PROTECT: A Deployed Game Theoretic System to Protect the Ports of the United States

Eric Shieh[+], Bo An[+], Rong Yang[+], Milind Tambe[+], Craig Baldwin*, Joseph DiRenzo*, Ben Maule*, Garrett Meyer*
[+]University of Southern California
[+]{eshieh, boa, yangrong, tambe}@usc.edu
*United States Coast Guard
*{Craig.W.Baldwin, Joseph.DiRenzo, Ben.J.Maule, Garrett.R.Meyer}@uscg.mil

ABSTRACT

While three deployed applications of game theory for security have recently been reported at AAMAS [12], we as a community remain in the early stages of these deployments; there is a continuing need to understand the core principles for innovative security applications of game theory. Towards that end, this paper presents PROTECT, a game-theoretic system deployed by the United States Coast Guard (USCG) in the port of Boston for scheduling their patrols. USCG has termed the deployment of PROTECT in Boston a success, and efforts are underway to test it in the port of New York, with the potential for nationwide deployment.

PROTECT is premised on an attacker-defender Stackelberg game model and offers five key innovations. First, this system is a departure from the assumption of perfect adversary rationality noted in previous work, relying instead on a quantal response (QR) model of the adversary's behavior — to the best of our knowledge, this is the first real-world deployment of the QR model. Second, to improve PROTECT's efficiency, we generate a compact representation of the defender's strategy space, exploiting equivalence and dominance. Third, we show how to practically model a real maritime patrolling problem as a Stackelberg game. Fourth, our experimental results illustrate that PROTECT's QR model more robustly handles real-world uncertainties than a perfect rationality model. Finally, in evaluating PROTECT, this paper for the first time provides real-world data: (i) comparison of human-generated vs PROTECT security schedules, and (ii) results from an Adversarial Perspective Team's (human mock attackers) analysis.

Categories and Subject Descriptors

J.m [**Computer Applications**]: MISCELLANEOUS

General Terms

Security, Design

Keywords

Game Theory, Security, Applications, Stackelberg Games

1. INTRODUCTION

Appears in: *Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems – Innovative Applications Track (AAMAS 2012)*, Conitzer, Winikoff, Padgham, and van der Hoek (eds.), 4-8 June 2012, Valencia, Spain.

The global need for security of key infrastructure with limited resources has led to significant interest in research conducted in multiagent systems towards game-theory for real-world security. As reported previously at AAMAS, three applications based on Stackelberg games have been transitioned to real-world deployment. This includes ARMOR, used by the Los Angeles International Airport [12] to randomize checkpoints of roadways and canine patrols; IRIS which helps the US Federal Air Marshal Service [12] in scheduling air marshals on international flights; and GUARDS [12] which is under evaluation by the US Transportation Security Administration to allocate resources for airport protection. We as a community remain in the early stages of these deployments, and must continue to develop our understanding of core principles of innovative applications of game theory for security.

To this end, this paper presents a new game-theoretic security application to aid the United States Coast Guard (USCG), called *Port Resilience Operational/Tactical Enforcement to Combat Terrorism* (PROTECT). The USCG's mission includes maritime security of the US coasts, ports, and inland waterways; a security domain that faces increased risks in the context of threats such as terrorism and drug trafficking. Given a particular port and the variety of critical infrastructure that an adversary may attack within the port, USCG conducts patrols to protect this infrastructure; however, while the adversary has the opportunity to observe patrol patterns, limited security resources imply that USCG patrols cannot be at every location 24/7. To assist the USCG in allocating its patrolling resources, similar to previous applications [12], PROTECT uses an attacker-defender Stackelberg game framework, with USCG as the defender against terrorist adversaries that conduct surveillance before potentially launching an attack. PROTECT's solution is to typically provide a mixed strategy, i.e. randomized patrol patterns taking into account the importance of different targets, and the adversary's surveillance and anticipated reaction to USCG patrols.

While PROTECT builds on previous work, this paper highlights five key innovations. The first and most important is PROTECT's departure from the assumption of perfect rationality on the part of the human adversaries. While appropriate in the initial applications as a first step — ARMOR, IRIS, GUARDS — this assumption of perfect rationality is well-recognized as a limitation of classical game theory, and bounded rationality has received significant attention in behavioral game-theoretic approaches [4]. Within this behavioral framework, quantal response equilibrium has emerged as a promising approach to model human bounded rationality [4, 10, 14] including recent results illustrating the benefits of the quantal response (QR) model in security games contexts [15]. Therefore, PROTECT uses a novel algorithm called PASAQ [16] based on the QR model of a human adversary. To the best of our knowledge, this

is the *first time that the QR model has been used in a real-world security application*.

Second, PROTECT improves PASAQ's efficiency via a compact representation of defender strategies exploiting dominance and equivalence analysis. Experimental results show the significant benefits of this compact representation. Third, PROTECT addresses practical concerns of modeling real-world maritime patrolling application in a Stackelberg framework. Fourth, this paper presents a detailed simulation analysis of PROTECT's robustness to uncertainty that may arise in the real-world. For various cases of added uncertainty, the paper shows that PROTECT's quantal-response-based approach leads to significantly improved robustness when compared to an approach that assumes full attacker rationality.

PROTECT has been in use at the port of Boston since April 2011 and been evaluated by the USCG. This evaluation brings forth our final key contribution: for the first time, this paper provides real-world data comparing human-generated and game-theoretic schedules. We also provide results from an Adversarial Perspective Team's (APT) analysis and comparison of patrols before and after the use of the PROTECT system from a viewpoint of an attacker. Given the success of PROTECT in Boston, we are now extending it to the port of New York, and based on the outcome there, it may potentially be extended to other ports in the US.

