patrol path generation methods in order to assess the efficiency of our genetic algorithm and patrol routes.

Based on some of the studies that have been outlined in this section, Aguirre et al. developed a decision support tool to allocate resources for border security. The patrol route that

is produced from this system is based on multi-objective optimization. In this sense, the patroller must weigh its objectives and its options in order to travel along an optimized patrol route. This system uses intruders and patrollers as well and the game space symbolizes a region of the border. The game space is typically made of road networks, and solutions become less effective as the size of the game space increases. This system also uses a genetic algorithm and the fitness is evaluated by number of captures per patrolling strategy (Aguirre et al., 2011).[1]

4.2. Conclusions and Future Work

This research has proposed four contributions, and three research questions. The research questions are related to the contributions of this system. In order to ensure that the proper tests are constructed, conducted, and evaluated, certain enhancements and features can be implemented into GAMMASys to better ensure that real-world applications of this system will be beneficial. These improvements are not ranked by importance.

The first improvement can be for intruder behavior and intelligence: non-cautious—random and cautious—random intruders rarely use long paths based on source/target nodes pairs. As mentioned in the analysis accompanying the first and second research questions, the random intruders are only given nine possible routes, where two in particular are not very effective, because of these long routes. These routes cause the intruders to cross the grid from bottom left to top right, or bottom right to top left of the grid, greatly increasing the amount of hexagons that must be travelled. By conducting these routes nearly 22% of the time (2 out of 9 routes), the intruders become extremely vulnerable. A possible fix for further tests to make random intruders more formidable is to allow for a set that utilized the whole 15th row for source nodes, and the 1st row for target nodes. Then, before the simulations

begin, the intruder can randomly choose its routes from their most effective intrusion decisions, and very rarely choose routes that are much longer. For instance, the most effective routes can have a higher probability to be travelled, while non-effective routes have a very low, or zero probability of being travelled. This can make random intruders more evasive and unpredictable.

The second improvement can be towards patroller decision-making: restricted consistent distribution and non-restricted Markov distribution strategies can have an option to become more mobile. The patroller can have an option that either greatly decreases or removes the probability for these types of patrollers to stay still at any given time step. As seen in the analysis of the second research question, a possible reason for the deterministic probability distribution patroller strategy's success was because of their mobility. This can be seen where the patroller was most effective against the random intruder strategies, where they could change their attack locations to the other side of the grid. This feature should be able to be turned on or off, for aggressive and passive patrolling strategies.

The third improvement for the system can be towards more realistic tests with regards to the hexagonal grid of GAMMASys. For instance, tests can be constructed that test patroller team efficiency and effectiveness on more realistic terrain. Maps from real-world zones can be used to create a terrain map, and patroller teams can be pit against intelligent intruders. These results could be used to create patrol routes in real-world scenarios soon after. It is expected that teams will have to vary their means of travel, as some maps will have many types of terrain.

The fourth improvement can be for the terrain weights on the hexagonal grid in GAMMASys. By improving the accuracy of the terrain weights, the model becomes more

realistic. The terrain weights work in accordance with vehicle travel speed. In a sense, if a car travels five times faster than a human on paved road (may not be accurate; just an example), at cruising speed, then this must be represented in the model. Some of the vehicle types in the computational model have not been thoroughly tested, so these vehicles will have to be included as well. Improving terrain weight accuracy is relevant to approaching all three research questions.

The fifth improvement can benefit the efficiency of the genetic algorithm, in terms of parallelization and code optimization. In order to allow for the genetic algorithm to be more efficient, redundant and unnecessary functions within the algorithm must be removed or restated. The genetic algorithm does cause the system to run slow, which is why tests were constructed to find better parameter settings that are effective and time efficient. However, there are some programming practices that can make GAMMASys run faster, such as parallelization or code optimization. Parallelization can allow multiple simulations to be run at the same time, instead of sequentially, which should speed up the system greatly. Code optimization can get rid of unnecessary methods or commands that slow down the system.

The sixth improvement can benefit the game model and how simulations are conducted: increase the amount of intruders that are allowed in a single simulation. By allowing multiple intruders in a simulation, which can use multiple strategies and execute at different time steps, the patroller strategies can be graded with better effectiveness. In order to do this, the simulation method within GAMMASys must allow games to carry on, even after one intruder is captured. Also, a new scoring system can be implemented as well. All patrollers currently received 1 point for a capture, and 0 points for an intrusion. Future grading can give patrollers 2 or more points for a capture, 1 point for a coverage assist, 0 points for an intrusion,

and maybe even negative points for regret. By doing this, more intelligent team patrolling strategies can be created. Also, by keeping tracking of these captures, another improvement can be implemented: **increasing the accountability of patrolling units**. By increasing the accountability of patrolling units, users might be able to discern non-effective patrollers to effective patrollers from the system. These improvements can also allow for another improvement to be implemented: **allowing for communication between patrollers**. This will allow patrollers to use a swarm strategy, where some patrollers will use pursuit routes that prevent the intruder from exploiting the other patrollers by taking a different route than what is expected. Also, this can prevent patroller travelling redundancy, where patrollers travel to the same node, wasting time and making coverage less effective.

