Addressing Behavioral Uncertainty in Security Games: An Efficient Robust Strategic Solution for Defender Patrols

Thanh H. Nguyen, and Arunesh Sinha, Milind Tambe
University of Southern California, USA, {thanhhng,aruneshs,tambe}@usc.edu

***Abstract*—Stackelberg Security Games (SSG) have been widely applied for solving real-world security problems — with a significant research emphasis on modeling attackers' behaviors to handle their bounded rationality. However, access to real-world data (used for learning an accurate behavioral model) is often limited, leading to uncertainty in attacker's behaviors while modeling. This paper therefore focuses on addressing behavioral uncertainty in SSG with the following main contributions: 1) we present a new uncertainty game model that integrates uncertainty intervals into a behavioral model to capture behavioral uncertainty; and 2) based on this game model, we propose a novel robust algorithm that approximately computes the defender's optimal strategy in the worst-case scenario of uncertainty. We show that our algorithm guarantees an additive bound on its solution quality.**

I. INTRODUCTION

Real-world deployed applications of Stackelberg Security Games [17], [16], [2], [9] have led to significant research emphasis on modeling the attacker's bounded rationality [21], [15], [5]. One key assumption in behavioral modeling is the availability of a significant amount of data to obtain an accurate prediction. However, in real-world security domains such as the wildlife protection, this assumption may be inapplicable due to the limited access to real-world data [8], leading to uncertainty in the attacker's behaviors — a key research challenge of security problems.

Recent research has focused on addressing the behavioral uncertainty in modeling, following two different approaches: 1) one approach assumes a known distribution of multiple attacker types, each follows a certain behavioral model, and attempts to solve the resulting Bayesian games [20]; and 2) another approach considers the existence of multiple attacker types of which behavioral models are perfectly known, but without a known distribution over the types. It then focuses on addressing the worst attacker type for the defender [3]. These two approaches have several limitations. First, both still require a sufficient amount of data to precisely estimate either the distribution over attacker types (the former approach) or the decision making for each individual type (the latter approach). Second, solving the resulting Bayesian games in the former case is computationally expensive. Third, the latter approach tends to be overly conservative as it only focuses on the worst case.

This paper is an extension of the work in [14] which remedies these shortcomings of state-of-the-art approaches when addressing behavioral uncertainty in SSG by providing two key contributions. First, we present a new uncertainty game model in which we consider a single behavioral model to capture decision making of the whole attacker population (instead of multiple behavioral models), while uncertainty intervals are integrated with the chosen model to capture behavioral uncertainty. The idea of uncertainty interval is commonly used in literature [1] and has been shown to effectively represent uncertainty in SSG [6], [13]. Second, based on this game model, we propose a new efficient robust algorithm, CUBIS (**C**ompeting **U**ncertainty in attacker **B**ehaviors using **I**nterval-based maximin **S**olution), that approximately computes the defender's optimal strategy which that robust to the uncertainty; we provide a bound guarantee for CUBIS's solution quality.

Overall, the resulting robust optimization problem for computing the defender's optimal strategy against the worst case of behavioral uncertainty is a non-linear non-convex fractional maximin problem. Our new CUBIS algorithm efficiently solves this problem based on the following key insights: 1) it converts the problem into a single maximization problem via a non-linear conversion for fractional terms and the duality of the inner minimization in maximin; 2) a binary search is then applied to simplify the resulting optimization problem; and 3) CUBIS explores extreme points of the feasible solution region and use the piece-wise linear approximation to convert the problem into a Mixed Integer Linear Program (MILP). CUBIS provides an $O(\epsilon + \frac{1}{K})$-optimal solution where ϵ is the convergence threshold for the binary search and K is the number of segments in the piecewise linear approximation.

II. BACKGROUND AND RELATED WORK

Stackelberg security games (SSG). As defined in [14], SSG refer to a class of defender-attacker games in which the defender attempts to optimally allocate her limited security resources to protect a set of T targets from being attacked by the attacker. The key assumption of SSG is that the defender has to commit to a (*mixed*) strategy first and the attacker can observe that strategy and then attack one of the targets [4], [7], [18]. Suppose that the defender has R resources ($R \ll T$) and $\mathbf{x} = \{x_i\}$ is a defender's mixed strategy where x_i is the defender's coverage probability at target i, the defender's feasible strategy set is defined as follows:

$\mathbf{X} = \{\mathbf{x} : 0 \leq x_i \leq 1, \sum_i x_i = R\}$. Suppose that the attacker attacks target i, he will obtain a reward R_i^a if the defender is not protecting that target, otherwise he will get a penalty P_i^a. Conversely, the defender receives a penalty P_i^d in the former case while she obtains a reward R_i^d in the latter case. The expected utility for the two players at target i can be computed as follows:

$$U_i^d(x_i) = x_i R_i^d + (1 - x_i) P_i^d \quad (1)$$

$$U_i^a(x_i) = x_i P_i^a + (1 - x_i) R_i^a \quad (2)$$

Adversarial behavioral models. Recent research in SSG has focused on modeling the attacker's bounded rationality and computing the defender's optimal strategy, assuming the attacker's response follows the given behavioral model. One leading class of behavioral models is Quantal Response (QR) [10], [11], [12]. The SUQR model builds on QR by integrating a subjective utility (Equation 3) into QR, which was shown to provide a better prediction accuracy than QR [15].

$$\hat{U}_i^a(x_i) = w_1 x_i + w_2 R_i^a + w_3 P_i^a \quad (3)$$

where (w_1, w_2, w_3) are key parameters indicating the importance of corresponding features for the attacker.