(a) PROTECT is being used in Boston

(b) Extending PROTECT to NY

Figure 1: USCG boats patrolling the ports of Boston and NY

2. USCG AND PROTECT'S GOALS

The USCG continues to face challenges with evolving asymmetric threats within the maritime environment not only within the Maritime Global Commons, but also within the ports and waterways that make up the United States Maritime Transportation System. The former Director of National Intelligence, Dennis Blair noted in 2010 a persistent threat "from al-Qa'ida and potentially others who share its anti-Western ideology. A major terrorist attack may emanate from either outside or inside the United States" [3]. This threat was reinforced in May of 2011 following the raid on Osama Bin Laden's home, where a large trove of material was uncovered, including plans to attack an oil tanker. "There is an indication of intent, with operatives seeking the size and construction of tankers, and concluding it's best to blow them up from the inside because of the strength of their hulls" [6]. These oil tankers transit the U.S. Maritime Transportation System. The USCG plays a key role in the security of this system and the protection of seaports to support the economy, environment, and way of life in the US.

Coupled with challenging economic times, USCG must operate as effectively as possible, achieving maximum benefit from every hour spent on patrol. As a result, USCG is compelled to re-examine the role that optimization of security resource usage plays in its mission planning — and how innovation provided by game theory can be effectively employed.

The goal of PROTECT is to use game theory to assist the USCG in maximizing its effectiveness in the Ports, Waterways, and Coastal Security (PWCS) Mission. PWCS patrols are focused on protecting critical infrastructure; without the resources to provide one hundred percent on scene presence at any, let alone all of the critical infrastructure, optimization of security resource is critical. Towards that end, unpredictability creates situations of uncertainty for an enemy and can be enough to deem a target less appealing.

The PROTECT system, focused on the PWCS patrols, addresses how the USCG should optimally patrol critical infrastructure in a port to maximize protection, knowing that the adversary may conduct surveillance and then launch an attack. While randomizing patrol patterns is key, PROTECT also addresses the fact that the targets are of unequal value, understanding that the adversary will adapt to whatever patrol patterns USCG conducts. The output of PROTECT is a schedule of patrols which includes when the patrols are to begin, what critical infrastructure to visit for each patrol, and what activities to perform at each critical infrastructure. While initially pilot tested in the port of Boston, the solution technique was intended to be generalizable and applicable to other ports.

3. KEY INNOVATIONS IN PROTECT

The PWCS patrol problem was modeled as a leader-follower (or attacker-defender) Stackelberg game [7] with USCG as the leader (defender) and the terrorist adversaries in the role of the follower. The choice of this framework was supported by prior successful applications of Stackelberg games [12]. In this Stackelberg game framework, the defender commits to a mixed (randomized) strategy of patrols, whereas the attacker conducts surveillance of these mixed strategies and responds with a pure strategy of an attack on a target. The objective of this framework is to find the optimal mixed strategy for the defender.

Stackelberg games have been well established in the multi-agent systems literature [5, 8, 9, 12]. Therefore, rather than providing further background in these games, this section immediately transitions to three of PROTECT's key innovations. We begin by discussing how to practically cast this real-world maritime patrolling problem of PWCS patrols as a Stackelberg game (Section 3.1). We also show how to reduce the number of defender strategies (Section 3.2) before addressing the most important of the innovations in PROTECT: its use of the quantal response model (Section 3.3).

3.1 Game Modeling

To model the USCG patrolling domain as a Stackelberg game, we need to define (i) the set of attacker strategies, (ii) the set of defender strategies, and (iii) the payoff function. These strategies and payoffs center on the targets in a port — ports, such as the port of Boston, have a significant number of potential targets (critical infrastructure). In our Stackelberg game formulation, the attacker conducts surveillance on the mixed strategies that the defender has committed to, and can then launch an attack. Thus, the attacks an attacker can launch on different possible targets are considered as his/her pure strategies.

However, the definition of defender strategies is not as straightforward. Patrols last for some fixed duration during the day as specified by USCG, e.g. 4 hours. Our first attempt was to model each target as a node in a graph and allow patrol paths to go from each individual target to (almost all) other targets in the port, generating an almost complete graph on the targets. This method yields the most flexible set of patrol routes that would fit within the maximum duration, covering any permutation of targets within a single patrol.

This method unfortunately faced significant challenges: (i) it required determining the travel time for a patrol boat for each pair of targets, a daunting knowledge acquisition task given the hundreds of pairs of targets; (ii) it did not maximize the use of port geography whereby boat crews could observe multiple targets at once and; (iii) it was perceived as micromanaging the activities of the USCG boat crews, which was undesirable.

Our improved approach to generating defender strategies therefore grouped nearby targets into patrol areas. The presence of patrol areas led the USCG to redefine the set of defensive activities to be performed on patrol areas to provide a more accurate and expressive model of the patrols. Activities that take a longer time provide the defender a higher payoff compared to activities that take a shorter time to complete. This impacts the final patrol schedule as one patrol may visit fewer areas but conduct longer duration defensive activities at the areas, while another patrol may have more areas with shorter duration activities.

To generate all the permutations of patrol schedules, a graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ is created with the patrol areas as vertices $\mathcal{V}$ and adjacent patrol areas as edges $\mathcal{E}$. Using the graph of patrol areas, PROTECT generates all possible patrol schedules, each of which is a closed walk of $\mathcal{G}$ that starts and ends at the patrol area $b \in \mathcal{V}$, the base patrol area for the USCG. The patrol schedules are a sequence of patrol areas and associated defensive activities, and are constrained by a maximum patrol time τ.

The graph $\mathcal{G}$ along with the constraints b and τ are used to generate the defender strategies (patrol schedules). Given each patrol schedule, the total patrol schedule time is calculated (this also includes traversal time between areas, but we ignore it in the following for expository purposes); we then verify that the total time is less than or equal to the maximum patrol time τ. After generating all possible patrol schedules, a game is formed where the set of defender strategies is composed of patrol schedules and the set of attacker strategies is the set of targets. The attacker's strategy was based on targets instead of patrol areas because an attacker will choose to attack a single target.