The seventh improvement can benefit real-world strategy generation preparation: including sensor and other travel units. Other units can be implemented, such as ground sensors, unmanned aerial vehicles, and helicopters. All these units could be sensors, and when an intruder is spotted, their location can be communicated to the patrollers on the grid, and these patrollers can take shortest paths to the sensor, or use pursuit routes to contain the intruder.

The eighth improvement could benefit resource allocation practices: the user is given a funding amount, and can buy units for protection. This feature might allow for more realism, where the user has a budget and can buy a variety of patrollers and place them on locations on the map, and GAMMASys can search for the optimal or near-optimal solution on the map. This could be very beneficial for real-world patrol planning.

The ninth improvement could benefit the observation of human decision-making: where GAMMASys could allow users to play against the patroller or intruder. The

problem with AI agents is that they do not act similarly to humans. By allow humans to interact with the agents provided by GAMMASys, better patrolling decisions could be constructed.

The tenth improvement could benefit GAMMASys applicability: constructing capabilities for test construction and altering the game model rules for search and rescue patrol routes. Currently, the game model ends a simulation when a capture occurs, the intruder runs out of time steps, or when the intruder reaches their target zone. If the amount of time steps allowed is increased, a user could place still or slow moving intruders that are not using shortest path algorithms, and patrollers can be tasked with traversing the hexagonal grid until all agents are found and rescued. These "intruders" would have to be labeled as evading or hiding agents.

Bibliography

Aguirre, Oswaldo; Lopez, Nicholas; Gutierrez, Eric; Toboada, Heidi; Espiritu, Jose; Kiekintveld, Christopher. "Towards the Integration of Multi-Attribute Optimization and Game Theory for Border Security Patrolling Strategies." *Applied Adversarial Reasoning and Risk Modeling: Papers from the 2011 AAAI Workshop* (2011).

Ahn, Chang Wook, and Rudrapatna S. Ramakrishna. "A genetic algorithm for shortest path routing problem and the sizing of populations." *Evolutionary Computation, IEEE Transactions on* 6, no. 6 (2002): 566-579.

Alden, Edward. "Immigration and Border Control." *CATO Journal* 32, no. 1 (Winter 2012): 107-124. *Academic Search Complete*, EBSCO*host* (accessed March 24, 2013).

Amigoni, Francesco, Nicola Basilico, Nicola Gatti, Alessandro Saporiti, and Stefano Troiani. "Moving game theoretical patrolling strategies from theory to practice: An usarsim simulation." In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pp. 426-431. IEEE, 2010.

An, Bo, James Pita, Eric Shieh, Milind Tambe, Chris Kiekintveld, and Janusz Marecki. "Guards and Protect: Next Generation Applications of Security Games." *ACM SIGecom Exchanges* 10, no. 1 (2011): 31-34.

Basilico, Nicola, Nicola Gatti, and Francesco Amigoni. "Developing a deterministic patrolling strategy for security agents." In *Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology-Volume 02*, pp. 565-572. IEEE Computer Society, 2009.

Basilico, Nicola, Nicola Gatti, Sofia Ceppi, and Francesco Amigoni. "Extending Algorithms for Mobile Robot Patrolling in the Presence of Adversaries to More Realistic Settings." *Proceeding WI-IAT '09 Proceedings of the 2009 IEEE/WIC/ACM International Joint Conference on Web Intelligence and Intelligent Agent Technology* 02 (2009): 557-564.

Bonanno, Giacomo. *Non-Cooperative Game Theory*. No. 86. 2008.

"Border Patrol National Strategy 2012-2016."(2012). http://www.krgv.com/files/2012-2016_BP_Strategy.pdf (accessed March 22, 2013).

Camerer, Colin. *Behavioral Game Theory: Experiments in Strategic Interaction*. New York, N.Y.: Russell Sage Foundation ;, 2003.

"CBP Border Security Spotlight." CBP.gov - home page. http://www.cbp.gov/xp/cgov/border_security/ (accessed July 5, 2013).

Chawathe, Sudarshan S. "Organizing hot-spot police patrol routes." In *Intelligence and Security Informatics, 2007 IEEE*, pp. 79-86. IEEE, 2007.

Chevaleyre, Yann, Francois Sempe, and Geber Ramalho. "A theoretical analysis of multi-agent patrolling strategies." In *Proceedings of the Third International Joint Conference on Autonomous Agents and Multiagent Systems-Volume 3*, pp. 1524-1525. IEEE Computer Society, 2004.

Cornelius, Wayne. "Evaluating Enhanced US Border Enforcement." *Migration Information Source: Fresh Thought, Authoritative, Global Reach* (2004). Accessed March 22, 2013).