In modeling the attacker's decision making, we consider a general discrete choice model of QR to capture behaviors of the attacker [19] in which the probability that the attacker chooses target i, $q_i(\mathbf{x})$, is predicted as follows:

$$q_i(x_i) = \frac{F_i(x_i)}{\sum_j F_j(x_j)} \quad (4)$$

where $F_i(x_i) : [0, 1] \rightarrow \mathbb{R}^*$ is a positive and monotonically decreasing function of x_i at target i, which refers to the attacker's utility at that target. For example, SUQR is a special case of (4) in which $F_i(x_i) = e^{\hat{U}_i^a(x_i)}$.

Behavioral uncertainty. Previous work has proposed different methods to handle the behavioral uncertainty. One method follows a Bayesian-based approach which assumes a known distribution of multiple SUQR-followed attackers [20]. Yet, this method requires a significant amount of data to precisely estimate a Bayesian distribution. Furthermore, there does not exist an efficient algorithm to solve the resulting complicated Bayesian games. The second method, on the other hand, considers a robust-based approach which assumes the existence of multiple SUQR-followed attackers without a known distribution and then attempts to address the worst attacker type [3]. This method has two disadvantages: 1) it requires a precise prediction of behaviors for each individual attacker type; and 2) it is overly conservative as it does not consider the fact that all these attacker types exist simultaneously.

Here, we consider a single behavioral model for the whole attacker population and present a new game model that integrates uncertainty intervals into the chosen model, allowing a flexible game representation that captures the behavioral uncertainty. Based on that, we present a new efficient robust algorithm that computes the defender's optimal strategy.

Table I
A 2-TARGET, 1-RESOURCE GAME.

Targets	Att. reward	Att. penalty
1	[1, 5]	[-7, -3]
2	[5, 9]	[-9, -5]

III. BEHAVIORAL ROBUST PROBLEM

In this work, we consider a general model of QR (Equation 4) to reason about the attacker's decision making. However, due to the behavioral uncertainty, we assume that the value of $F_i(x_i)$ in (4) is not perfectly known given x_i. Instead, $F_i(x_i)$ has known lower and upper bounds,

$$L_i(x_i) \leq F_i(x_i) \leq U_i(x_i)$$

where $L_i(x_i), U_i(x_i) : [0, 1] \rightarrow \mathbb{R}^*$ are positive functions of the defender coverage at target i. Denote by $\mathbf{I}(x_i) = [L_i(x_i), U_i(x_i)]$ the uncertainty interval, the interval size indicates the uncertainty level when modeling, which could be specified based on the available data for learning. We aim at computing the defender's optimal strategy by maximizing her utility under the worst case resulting from the behavioral uncertainty. The corresponding robust optimization problem is represented as follows:

$$\max_{\mathbf{x} \in \mathbf{X}} \min_{F_i(x_i) \in \mathbf{I}(x_i), \forall i} \sum_i q_i(x_i) U_i^d(x_i) \quad (5)$$

For example, in a 2-target game as shown in Table I, each target is associated with uncertainty intervals of the attacker's payoffs. For example, if the attacker successfully attacks target 1, he receives a reward which belongs to $[2, 3]$. Conversely, if he gets caught by the defender at that target, the attacker will get a penalty which lies within $[-2, 0]$. Furthermore, parameters for behavioral models of the attacker, i.e., SUQR, are often difficult to precisely estimate due to lack of data. Then the defender can only determine the lower bounds and upper bounds of these parameters. For example, values of (w_1, w_2, w_3) belong to $[-6.0, -2.0]$, $[0.5, 1.0]$, and $[0.4, 0.9]$ respectively.

As a result of uncertainties, the defender can only predict that the value of $F_1(x_1)$ at target 1 where $x_1 = 0.3$ has lower bound $L_1(x_1) = e^{-6.0 \times 0.3 + 0.5 \times 1 + 0.4 \times (-7)} = e^{-4.1}$ and upper bound $U_1(x_1) = e^{-2.0 \times 0.3 + 1.0 \times 5 + 0.9 \times (-3)} = e^{1.7}$. Similarly, only lower bound $L_2(x_2)$ and upper bound $U_2(x_2)$ for $F_2(x_2)$ at target 2 are known. If the defender simply uses the mid points of the uncertainty intervals to compute the optimal strategy for the defender which is $(0.34, 0.66)$, she will receive a utility of -2.26 in the worse case of uncertainty. On the other hand, if the defender plays the robust optimal strategy based on (5) which is $(0.46, 0.54)$, she obtains a utility of -0.90 in the worst-case scenario, which is significantly higher than the former case.

Now, we will introduce our new algorithm to solve the behavioral robust problem represented in (5). Overall, the problem (5) is a non-convex fractional and 2-layer optimization problem (its objective is non-convex fractional) which is not straightforward to solve. In this paper, we present our novel algorithm, CUBIS, which efficiently solves the maximin problem (5) with a bound guarantee on its approximate solution.

IV. The CUBIS Algorithm

In short, there are three key ideas in CUBIS: 1) convert (5) into a single maximization problem via a non-linear conversion for fractional terms and dualty of the inner minimization in (5); 2) apply a binary search to simplify the resulting optimization problem; and 3) explore extreme points of the feasible solution region and use piece-wise linear approximation to convert the resulting feasibility problem at each binary step into a MILP.