Table 1 gives an example, where the rows correspond to the defender's strategies and the columns correspond to the attacker's strategies. In this example, there are two possible defensive activities, activity k_1 and k_2, where k_2 provides a higher payoff for the defender than k_1. Suppose that the time bound disallows more than two k_2 activities (given the time required for k_2) within a patrol. Patrol area 1 has two targets (target 1 and 2) while patrol areas 2 and 3 each have one target (target 3 and 4 respectively). In the table, a patrol schedule is composed of a sequence of patrol areas and a defensive activity in each area. The patrol schedules are ordered so that the first patrol area in the schedule denotes which patrol area the defender needs to visit first. In this example, patrol area 1 is the base patrol area, and all of the patrol schedules begin and end at patrol area 1. For example, the patrol schedule in row 2 first visits patrol area 1 with activity k_2, then travels to patrol area 2 with activity k_1, and returns back to patrol area 1 with activity k_1. For the payoffs, if a target i is the attacker's choice and is also part of a patrol schedule, then the defender would gain a reward R_i^d while the attacker would receive a penalty P_i^a, else the defender would receive a penalty P_i^d and the attacker would gain a reward R_i^a. Furthermore, let G_{ij}^d be the payoff for the defender if the defender chooses patrol j and the attacker chooses to attack target i. G_{ij}^d can be represented as a linear combination of the defender reward/penalty on target i and A_{ij}, the effectiveness probability of the defensive activity performed on target i for patrol j, as described by Equation 1. The value of A_{ij} is 0 if target i is not in patrol j.

$$G_{ij}^d = A_{ij}R_i^d + (1 - A_{ij})P_i^d \quad (1)$$

For instance, suppose target 1 is covered using k_1 in strategy 5, and the value of A_{15} is 0.5. If $R_1^d = 150$ and $P_1^d = -50$, then $G_{15}^d = 0.5(150) + (1 - 0.5)(-50) = 50$. ($G_{ij}^a$ would be computed in a similar fashion.) If a target is visited multiple times with different activities, only the highest quality activity is considered.

In the USCG problem, rewards and penalties are based on an analysis completed by a contracted company of risk analysts that looked at the targets in the port of Boston and assigned corresponding values for each one. The types of factors taken into consideration for generating these values include economic damage and injury/loss of life. Meanwhile, the effectiveness probability, A_{ij}, for different defensive activities are decided based on the duration of the activities. Longer activities lead to a higher possibility of capturing the attackers. While Table 1 shows a zero-sum game, the algorithm used by PROTECT is *not limited to a zero-sum game*; the actual payoff values are determined by the USCG.

Patrol Schedule	Target 1	Target 2	Target 3	Target 4
(1:k_1), (2:k_1), (1:k_1)	50,-50	30,-30	15,-15	-20,20
(1:k_2), (2:k_1), (1:k_1)	100,-100	60,-60	15,-15	-20,20
(1:k_1), (2:k_1), (1:k_2)	100,-100	60,-60	15,-15	-20,20
(1:k_2), (2:k_1), (1:k_2)	100,-100	60,-60	15,-15	-20,20
(1:k_1), (3:k_1), (2:k_1), (1:k_1)	50,-50	30,-30	15,-15	10,-10
(1:k_1), (2:k_1), (3:k_1), (1:k_1)	50,-50	30,-30	15,-15	10,-10

Table 1: Portion of a simplified example of a game matrix

3.2 Compact Representation

In our game, the number of defender strategies, i.e. patrol schedules, grows combinatorially, generating a scale-up challenge. To achieve scale-up, PROTECT uses a compact representation of the patrol schedules using two ideas: (i) combining equivalent patrol schedules and; (ii) removal of dominated patrol schedules.

With respect to equivalence, different permutations of patrol schedules provide identical payoff results. Furthermore, if an area is visited multiple times with different activities in a schedule, only the activity that provides the defender the highest payoff requires attention. Therefore, many patrol schedules are equivalent if the set of patrol areas visited and defensive activities in the schedules are the same even if their order differs. Such equivalent patrol schedules are combined into a single compact defender strategy, represented as a set of patrol areas and defensive activities (and minus any ordering information). Table 2 presents a compact version of Table 1, which shows how the game matrix is simplified by using equivalence to form compact defender strategies, e.g. the patrol schedules in the rows 2-4 from Table 1 are represented as a compact strategy Γ_2 = {(1,k_2), (2,k_1)} in Table 2.

Compact Strategy	Target 1	Target 2	Target 3	Target 4
Γ_1 = {(1:k_1), (2:k_1)}	50,-50	30,-30	15,-15	-20,20
Γ_2 = {(1:k_2), (2:k_1)}	100,-100	60,-60	15,-15	-20,20
Γ_3 = {(1:k_1), (2:k_1), (3:k_1)}	50,-50	30,-30	15,-15	10,-10

Table 2: Example compact strategies and game matrix

Next, the idea of dominance is illustrated using Table 2 and noting the difference between Γ_1 and Γ_2 is the defensive activity on patrol area 1. Since activity k_2 gives the defender a higher payoff than k_1, Γ_1 can be removed from the set of defender strategies because Γ_2 covers the same patrol areas while giving a higher

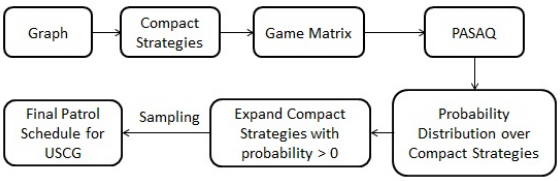


Figure 2: Flow chart of the PROTECT system

payoff for patrol area 1. To generate the set of compact defender strategies, a naive approach would be to first generate the full set of patrol schedules and then prune the dominated and equivalent schedules. Instead, PROTECT uses three ideas to quickly compute the compact strategies: (i) computation of a starting point for compact strategy generation; (ii) computation of a stopping point and; (iii) verification of feasibility in compact strategies.