Davis, Morton D. *Game theory: a Nontechnical Introduction*. Courier Dover Publications, 1983.

Garnaev, A., G. Garnaeva, and P. Goutal. "On the infiltration game."*International Journal of Game Theory* 26, no. 2 (1997): 215-221.

Gatti, Nicola. "Game Theoretical Insights in Strategic Patrolling: Model and Algorithm in Normal-Form." *ECAI-08* (2008): 403-407.

Geyer, Charles J. "Practical Markov Chain Monte Carlo." *Statistical Science* 7, no. 4 (1992): 473-483.

Gutierrez, Eric, Jonathan Juett, and Christopher Kiekintveld. "Generating Effective Patrol Strategies to Enhance U.S. Border Security." *Journal of Strategic Security* 6, no. 3 Suppl. (2013): 152-159.

Jain, Manish, Jason Tsai, James Pita, Christopher Kiekintveld, Shyamsunder Rathi, Milind Tambe, and Fernando Ordóñez. "Software Assistants for Randomized Patrol Planning for the LAX Airport Police and the Federal Air Marshal Service." *Interfaces* 40, no. 4 (2010): 267-290.

Jakob, Michal, Ondrej Vanek, and Michal Pechoucek. "Using agents to improve international maritime transport security." *Intelligent Systems, IEEE* 26, no. 1 (2011): 90-96.

Liu, Wei, and Sanjay Chawla. "A Game Theoretical Model for Adversarial Learning." In *Data Mining Workshops, 2009. ICDMW'09. IEEE International Conference on*, pp. 25-30. IEEE, 2009.

Marino, Alessandro, Lynne Parker, Gianluca Antonelli, and Fabrizio Caccavale. "Behavioral Control for Multi-Robot Perimeter Patrol: A Finite State Automata Approach." (2009): ieeexplore.ieee.org (accessed March 18, 2013).

Melo, Adriano, Mairon Belchior, and Vasco Furtado. "Analyzing Police Patrol Routes with the Simulation of the Physical Reorganization of Agents." (2005).

North American Publishing Co. "US Border Patrol 50th Anniversary, 1924-1974." *State Police Officers Journal* 16, no. 37 (1974): 50-56.

Nunez-Neto, Blas. "Border Security: The Role of the U.S. Border Patrol." *Cornell University ILR School* (2008).

Pita, James, Milind Tambe, Chris Kiekintveld, Shane Cullen, and Erin Steigerwald. "Guards - Game Theoretic Security Allocation on a National Scale." In *Proc. of The 10th International Conference on Autonomous Agents and Multiagent Systems (AAMAS)*, vol. 3. 2011.

Predd, Joel, Henry Willis, Claude Setodji, and Chuck Stelzner. "Using Pattern Analysis and Systematic Randomness to Allocate U.S. Border Security Resources."*RAND: Homeland Security and Defense Center* (2012).

Tormos, P., A. Lova, Federico Barber, L. Ingolotti, Montserrat Abril, and Miguel A. Salido. "A genetic algorithm for railway scheduling problems." In*Metaheuristics for Scheduling in Industrial and Manufacturing Applications*, pp. 255-276. Springer Berlin Heidelberg, 2008.

Tsai, Jason, Christopher Kiekintveld, Fernando Ordonez, Milind Tambe, and Shyamsunder Rathi. "IRIS-a tool for strategic security allocation in transportation networks." (2009).

Curriculum Vita

Eric Gutierrez graduated from Norman High School in Norman, Oklahoma. He received a B.S. in Computer Science at The University of Texas at El Paso under the College of Engineering in 2011. In 2014, he received an M.S. in Intelligence and National Security Studies with a Graduate Certificate in Homeland Security, receiving a 4.0/4.0 GPA. Eric Gutierrez received an Air Force ROTC Scholarship from 2006-2008 and received the INSS Graduate Boeing Scholarship in 2013-2014. In ROTC, he received numerous awards, including Distinctive GMC of the Semester in 2005 and GMC of the Semester in 2006. He was a member of the Arnold Air Society and a member of The Blue Knights Honor Guard from 2005-2008 at New Mexico State University. He also joined the Army ROTC Pershing Rifles in 2007 at UTEP. He has conducted research at UTEP in the Computer Science Department focusing on Fuzzy Methodology and Game Theory. He has participated with the Intelligent Agents and Strategic Reasoning Laboratory from 2011-2014, and has developed GAMMASys while working with this research group. He has published *Fundamental Physical Equations Can Be Derived By Applying Fuzzy Methodology to Informal Physical Ideas,* which was presented at the *North American Fuzzy Information Processing Society Conference* in 2011. He has also published *Generating Effective Patrol Strategies to Enhance U.S. Border Security* in the *Journal of Strategic Security* in 2013. This publication was also presented at the *International Association For Intelligence Education Conference* in 2013.

Contact: ejgutierrez@miners.utep.edu