A. Convert to a Single Maximization Problem

As a first step, CUBIS attempts to convert (5) into a single maximization problem. Given a defender strategy $\mathbf{x}$, the objective of (5) remains a non-linear fractional function of $F_i(x_i)$, thus making the inner minimization problem in (5) non-linear and fractional. CUBIS first tries to convert the inner minimization problem of (5) into a linear minimization problem through a non-linear variable conversion. In particular, it introduces the following new variables: $y_i = q_i(x_i) = \frac{F_i(x_i)}{\sum_j F_j(x_j)}$ which is the attacking probability at target i and $z = \frac{1}{\sum_j F_j(x_j)}$ which is the normalization term in the attacking probabilities. By replacing $F_i(x_i)$ with the new variables and denote by $\mathbf{y} = \{y_i\}$, CUBIS can represent the inner minimization of (5) as the following linear minimization problem of the new variables $\mathbf{y}$ and z:

$$\min_{\mathbf{y},z} \sum\nolimits_i y_i U_i^d(x_i) \tag{6}$$

$$\text{s.t. } \sum\nolimits_i y_i = 1 \tag{7}$$

$$L_i(x_i)z \leq y_i \leq U_i(x_i)z, \forall i. \tag{8}$$

where constraint (7) ensures the condition on the attacking probability distribution that $\sum_i q_i(x_i) = 1$ holds. In addition, constraint (8) is equivalent to the condition on the lower and upper bound of $F_i(x_i)$ that $F_i(x_i) \in [L_i(x_i), U_i(x_i)]$.

As (6 – 8) is a linear minimization problem of $\mathbf{y}$ and z, its optimal solution is equivalent to the optimal solution of its duality which is the following linear maximization problem:

$$\max_{\{\alpha_i\},\{\beta_i\},\eta} \eta \tag{9}$$

$$\text{s.t. } U_i^d(x_i) - \alpha_i + \beta_i - \eta = 0 \tag{10}$$

$$\sum\nolimits_i L_i(x_i)\alpha_i - \sum\nolimits_i U_i(x_i)\beta_i = 0 \tag{11}$$

$$\alpha_i, \beta_i \geq 0, \forall i. \tag{12}$$

where $\{\alpha_i\}, \{\beta_i\}, \eta$ are dual variables corresponding to the lower bound constraint on the attacker utility $F_i(x_i) \geq L_i(x_i)$ (LHS inequality of (8)), the upper bound constraint $F_i(x_i) \leq U_i(x_i)$ (RHS inequality of (8)), and attacking probability constraint (7) respectively. CUBIS then takes a further step to simplify the problem (9 – 12) via reducing the number of variables by replacing α_i and η with the following equations:

$$\alpha_i = U_i^d(x_i) + \beta_i - \eta \tag{13}$$

$$\eta = \frac{\sum_i L_i(x_i)U_i^d(x_i) - \sum_i [U_i(x_i) - L_i(x_i)]\beta_i}{\sum_i L_i(x_i)} \tag{14}$$

where (13) is obtained based on the constraint (10) and (14) results from replacing α_i with (13) to the constraint (11). Finally, after replacing α_i and η with (13) and (14), CUBIS merges (9 – 12) with the outer maximization of (5) to induce the final single maximization problem (15 – 17) which consists of only two variables: the defender's strategy $\mathbf{x}$ and the dual variable $\boldsymbol{\beta}$ of the upper bound constraint on the attacker utility, $F_i(x_i) \leq U_i(x_i)$, $\forall i$.

$$\max_{\mathbf{x},\boldsymbol{\beta}} H(\mathbf{x}, \boldsymbol{\beta}) \tag{15}$$

$$\text{s.t. } U_i^d(x_i) + \beta_i - H(\mathbf{x}, \boldsymbol{\beta}) \geq 0, \forall i \tag{16}$$

$$\mathbf{x} \in \mathbf{X}, \beta_i \geq 0, \forall i. \tag{17}$$

where $H(\mathbf{x}, \boldsymbol{\beta})$ is the RHS of Equation (14) which is a non-convex fractional function of $\mathbf{x}$ and $\boldsymbol{\beta} = \{\beta_i\}$. In fact, $H(\mathbf{x}, \boldsymbol{\beta})$ is the defender's utility for playing $\mathbf{x}$ in the worst case of uncertainty. In addition, constraint (16) is equivalent to the constraint $\alpha_i \geq 0$ in (12). Although the maximin problem (5) is now transformed into a maximization problem (15 – 17), this optimization problem remains non-convex. We can use any non-convex solver, e.g., Fmincon of MATLAB to solve (15 – 17) with multiple starting points. However, using such non-convex solver is time-consuming.

To solve (15 – 17) efficiently, CUBIS first tries to simplify (15 – 17) by applying a binary search and then apply piecewise linear approximation to linearize the resulting feasibility problem at each binary search step.

B. Binary Search to Remove Fractional Terms

The idea of applying binary search to remove fractional terms is commonly used in security optimization with QR-based adversary behavioral models [21], [3]. However, in these optimization problems, only either objectives or constraints are fractional. Yet, in our modeling robust problem (15 – 17), both the objective and constraints are fractional (the function $H(\mathbf{x}, \boldsymbol{\beta})$ is fractional). Therefore, it is impossible to remove fractional terms by directly checking *value range feasibility* at each binary search step as done in previous work, i.e., given a value of defender utility c, checking if there exist $(\mathbf{x}, \boldsymbol{\beta})$ such that the utility of the defender $H(\mathbf{x}, \boldsymbol{\beta}) \geq c$ and simultaneously constraints (15 – 16) are satisfied. Here, in each binary search step, given a value c, CUBIS instead considers the following *value point feasibility* problem **(P1)** which can be easily converted into a non-fractional feasibility problem (as explained later). The result of **(P1)** then can be used to solve the *value range feasibility* problem of binary search based on Proposition 1.