While generating compact strategies, we first generate compact strategies containing $\hat{n}$ patrol areas, then $\hat{n}-1$ patrol areas and so on until $\check{n}$ patrol areas. $\hat{n}$ is called the starting point and is defined as τ/ρ where τ is the maximum patrol time and ρ shortest duration of a defensive activity. The maximum number of areas in any compact strategy must be less than or equal to $\hat{n}$. For example, if there are 20 patrol areas, τ =100 minutes and $\rho = 10$ minutes, then the algorithm will start by generating compact strategies with 10 patrol areas. It must be verified that a feasible patrol schedule can be formed from each compact strategy. This is achieved by constructing the shortest patrol schedule that is equivalent to the compact strategy, and comparing the patrol travel time against τ.

Let $S(n)$ represent all the compact strategies that contain n patrol areas. If $S(\check{n})$ contains all the compact strategies that are covered with the highest quality defensive activity at each patrol area, the process of generating compact strategies will terminate and $\check{n}$ is called the stopping point of enumeration. Any compact strategy that contains fewer than $\check{n}$ patrol areas will be dominated by a compact strategy in $S(\check{n})$.

Figure 2 shows a high level view of the steps of the algorithm using the compact representation. The compact strategies are used instead of full patrol schedules to generate the game matrix. Once the optimal probability distribution is calculated (as explained in Section 3.3) for the compact strategies, the strategies with a probability greater than 0 are expanded to a complete set of patrol schedules.

In this expansion from a compact strategy to a full set of patrol schedules, we need to determine the probability of choosing each patrol schedule, since a compact strategy may correspond to multiple patrol schedules. The focus here is to increase the difficulty for the attacker to conduct surveillance by increasing unpredictability[1], which we achieve by randomizing uniformly over all expansions of the compact defender strategies. The uniform distribution provides the maximum entropy (greatest unpredictability). Thus, all the patrol schedules generated from a single compact strategy are assigned a probability of v_i/w_i where v_i is the probability of choosing a compact strategy Γ_i and w_i is the total number of expanded patrol schedules for Γ_i. The complete set of patrol schedules and the associated probabilities are then sampled and provided to the USCG, along with the start time of the patrol generated via uniform random sampling.

3.3 Human Adversary Modeling

[1]Creating optimal Stackelberg defender strategies that increase the attacker's difficulty of surveillance is an open research issue in the literature; here we choose to maximize unpredictability as the first step.

t_i	Target i
R_i^d	Defender reward on covering t_i if it's attacked
P_i^d	Defender penalty on not covering t_i if it's attack
R_i^a	Attacker reward on attacking t_i if it's not covered
P_i^a	Attacker penalty on attacking t_i if it's covered
A_{ij}	Effectiveness probability of compact strategy Γ_j on t_i
a_j	Probability of choosing compact strategy Γ_j
J	Total number of compact strategies
x_i	Marginal coverage on t_i

Table 3: PASAQ notation as applied to PROTECT

While previous game-theoretic security applications have assumed a perfectly rational attacker, PROTECT takes a step forward by addressing this limitation of classical game theory. Instead, PROTECT uses a model of a boundedly rational adversary by using a quantal response (QR) model of an adversary, which has shown to be a promising model of human decision making [10, 11, 15]. A recent study demonstrated the use of QR as an effective prediction model of humans [14]. An even more relevant study of the QR model was conducted by Yang et al. [15] in the context of security games where this model was shown to outperform competitors in modeling human subjects. Based on this evidence, PROTECT uses a QR model of a human adversary. (Aided by a software assistant, the defender still computes the optimal mixed strategy.)

The QR model adapts ideas from the literature which presumes that humans will choose better actions at a higher frequency, but with noise added to the decision making process following a logit distribution as defined below

$$q_i = \frac{e^{\lambda G_i^a(x_i)}}{\sum_{j=1}^{T} e^{\lambda G_j^a(x_i)}} \tag{2}$$

The parameter λ represents the amount of noise in the attacker's strategy. λ can range from 0 to ∞ with a value of 0 representing a uniform random probability over attacker strategies while a value of ∞ representing a perfectly rational attacker. q_i corresponds to the probability that the attacker chooses a target i; $G_i^a(x_i)$ corresponds to the attacker's expected utility of attacking target i given x_i, the probability that the defender covers target i; and T is the total number of targets.

To apply the QR model in a Stackelberg framework, PROTECT employs an algorithm known as PASAQ [16]. PASAQ computes the optimal defender strategy (within a guaranteed error bound) given a QR model of the adversary by solving the following non-linear and non-convex optimization problem P, with Table 3 listing the notation:

$$\mathtt{P}: \begin{cases} \max\limits_{x,a} \dfrac{\sum_{i=1}^{T} e^{\lambda R_i^a} e^{-\lambda(R_i^a - P_i^a)x_i}((R_i^d - P_i^d)x_i + P_i^d)}{\sum_{i=1}^{T} e^{\lambda R_i^a} e^{-\lambda(R_i^a - P_i^a)x_i}} \\ x_i = \sum\limits_{j=1}^{J} a_j A_{ij}, \quad \forall i \\ \sum\limits_{j=1}^{J} a_j = 1 \\ 0 \le a_j \le 1, \quad \forall j \end{cases}$$

The first line of the problem corresponds to the computation of the defender's expected utility resulting from a combination of Equations 1 and 2. Unlike previous applications [8, 12], x_i in this case not just summarizes presence or absence on a target, but also the effectiveness probability A_{ij} on the target as well.

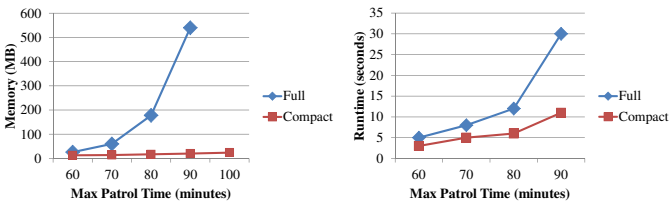
As with all QR models, a value for λ is needed to represent the noise in the attacker's strategy. Based on discussions with USCG experts about the attacker's behavior, a λ value of 0 (uniform random) and ∞ (fully rational) were ruled out. Given the payoff data for Boston, an attacker's strategy with $\lambda = 4$ starts approaching a fully rational attacker — the probability of attack focuses on a single target. It was determined from the knowledge gathered from USCG that the attacker's strategy is best modeled with a λ value that is in the range [0.5, 4]. A discrete sampling approach was used to determine a λ value that gives the highest average expected utility across attacker strategies within this range to get $\lambda = 1.5$. Selecting an appropriate value for λ remains a complex issue however, and it is a key agenda item for future work.

4. EVALUATION

This section presents evaluations based on (i) experiments completed via simulations and (ii) real-world patrol data along with USCG analysis. All scenarios and experiments, including the payoff values and graph (composed of 9 patrol areas), were based off the port of Boston. The defender's payoff values have a range of [-10,5] while the attacker's payoff values have a range of [-5,10]. The game was modeled as a zero-sum game[2] in which the attacker's loss or gain is balanced precisely by the defender's gain or loss. For PASAQ, the defender's strategy uses $\lambda = 1.5$ as mentioned in Section 3.3. All experiments are run on a machine with an Intel Dual Core 1.4 GHz processor and 2 GB of RAM.

4.1 Memory and Run-time Analysis

This section presents the results based on simulation to show the efficiency in memory and run-time of the compact representation versus the full representation (Section 3.2). In Figure 3(a), the x-axis is the maximum patrol time allowed and the y-axis is the memory needed to run PROTECT. In Figure 3(b), the x-axis is the maximum patrol time allowed and the y-axis is the run-time of PROTECT. The maximum patrol time allowed determines the number of combinations of patrol areas that can be visited — so the x-axis indicates a scale-up in the number of defender strategies. When the maximum patrol time is set to 90 minutes, the full representation takes 30 seconds and uses 540 MB of memory while the compact representation takes 11 seconds to run and requires 20 MB of memory. Due to the exponential increase in the memory and run-time that is needed for the full representation, it cannot be scaled up beyond 90 minutes.



(a) Memory comparison (b) Runtime comparison

Figure 3: Comparison of full vs. compact representation

4.2 Utility Analysis

Given that we are working with real data, it is useful to understand whether PROTECT using PASAQ with $\lambda = 1.5$ provides an advantage when compared to: (i) a uniform random defender's strategy; (ii) a mixed strategy with the assumption of the attacker attacking any target uniformly at random ($\lambda = 0$) or; (iii) a mixed strategy assuming a fully rational attacker ($\lambda = \infty$). The previously existing DOBSS algorithm was used for $\lambda = \infty$ [12]. Additionally, comparison with the $\lambda = \infty$ approach is important because of the extensive use of this assumption in previous applications (for our zero-sum case, DOBSS is equivalent to minimax but the utility does not change). Typically, we may not have an estimate of the exact value of the attacker's λ value, only a possible range. Therefore, ideally we would wish to show that PROTECT (with $\lambda = 1.5$) provides an advantage over a range of λ values assumed for the attacker (not just over a point estimate), justifying our use of the PASAQ algorithm.

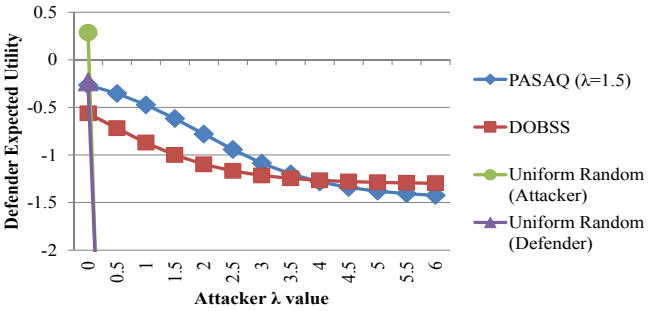


Figure 4: Defender's Expected Utility when varying λ for attacker's strategy(color)

To achieve this, we compute the average defender utility of the four approaches above as the λ value of the attacker's strategy changes from [0, 6], which subsumes the range [0.5, 4] of reasonable attacker strategies. In Figure 4, the y-axis represents the defender's expected utility and the x-axis is the λ value that is used for the attacker's strategy. Both uniform random strategies perform well when the attacker's strategy is based on $\lambda = 0$. However, as λ increases, both strategies quickly drop to a very low defender expected utility. In contrast, the PASAQ strategy with $\lambda = 1.5$ provides a higher expected utility than that assuming a fully rational attacker over a range of attacker λ values (and indeed over the range of interest), not just at $\lambda = 1.5$.

4.3 Robustness Analysis

In the real world, observation, execution, and payoffs, are not always perfect due to the following: noise in the attacker's surveillance of the defender's patrols, the many tasks and responsibilities of the USCG where the crew may be pulled off a patrol, and limited knowledge of the attacker's payoff values. Our hypothesis is that PASAQ with $\lambda = 1.5$ is more robust to such noise than a defender strategy which assumes full rationality of the attacker such as DOBSS [12], i.e. PASAQ's expected defender utility will not degrade as much as DOBSS over the range of attacker λ of interest. This is illustrated by comparing both PASAQ and DOBSS against observation, execution, and payoff noise [8, 9, 17]. (A comparison of the uniform random strategies was not included due to its poor performance shown in Figure 4.) All experiments were run generating 200 samples with added noise and averaging over all the samples. For Figures 5, 6, and 7, the y-axis represents the defender's expected utility and the x-axis is the attacker's λ value, with error bars depicting the standard error.

The first experiment considers observational noise, which means that the attacker has noise associated with observing the defender's patrol strategy as shown in Figure 5. In this scenario, if the defender

[2]In general these types of security games are non-zero-sum [12], however for Boston as a first step it was decided to cast the game as zero-sum.

covered a target with probability p, the attacker may perceive the probability to be uniformly distributed in $[p - x, p + x]$ where x is the noise. The low observation error corresponds to $x = 0.1$ while for high error $x = 0.2$. Contrary to expectation, observation error leads to an increase in defender expected utility in PASAQ, but a potential decrease (or no change) in DOBSS — thus PASAQ ends up dominating DOBSS by a larger margin over bigger ranges of λ, further consolidating the reason to use PASAQ rather than a full-rationality model.