Problem 1. **(P1)** *Given a value* c*,* $\exists \mathbf{x} \in \mathbf{X}$ *and* $\boldsymbol{\beta} \geq 0$ *such that* $H(\mathbf{x}, \boldsymbol{\beta}) = c$ *and* $U_i^d(x_i) + \beta_i \geq c, \forall i$*?*

Given this *value point feasibility* problem (**P1**), CUBIS now can determine the *value range feasibility* at each binary search step based on the following proposition:

Proposition 1. *If (**P1**) is infeasible for a given value* c_0*, then (**P1**) is infeasible for all value* $c \geq c_0$*.*

Proof: We denote by $S(c) = \{(\mathbf{x}, \boldsymbol{\beta}) : U_i^d(x_i) + \beta_i \geq c, \forall i\}$. We observe that $\inf_{(\mathbf{x},\boldsymbol{\beta}) \in S(c)} H(\mathbf{x}, \boldsymbol{\beta}) = -\infty$ for any given value of c (by taking β_i to $+\infty$, $\forall i$). Therefore, if the problem (**P1**) is infeasible for c_0, it means that for all values of $(\mathbf{x}, \boldsymbol{\beta}) \in S(c_0)$, the value of $H(\mathbf{x}, \boldsymbol{\beta})$ must be less than c_0. Otherwise, if $H(\mathbf{x}, \boldsymbol{\beta}) > c_0$ for some value of $(\mathbf{x}, \boldsymbol{\beta}) \in S(c_0)$, since $H(\mathbf{x}, \boldsymbol{\beta})$ is continuous over the set $S(c_0)$, the value range of $H(\mathbf{x}, \boldsymbol{\beta})$ will consist of the range $(-\infty, c_0]$. Therefore, there must exist $(\mathbf{x}, \boldsymbol{\beta}) \in S(c_0)$ such that $H(\mathbf{x}, \boldsymbol{\beta}) = c_0$, meaning that (**P1**) is feasible.

Now given that (**P1**) is infeasible for c_0, it means that $H(\mathbf{x}, \boldsymbol{\beta}) < c_0$ for all $(\mathbf{x}, \boldsymbol{\beta}) \in S(c_0)$ as explained before. Let's consider a value $c > c_0$, we have $S(c) \subseteq S(c_0)$. Thus, since $H(\mathbf{x}, \boldsymbol{\beta}) < c_0 < c$ for all $(\mathbf{x}, \boldsymbol{\beta}) \in S(c_0)$, then $H(\mathbf{x}, \boldsymbol{\beta}) < c$ for all $(\mathbf{x}, \boldsymbol{\beta}) \in S(c)$ which means that (**P1**) is also infeasible for c. ■

Based on Proposition 1, CUBIS provides the binary search method that iteratively searches over the defender's utility space to find the optimal solution of (15 – 17). At each binary step, CUBIS attempts to solve the feasibility problem (**P1**) that can be transformed to the following feasibility problem which has only non-fractional terms:

Problem 2. (Reformulation of (**P1**)) *Given a value* c*,* $\exists \mathbf{x} \in \mathbf{X}, \boldsymbol{\beta} \geq 0$ *such that* $G(\mathbf{x}, \boldsymbol{\beta}) = 0$ *and* $U_i^d(x_i) + \beta_i \geq c, \forall i$*?*

Here, the non-convex non-fractional function $G(\mathbf{x}, \boldsymbol{\beta})$ is the enumerator of the fractional function $H(\mathbf{x}, \boldsymbol{\beta}) - c$ which is formulated as the follows:

$$G(\mathbf{x}, \boldsymbol{\beta}) = \sum\nolimits_i L_i(x_i) U_i^d(x_i) - \sum\nolimits_i [U_i(x_i) - L_i(x_i)] \beta_i - c[\sum\nolimits_i L_i(x_i)] \tag{18}$$

In order to solve this feasibility problem (**P1**), CUBIS considers the following optimization problem of which optimal solution can be used to determine the feasibility of (**P1**) according to Proposition 2:

$$\max_{\mathbf{x},\beta} G(\mathbf{x}, \boldsymbol{\beta}) \tag{19}$$
$$\text{s.t. } U_i^d(x_i) + \beta_i \geq c, \forall i \tag{20}$$
$$\mathbf{x} \in \mathbf{X}, \beta_i \geq 0, \forall i. \tag{21}$$

Proposition 2. *Denote by* $(\mathbf{x}^*, \boldsymbol{\beta}^*)$ *the optimal solution of (19 – 21), if* $G(\mathbf{x}^*, \boldsymbol{\beta}^*) < 0$*, then the problem (**P1**) is infeasible. Otherwise, if* $G(\mathbf{x}^*, \boldsymbol{\beta}^*) \geq 0$*, there exists* $\boldsymbol{\beta}$ *such that* $(\mathbf{x}^*, \boldsymbol{\beta})$ *is a feasible solution of (**P1**).*

Proof: Denote by $S(c) = \{(\mathbf{x}, \boldsymbol{\beta}) : U_i^d(x_i) + \beta_i \geq c, \forall i, \mathbf{x} \in \mathbf{X}, \boldsymbol{\beta} \geq 0\}$ the set of feasible solutions for (19 – 21). If the optimal solution $G(\mathbf{x}^*, \boldsymbol{\beta}^*) < 0$, it means that $G(\mathbf{x}, \boldsymbol{\beta}) < 0$ for all $(\mathbf{x}, \boldsymbol{\beta}) \in S(c)$ and thus (**P1**) is infeasible.

Conversely, let's suppose that the optimal objective value $G(\mathbf{x}^*, \boldsymbol{\beta}^*) > 0$. Since $\inf_{(\mathbf{x}^*,\boldsymbol{\beta}) \in S(c)} G(\mathbf{x}^*, \boldsymbol{\beta}) = -\infty$ (when $\boldsymbol{\beta}$ goes to $+\infty$) and $G(\mathbf{x}^*, \boldsymbol{\beta})$ is continuous, the feasible value range of $G(\mathbf{x}^*, \boldsymbol{\beta})$ is $(-\infty, G(\mathbf{x}^*, \boldsymbol{\beta}^*)]$ which contains 0. Hence, there exists $(\mathbf{x}^*, \boldsymbol{\beta}) \in S(c)$ such that $G(\mathbf{x}^*, \boldsymbol{\beta}) = 0$, meaning that (**P1**) is feasible. ■

According to Proposition 2, the feasibility of **P1** can be determined via solving the optimization problem (19 – 21). However, the objective function, $G(\mathbf{x}, \boldsymbol{\beta})$, of (19 – 21) remains non-linear non-convex and thus, (19 – 21) is still difficult to solve. CUBIS then attempts to linearize non-linear non-convex terms of this optimization problem by applying piecewise linear approximation.

C. Piecewise Linear Approximation

In general, in order to apply piecewise linear approximation for a non-linear multivariate function $f(\mathbf{y})$ of the variable vector $\mathbf{y}$, it is essential that $f(\mathbf{y})$ is separable, meaning that it can be represented as a sum of terms which are univariate functions of individual scalar variables, i.e., $f(\mathbf{y}) = \sum_i f_i(y_i)$. In previous work on QR-based security problems, it is straightforward to apply piecewise linear approximation for these problems since all related functions are separable [21], [3]. However, in (19 – 21), the current form of the objective $G(\mathbf{x}, \boldsymbol{\beta})$ has the term $\sum_i [U_i(x_i) - L_i(x_i)] \beta_i$ which is not separable between x_i and β_i. Thus, to overcome this issue, CUBIS attempts to explore the extreme points of the feasible solution space of β_i (Proposition 3) and then convert $G(\mathbf{x}, \boldsymbol{\beta})$ into a separable form through a variable conversion (as explain next).

Proposition 3. *If* $(\mathbf{x}^*, \boldsymbol{\beta}^*)$ *is an optimal solution of (19 – 21), then* $\beta_i^* = \max\{0, c - U_i^d(x_i^*)\}, \forall i$*.*

Proof: According to the constraints (20 – 21), for any $\mathbf{x} \in \mathbf{X}$, $(\mathbf{x}, \boldsymbol{\beta})$ is a feasible solution of the problem (19 – 21) if and only if $\beta_i \geq \max\{0, c - U_i^d(\mathbf{x})\}, \forall i$. On the other hand, as $U_i(x_i) - L_i(x_i) \geq 0, \forall i$, the objective function $G(\mathbf{x}, \boldsymbol{\beta})$ is a monotonically decreasing in β_i for all i. Therefore, $G(\mathbf{x}, \boldsymbol{\beta})$ is maximized when β_i is minimized for all i. Thus, if $(\mathbf{x}^*, \boldsymbol{\beta}^*)$ is an optimal solution, then $\beta_i^* = \max\{0, c - U_i^d(x_i^*)\}, \forall i$. ■

Based on this proposition, CUBIS replaces $\boldsymbol{\beta}$ with $\beta_i = \max\{0, c - U_i^d(x_i)\}, \forall i$. CUBIS then substitutes each non-separable term in $G(\mathbf{x}, \boldsymbol{\beta})$, i.e., $[U_i(x_i) - L_i(x_i)]\beta_i$ by a new variable $v_i = [U_i(x_i) - L_i(x_i)]\beta_i$ of which values are determined by the following mixed integer linear constraints:

$$0 \leq v_i \leq Mq_i \tag{22}$$

$$[c - U_i^d(x_i)][U_i(x_i) - L_i(x_i)] \leq v_i \tag{23}$$

$$v_i \leq [c - U_i^d(x_i)][U_i(x_i) - L_i(x_i)] + M(1 - q_i) \tag{24}$$

where M is a sufficient large positive constant to ensure that the RHS of constraint (22) and constraint (24) are effective only when $q_i = 0$ and $q_i = 1$ respectively. In addition, q_i is an integer variable which indicates whether $v_i = 0$ ($q_i = 0$) or $v_i = [U_i(x_i) - L_i(x_i)][c - U_i^d(x_i)]$ ($q_i = 1$). In other words, these constraints ensure that $\beta_i = \max\{0, c - U_i^d(x_i)\}, \forall i$ as following Proposition 3. In particular, constraints (22–23) guarantee that $v_i \geq \max\{0, [c-U_i^d(x_i)][U_i(x_i)-L_i(x_i)]\}$ which is equivalent to $\beta_i \geq \max\{0, c-U_i^d(x_i)\}$. When $q_i = 1$, constraints (23–24) enforce that $v_i = [c - U_i^d(x_i)][U_i(x_i) - L_i(x_i)]$, indicating that $\beta_i = c - U_i^d(x_i)$. Conversely, when $q_i = 0$, constraint (22) guarantees that $v_i = 0$ (or $\beta_i = 0$).