An example illustrates PASAQ's unexpected behavior. Suppose the defender's strategy is $\mathbf{c}$ and there are two targets, t_1 and t_2 with defender expected utilities of $U_1^d(\mathbf{c}) = -2$ and $U_2^d(\mathbf{c}) = -1$, with the attacker's expected utility $U^a(\mathbf{c})$ being the opposite because this is a zero-sum game. For an attacker strategy with a higher λ, the adversary will choose to attack t_1 and the defender would get a utility of -2. When observation noise is added, increases in the coverage of t_1 results in decreases in $U_1^a(\mathbf{c}')$ so the attacker might choose to attack t_2 instead, giving the defender a higher utility than when noise is absent. If the coverage of t_1 decreases, $U_1^a(\mathbf{c}')$ will increase and the attacker will still choose to attack t_1, but $U_1^d(\mathbf{c})$ will remain the same as when there was no noise.

The reason there is a different trend for DOBSS is because DOBSS minimizes the maximum attacker's expected utility or, in our situation, also maximizes the minimum defender's expected utility. This results in multiple targets with the same minimum defender's utility; these targets are referred to as an *attack set* [12]. Typically, when the coverage over the attack set varies due to observation error, some of the targets have less and some have more coverage, but the attacker ends up attacking the targets in the attack set regardless, giving the defender almost no change in its expected utility.

For the second experiment, noise is added to the execution phase of the defender as shown in Figure 6. If the defender covered a target with probability p, this probability now changes to be uniformly distributed in $[p - x, p + x]$ where x is the noise. The low execution error corresponds to $x = 0.1$ whereas high error corresponds to $x = 0.2$. The key takeaway here is that execution error leads to PASAQ dominating DOBSS over all tested values of λ, further strengthening the reason to use PASAQ rather than a full-rationality model. When execution error is added, PASAQ dominates DOBSS because the latter seeks to maximize the minimum defender's expected utility so multiple targets will have the same minimum defender utility. For DOBSS, when execution error is added, there is a greater probability that one of these targets will have less coverage, resulting in a lower defender's expected utility. For PASAQ, typically only one target has the minimum defender expected utility. As a result changes in coverage do not impact it as much as DOBSS. Similar to observation error, as execution error increases, the advantage in the defender's expected utility of PASAQ over DOBSS increases even more.

In the third experiment shown in Figure 7, payoff noise is added by aggregating mean-0 Gaussian noise to the attacker's original payoff values (similar to [8]). As more noise is added to the payoffs, both defenders' strategies result in an increase in the defender's expected utility because the game is no longer zero-sum. The low payoff noise corresponds to a standard deviation of 1 while a high payoff noise corresponds to a standard deviation of 1.5. Similar to the previous experiments, when payoff noise is added, DOBSS is dominated by PASAQ, indicating the robustness of PASAQ. As noise is added to the attacker's payoff but not the defender's payoff, the attacker's strategy may no longer result in the lowest possible defender expected utility. For example, with no payoff noise, target t_1 gives the attacker the highest utility and the defender the lowest utility. When noise is added to the attacker's payoffs, t_1 may no longer give the attacker the highest utility; instead, he/she will choose to attack target t_2, and the defender receives a higher utility than t_1. In essence, with a zero-sum game, the defender has planned a conservative strategy, based on maximin, and as such any change in the attacker is to the defender's benefit in this case.

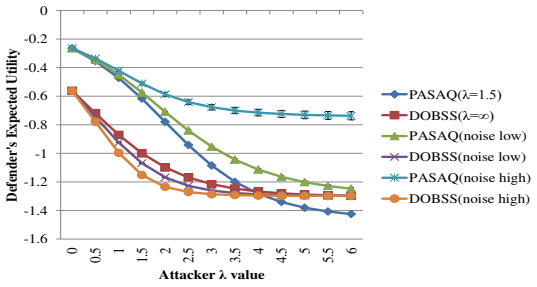


Figure 5: Defender's expected utility: Observation noise(color)

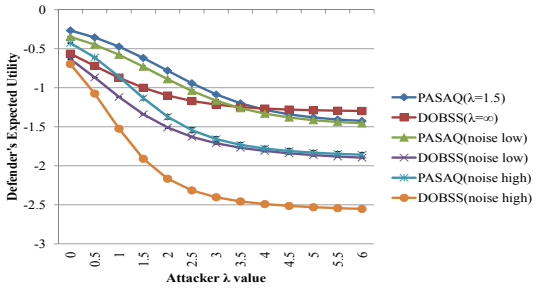


Figure 6: Defender's expected utility: Execution noise(color)

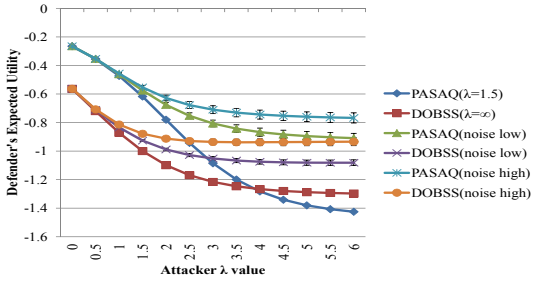


Figure 7: Defender's expected utility: Payoff noise(color)

4.4 USCG Real-World Evaluation

In addition to the data made available from simulations, the USCG conducted its own real-world evaluation of PROTECT. With permission, some aspects of the evaluation are presented in this paper.

Real-world scheduling data: Unlike prior publications at AAMAS of real-world applications of game theory for security, a key novelty of this paper is the inclusion of actual data from USCG patrols before and after the deployment of PROTECT at the port of Boston. Figure 8 and Figure 9 show the frequency of visits by USCG to different patrol areas over a number of weeks. The x-axis is the day of the week, and the y-axis is the number of times a patrol area is visited for a given day of the week. The y-axis is intentionally blurred for security reasons as this is real data from Boston. There are more lines in Figure 8 than in Figure 9 because during the implementation of PROTECT, new patrol areas were formed which contained more targets and thus fewer patrol areas in the post-PROTECT figure. Figure 8 depicts a definite pattern in the

patrols. While there is a spike in patrols executed on Day 5, there is a dearth of patrols on Day 2. Besides this pattern, the lines in Figure 8 intersect, indicating that some days, a higher value target was visited more often while on other days it was visited less often, even though the value of a target does not change day-to-day. This means that there was not a consistently high frequency of coverage of higher value targets before PROTECT.