We denote by $f_i^1(x_i) = L_i(x_i)[U_i^d(x_i) - c]$ and $f_i^2(x_i) = U_i(x_i)[U_i^d(x_i) - c]$ for the sake of simplifying the description of the piecewise linear approximation later. These two functions are non-convex. The objective of (19 – 21), $G(\mathbf{x}, \boldsymbol{\beta})$, now can be formulated as follows which only consists of separable terms w.r.t variables x_i and v_i:

$$G(\mathbf{x}, \boldsymbol{\beta}) = \sum\nolimits_i f_i^1(x_i) - \sum\nolimits_i v_i \tag{25}$$

As a result, the optimization problem (19 – 21) can be reformulated as the following mixed integer non-linear programming where all terms are separable w.r.t $\mathbf{x}$ and $\mathbf{v}$:

$$\max\nolimits_{\mathbf{x},\mathbf{v},q_i} \sum\nolimits_i f_i^1(x_i) - \sum\nolimits_i v_i \tag{26}$$

$$\text{s.t. } 0 \leq v_i \leq Mq_i, \forall i \tag{27}$$

$$f_i^1(x_i) - f_i^2(x_i) \leq v_i, \forall i \tag{28}$$

$$v_i \leq f_i^1(x_i) - f_i^2(x_i) + M(1 - q_i), \forall i \tag{29}$$

$$\mathbf{x} \in \mathbf{X}, q_i \in \{0, 1\}, \forall i. \tag{30}$$

where constraints (27 – 29) are equivalent to constraints (22 – 24) respectively. In (26 – 30), the functions $f_i^1(x_i)$ and $f_i^2(x_i)$ are non-linear in x_i for all target i. Therefore, CUBIS then applies piecewise linear approximations for these two functions. Overall, the feasible region of the defender's coverage x_i for all target i, $[0, 1]$, is divided into K equal segments $\left\{\left[0, \frac{1}{K}\right], \left[\frac{1}{K}, \frac{2}{K}\right], \ldots, \left[\frac{K-1}{K}, 1\right]\right\}$ where K is given. The values of $f_i^1(x_i)$ are then approximated by using the segments connecting pairs of points $\left(\frac{k-1}{K}, f_i^1\left(\frac{k-1}{K}\right)\right)$ and $\left(\frac{k}{K}, f_i^1\left(\frac{k}{K}\right)\right)$ for all $k = 1 \ldots K$. Specifically, $f_i^1(x_i)$ is approximated as the follows:

$$f_i^1(x_i) \approx f_i^1(0) + \sum\nolimits_{k=1}^{K} s_k^1 x_{ik} \tag{31}$$

where s_k^1 are the slopes of the k^{th} segment for $f_i^1(x_i)$ which can be determined based on the two end points of the segment, i.e., $s_k^1 = K \times \left[f_i^1\left(\frac{k}{K}\right) - f_i^1\left(\frac{k-1}{K}\right)\right]$. In addition, $x_{i,k}$ refers to the portion of the defender's coverage, x_i, at target i that belongs to the k^{th} segment. In other words, $x_i = \sum_k x_{i,k}$. Here, $x_{i,k} = \frac{1}{K}$ if $x_i \geq \frac{k}{K}$ or $x_{i,k} = 0$ if $x_i < \frac{k-1}{K}$; otherwise, $x_{i,k} = \frac{k}{K} - x_i$.

Example 1. *Suppose that the number of segments, $K = 5$, and the defender's coverage at target i, $x_i = 0.3$, as $\frac{1}{5} < x_i < \frac{2}{5}$, we obtain $x_{i,1} = \frac{1}{5}$, $x_{i,2} = 0.1$, and $x_{i,3} = x_{i,4} = x_{i,5} = 0$. Then $f_i^1(x_i)$ can be piecewise-linearly approximated as $f_i^1(x_i) \approx f_i^1(0) + 5 \times \left[f_i^1\left(\frac{1}{5}\right) - f_i^1(0)\right] \times \frac{1}{5} + 5 \times \left[f_i^1\left(\frac{2}{K}\right) - f_i^1\left(\frac{1}{K}\right)\right] \times 0.1$.*

Similarly, we also obtain the following piecewise-linearly approximation of $f_i^2(x_i)$:

$$f_i^2(x_i) \approx f_i^2(0) + \sum\nolimits_{k=1}^{K} s_k^2 x_{ik} \tag{32}$$

where s_k^2 are the slopes of the k^{th} segments for $f_i^2(x_i)$. As a result, we obtain the final MILP representation for (19 – 21) as shown in (33 – 40) which can be solved by CPLEX.

$$\max_{\mathbf{x},\mathbf{v},\mathbf{q},\mathbf{h}} \sum_i f_i^1(0) + \sum_{k=1}^{K} s_k^1 x_{i,k} - \sum_i v_i \tag{33}$$

$$\text{s.t. } 0 \leq v_i \leq Mq_i, \forall i \tag{34}$$

$$\left[f_i^1(0) + \sum_{k=1}^{K} s_k^1 x_{i,k}\right] - \left[f_i^2(0) + \sum_{k=1}^{K} s_k^2 x_{i,k}\right] \leq v_i, \forall i \tag{35}$$

$$v_i \leq \left[f_i^1(0) + \sum_{k=1}^{K} s_k^1 x_{i,k}\right] - \left[f_i^2(0) + \sum_{k=1}^{K} s_k^2 x_{i,k}\right] + M(1 - q_i), \forall i \tag{36}$$