In Figure 9, we notice that the pattern of low patrols on Day 2 (from Figure 8) disappears. Furthermore, lines do not frequently intersect, i.e. higher valued targets are visited consistently across the week. The top line in Figure 9 is the base patrol area and is visited at a higher rate than all other patrol areas.

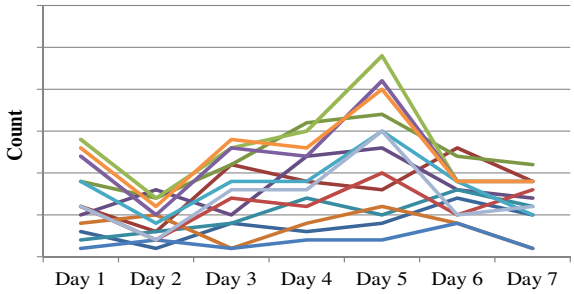


Figure 8: Patrol visits per day by area - pre-PROTECT(Color)

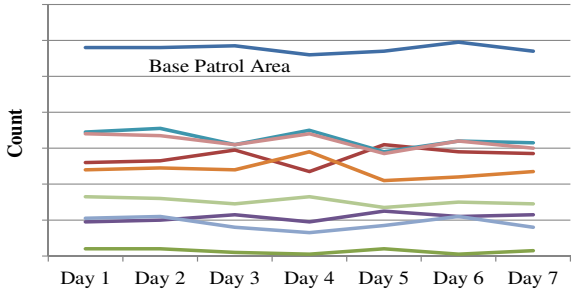


Figure 9: Patrol visits per day by area - post-PROTECT(Color)

Adversary Perspective Teams(APT): To obtain a better understanding of how the adversary views the potential targets in the port, the USCG created the Adversarial Perspective Team (APT), a mock attacker team. The APT provides assessments from the terrorist perspective and as a secondary function, assesses the effectiveness of the patrol activities before and after deployment of PROTECT. In their evaluation, the APT incorporates the adversary's known intent, capabilities, skills, commitment, resources, and cultural influences. In addition, it screens attack possibilities and assists in identifying the level of deterrence projected at and perceived by the adversary. For the purposes of this research, the adversary is defined as an individual(s) with ties to al-Qa'ida or its affiliates.

The APT conducted a pre- and post-PROTECT assessment of the system's impact on an adversary's deterrence at the port of Boston. This analysis uncovered a positive trend where the effectiveness of deterrence increased from the pre- to post- PROTECT observations.

Additional Real-world Indicators: The use of PROTECT and APT's improved guidance given to boat crews on how to conduct the patrol jointly provided a noticeable increase in the quality and effectiveness of the patrols. Prior to implementing PROTECT, there were no documented reports of illicit activity. After implementation, USCG crews, reported more illicit activities within the port and provided a noticeable "on the water" presence with industry port partners commenting, "the Coast Guard seems to be everywhere, all the time." With no actual increase in the number of resources applied, and therefore no increase in capital or operating costs, these outcomes support the practical application of game theory in the maritime security environment.

4.5 Outcomes after Boston Implementation

After evaluating the performance and impact of PROTECT at Boston, the USCG viewed this system as a success. As a result, PROTECT is now getting deployed in the port of New York. We were presented an award for the work on the PROTECT system for the Boston Harbor which reflects USCG's recognition of the impact and value of PROTECT.

5. LESSONS LEARNED: PUTTING THEORY INTO PRACTICE

Developing the PROTECT model was a collaborative effort involving university researchers and USCG personnel representing decision makers, planners and operators. Building on the lessons reported in [12] for working with security organizations, we informed the USCG of (i) the assumptions underlying the game-theoretic approaches, e.g. full adversary rationality, and strengths and limitations of different algorithms — rather than pre-selecting a simple heuristic approach; (ii) the need to define and collect correct inputs for model development and; (iii) a fundamental understanding of how the inputs affect the results. We gained three new insights involving real-world applied research; (i) unforeseen positive benefits because security agencies were compelled to reexamine their assumptions; (ii) requirement to work with multiple teams in a security organization at multiple levels of their hierarchy and; (iii) need to prepare answers to end-user practical questions not always directly related to the "meaty" research problems.

The first insight came about when USCG was compelled to reassess their operational assumptions as a result of working through the research problem. A positive result of this reexamination prompted USCG to develop new PWCS mission tactics, techniques and procedures. Through the iterative development process, USCG reassessed the reasons why boat crews performed certain activities and whether they were sufficient. For example, instead of "covered" vs "not covered" as the only two possibilities at a patrol point, there are now multiple sets of activities at each patrol point.

The second insight is that applied research requires the research team to collaborate with planners and operators on the multiple levels of a security organization to ensure the model accounts for all aspects of a complex real world environment. Initially when we started working on PROTECT, the focus was on patrolling each individual target. This appeared to micromanage the activities of boat crews, and it was through their input that individual targets were grouped into patrol areas associated with a PWCS patrol. On the other hand, input from USCG headquarters and the APT mentioned earlier, led to other changes in PROTECT, e.g. departing from a fully rational model of an adversary to a QR model.

The third insight is the need to develop answers to end-user questions which are not always related to the "meaty" research question but are related to the larger knowledge domain on which the research depends. One example of the need to explain results involved the user citing that one patrol area was being repeated and hence, randomization did not seem to occur. After assessing this concern, we determined that the cause for the repeated visits to a patrol area was its high reward — order of magnitude greater than the rarely visited patrol areas. PROTECT correctly assigned patrol schedules that covered the more "important" patrol areas more frequently. In another example, the user noted that PROTECT did not

assign any patrols to start at 4:00 AM or 4:00 PM over a 60 day test period. They expected patrols would be scheduled to start at any hour of the day, leading them to ask if there was a problem with the program. This required us to develop a layman's briefing on probabilities, randomness, and sampling. With 60 patrol schedules, a few start hours may not be chosen given our uniform random sampling of the start time. These practitioner-based issues demonstrate the need for researchers to not only be conversant in the algorithms and math behind the research, but also be able to explain from a user's perspective how solutions are accurate. An inability to address these issues would result in a lack of real-world user confidence in the model.