$$\sum\nolimits_{i,k} x_{i,k} \leq R, 0 \leq x_{i,k} \leq \frac{1}{K}, \forall i, k = 1 \ldots K \tag{37}$$

$$h_{i,k} \frac{1}{K} \leq x_{i,k}, \forall i, k = 1 \ldots K - 1 \tag{38}$$

$$x_{i,k+1} \leq h_{i,k}, \forall i, k = 1 \ldots K - 1 \tag{39}$$

$$q_i \in \{0, 1\}, h_{i,k} \in \{0, 1\}, \forall i, k = 1 \ldots K - 1. \tag{40}$$

where constraints (34 – 36) are equivalent to constraints (27 – 29) when $f^1(x_i)$ and $f^2(x_i)$ are replaced by its approximations. In addition, constraints (37 – 40) ensures that the segmentation is valid. In particular, $h_{i,k}$ is an auxiliary integer variable which indicates whether $x_{i,k} = \frac{1}{K}$ (when $h_{i,k} = 1$) or $x_{i,k+1} = 0$ (when $h_{i,k} = 0$).

D. Bound on Solution Quality

In providing a bound on solution quality of CUBIS, we consider the case where the lower and upper bound functions $L_i(x_i)$ and $U_i(x_i)$ are differentiable. We denote by $\overline{H}(\mathbf{x}, \boldsymbol{\beta})$ the resulting piecewise approximation of $H(\mathbf{x}, \boldsymbol{\beta})$. Also, lb and ub are the final upper bound and lower bound of the binary search in CUBIS, i.e., $ub - lb < \epsilon$. We first provide an error bound of the piecewise linear approximation for the defender utility $H(\mathbf{x}, \boldsymbol{\beta})$ in (15 – 17).

Lemma 1. *Given that β_i is determined as $\max\{0, c - U_i^d(x_i)\}$ for all i and for any utility value $c \in [\min_i P_i^d, \max_i R_i^d]$ (Proposition 3), the error bound for the approximation, $\left|H(\mathbf{x},\boldsymbol{\beta}) - \overline{H}(\mathbf{x},\boldsymbol{\beta})\right|$, is $O\left(\frac{1}{K}\right)$.*

Proof: In general, as β_i is determined as $\max\{0, c - U_i^d(x_i)\}$ for all target i, the objective function $H(\mathbf{x},\boldsymbol{\beta})$ has the form $H(\mathbf{x},\boldsymbol{\beta}) = \frac{N(\mathbf{x})}{D(\mathbf{x})}$. Similarly, the approximate function $\overline{H}(\mathbf{x},\boldsymbol{\beta})$ has the form $\overline{H}(\mathbf{x},\boldsymbol{\beta}) = \frac{\overline{N}(\mathbf{x})}{\overline{D}(\mathbf{x})}$ where $\overline{N}(\mathbf{x})$ and $\overline{D}(\mathbf{x})$ are piecewise linear approximations of the enumerator $N(\mathbf{x})$ and the denominator $D(\mathbf{x})$ respectively. The error bound is computed as the following:

$$\begin{aligned}&\left|H(\mathbf{x},\boldsymbol{\beta}) - \overline{H}(\mathbf{x},\boldsymbol{\beta})\right| = \left|\frac{N(\mathbf{x})}{D(\mathbf{x})} - \frac{\overline{N}(\mathbf{x})}{\overline{D}(\mathbf{x})}\right| \\ &= \left|\frac{N(\mathbf{x}) - \overline{N}(\mathbf{x})}{D(\mathbf{x})} + \overline{N}(\mathbf{x})\left[\frac{1}{\overline{D}(\mathbf{x})} - \frac{1}{D(\mathbf{x})}\right]\right| \\ &\leq \left|N(\mathbf{x}) - \overline{N}(\mathbf{x})\right| \frac{1}{|D(\mathbf{x})|} + \left|D(\mathbf{x}) - \overline{D}(\mathbf{x})\right| \frac{\left|\overline{N}(\mathbf{x})\right|}{|D(\mathbf{x})|\left|\overline{D}(\mathbf{x})\right|}\end{aligned} \tag{41}$$

As $N(\mathbf{x})$ and $D(\mathbf{x})$ are continuous functions over the compact set $\mathbf{X}$, the two terms $\frac{1}{|D(\mathbf{x})|}$ and $\frac{\left|\overline{N}(\mathbf{x})\right|}{|D(\mathbf{x})|\left|\overline{D}(\mathbf{x})\right|}$ are bounded for all $c \in [\min_i P_i^d, \max_i R_i^d]$. Thus, there exist constants $C^1, C^2 \geq 0$ such that the following inequality holds true:

$$\left|H(\mathbf{x},\boldsymbol{\beta}) - \overline{H}(\mathbf{x},\boldsymbol{\beta})\right| \leq C^1\left|N(\mathbf{x}) - \overline{N}(\mathbf{x})\right| + C^2\left|D(\mathbf{x}) - \overline{D}(\mathbf{x})\right| \tag{42}$$

On the other hand, the error for piecewise linearly approximating $D(\mathbf{x})$ satisfies:

$$\begin{aligned}\left|D(\mathbf{x}) - \overline{D}(\mathbf{x})\right| &= \left|\sum\nolimits_i L_i(x_i) - \overline{L}_i(x_i)\right| \\ &\leq \sum\nolimits_i \left|L_i(x_i) - \overline{L}_i(x_i)\right|\end{aligned} \tag{43}$$