6. SUMMARY AND RELATED WORK

This paper reports on PROTECT, a game-theoretic system deployed by the USCG in the port of Boston since April 2011 for scheduling their patrols. USCG has deemed the deployment of PROTECT in Boston a success and efforts are underway to deploy PROTECT in the port of New York, and to other ports in the United States. PROTECT uses an attacker-defender Stackelberg game model, and includes five key innovations.

First, PROTECT moves away from the assumption of perfect adversary rationality seen in previous work, relying instead on a quantal response (QR) model of the adversary's behavior. While the QR model has been extensively studied in the realm of behavioral game theory, to the best of our knowledge, this is its first real-world deployment. Second, to improve PROTECT's efficiency, we generate a novel compact representation of the defender's strategy space, exploiting equivalence and dominance. Third, the paper shows how to practically model a real-world (maritime) patrolling problem as a Stackelberg game. Fourth, we provide experimental results illustrating that PROTECT's QR model of the adversary is better able to handle real-world uncertainties than a perfect rationality model. Finally, for the first time in a security application evaluation, we use real-world data: (i) providing a comparison of human-generated security schedules versus those generated via a game-theoretic algorithm and; (ii) results from an APT's analysis of the impact of the PROTECT system. The paper also outlined the insights from the project which include the ancillary benefits due to a review of assumptions made by security agencies, and the need for knowledge to answer questions not directly related to the research problem.

As a result, PROTECT has advanced the state of the art beyond previous applications of game theory for security. Prior applications mentioned earlier, including ARMOR, IRIS or GUARDS [12], have each provided unique contributions in applying novel game-theoretic algorithms and techniques. Interestingly, these applications have revolved around airport and air-transportation security. PROTECT's novelty is not only its application domain in maritime patrolling, but also in the five key innovations mentioned above, particularly its emphasis on moving away from the assumption of perfect rationality by using the QR model.

In addition to game-theoretic applications, the issue of patrolling has received significant attention in the multi-agent literature. These include patrol work done by robots primarily for perimeter patrols that have been addressed in arbitrary topologies [2], maritime patrols in simulations for deterring pirate attacks [13], and in research looking at the impact of uncertainty in adversarial behavior [1]. PROTECT differs from these approaches in its use of a QR model of a human adversary in a game theoretic setting, and in being a deployed application. Building on this initial success of PROTECT, we hope to deploy it at more and much larger-sized ports. In so doing, in the future, we will consider significantly more complex attacker strategies, including potential real-time surveillance and coordinated attacks.

7. ACKNOWLEDGMENTS

We thank the USCG offices, and particularly sector Boston, for their exceptional collaboration. The views expressed herein are those of the author(s) and are not to be construed as official or reflecting the views of the Commandant or of the U.S. Coast Guard. This research was supported by the United States Department of Homeland Security through the Center for Risk and Economic Analysis of Terrorism Events (CREATE) under award number 2010-ST-061-RE0001.

8. REFERENCES

[1] N. Agmon, S. Kraus, G. A. Kaminka, and V. Sadov. Adversarial uncertainty in multi-robot patrol. In *IJCAI*, 2009.

[2] N. Basilico, N. Gatti, and F. Amigoni. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *AAMAS*, 2009.

[3] D. Blair. Annual threat assessment of the US intelligence community for the senate select committee on intelligence. *http://www.dni.gov/testimonies/20100202_testimony.pdf*, 2010.

[4] C. F. Camerer. *Behavioral Game Theory: Experiments in Strategic Interaction*. Princeton University Press, 2003.

[5] V. Conitzer and T. Sandholm. Computing the optimal strategy to commit to. In *ACM EC*, 2006.

[6] K. Dozier. Bin laden trove of documents sharpen US aim. *http://www.msnbc.msn.com/id/43331634/ns/us_news-security/t/bin-laden-trove-documents-sharpen-us-aim/*, 2011.

[7] D. Fudenberg and J. Tirole. *Game Theory*. MIT Press, 1991.

[8] C. Kiekintveld, J. Marecki, and M. Tambe. Approximation methods for infinite bayesian Stackelberg games: modeling distributional uncertainty. In *AAMAS*, 2011.

[9] D. Korzhyk, V. Conitzer, and R. Parr. Solving Stackelberg games with uncertain observability. In *AAMAS*, 2011.

[10] R. D. McKelvey and T. R. Palfrey. Quantal response equilibria for normal form games. *Games and Economic Behavior*, 10(1):6–38, 1995.

[11] B. W. Rogers, T. R. Palfrey, and C. F. Camerer. Heterogeneous quantal response equilibrium and cognitive hierarchies. *Journal of Economic Theory*, 2009.

[12] M. Tambe. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, 2011.

[13] O. Vanek, M. Jakob, O. Hrstka, and M. Pechoucek. Using multi-agent simulation to improve the security of maritime transit. In *MABS*, 2011.

[14] J. Wright and K. Leyton-Brown. Beyond equilibrium: Predicting human behavior in normal form games. In *AAAI*, 2010.

[15] R. Yang, C. Kiekintveld, F. Ordonez, M. Tambe, and R. John. Improving resource allocation strategy against human adversaries in security games. In *IJCAI*, 2011.

[16] R. Yang, M. Tambe, and F. Ordonez. Computing optimal strategy against quantal response in security games. In *AAMAS*, 2012.

[17] Z. Yin, M. Jain, M. Tambe, and F. Ordóñez. Risk-averse strategies for security games with execution and observational uncertainty. In *AAAI*, 2011.