In addition, suppose that $x_i \in \left[\frac{k-1}{K}, \frac{k}{K}\right]$ for some $1 \leq k \leq K$, according to the approximation, the following condition holds:

$$\begin{aligned}\min&\left\{L_i\left(\frac{k-1}{K}\right), L_i\left(\frac{k}{K}\right)\right\} \leq \overline{L}_i(x_i) \\ &\leq \max\left\{L_i\left(\frac{k-1}{K}\right), L_i\left(\frac{k}{K}\right)\right\}\end{aligned} \tag{44}$$

According to (44) and since $L_i(x_i)$ is a continuous function, there exists $\hat{x}_i \in \left[\frac{k-1}{K}, \frac{k}{K}\right]$ such that the value of $L_i(x_i)$ at $\hat{x}_i$ is equal to the value of its approximation at x_i, i.e., $L_i(\hat{x}_i) = \overline{L}_i(x_i)$. As a result, we obtain $\left|L_i(x_i) - \overline{L}_i(x_i)\right| = |L_i(x_i) - L_i(\hat{x}_i)|$. On the other hand, according to the Lagrange mean value theorem, there exists $a \in [\min\{x_i, \hat{x}_i\}, \max\{x_i, \hat{x}_i\}]$ such that the derivative at a satisfies the following equality:

$$L_i'(a) = \frac{L_i(x_i) - L_i(\hat{x}_i)}{x_i - \hat{x}_i} \tag{45}$$

Thus, we obtain the following inequality:

$$\begin{aligned}|L_i(x_i) - \overline{L}_i(x_i)| &= |L_i(x_i) - L_i(\hat{x}_i)| \\ &= |L_i'(a)|\,|x_i - \hat{x}_i| \leq \frac{1}{K}\max_{x_i \in [0,1]} |L_i'(x_i)|\end{aligned} \tag{46}$$

As a result, denote by $C_i = \max_{x_i \in [0,1]} |L_i'(x_i)|$ the maximum derivative value of $L_i(x_i)$ over the range $[0,1]$, by combining (43) and (46), we obtain an upper bound on the error for approximating $D(\mathbf{x})$ as the follows:

$$\left|D(\mathbf{x}) - \overline{D}(\mathbf{x})\right| \leq \frac{1}{K}\sum\nolimits_i C_i = O\left(\frac{1}{K}\right) \tag{47}$$

Similarly, we also have $\left|N(\mathbf{x}) - \overline{N}(\mathbf{x})\right|$ to be bounded by $O\left(\frac{1}{K}\right)$ for any value $c \in [\min_i P_i^d, \max_i R_i^d]$.

Finally, according to (41) and the error bounds $O\left(\frac{1}{K}\right)$ for two terms $\left|L_i(x_i) - \overline{L}_i(x_i)\right|$ and $\left|N(\mathbf{x}) - \overline{N}(\mathbf{x})\right|$, the error bound for the approximation, $\left|H(\mathbf{x},\boldsymbol{\beta}) - \overline{H}(\mathbf{x},\boldsymbol{\beta})\right|$, is $O\left(\frac{1}{K}\right)$. ■

The proof of Lemma 1 is described in Online Appendix A. Lemma 1 implies that when the number of segments, K, in piecewise linear approximation increases, the approximation error of $H(\mathbf{x},\boldsymbol{\beta})$ decreases and it converges to zero when $K \to +\infty$. Based on this lemma, we present the following two lemmas which in turn provide a lower bound for the approximated solution provided by CUBIS and an upper bound for the exact optimal solution of (15 – 17).

Lemma 2. *Suppose that $(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*)$ is the optimal solution of CUBIS and $H_\beta(\overline{\mathbf{x}}^*)$ is the defender's worst-case utility w.r.t $\overline{\mathbf{x}}^*$, then $H_\beta(\overline{\mathbf{x}}^*)$ has a lower bound of $lb + O\left(\frac{1}{K}\right)$.*

Proof: Denote by $S(\mathbf{x}, c) = \{\boldsymbol{\beta} \geq 0 : U_i^d(x_i) + \beta_i \geq c, \forall i\}$, then $S(\mathbf{x}, H(\mathbf{x},\boldsymbol{\beta}))$ is the set of feasible values of $\boldsymbol{\beta}$ for (15 – 17) given $\mathbf{x}$. The defender utility for playing the strategy $\overline{\mathbf{x}}^*$ in the worst case of uncertainty is computed by solving (15 – 17) given the defender's strategy $\mathbf{x} = \overline{\mathbf{x}}^*$:

$$H_\beta(\overline{\mathbf{x}}^*) = \max_{\boldsymbol{\beta} \in S(\overline{\mathbf{x}}^*, H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}))} H(\overline{\mathbf{x}}^*, \boldsymbol{\beta})$$

We consider the following two cases:

If $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}) \geq lb$ for some $\boldsymbol{\beta} \in S(\overline{\mathbf{x}}^*, lb)$, since $\inf\limits_{\boldsymbol{\beta} \in S(\overline{\mathbf{x}}^*, lb)} H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}) = -\infty$ and $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta})$ is continuous, the value range of $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta})$ covers the range $[-\infty, lb]$. Therefore, there exists $\boldsymbol{\beta}' \in S(\overline{\mathbf{x}}^*, lb)$ such that $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}') = lb$, hence this value of $\boldsymbol{\beta}'$ also belongs to the set $S(\overline{\mathbf{x}}^*, H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}'))$. We obtain $H_\beta(\overline{\mathbf{x}}^*) \geq H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}') = lb$.

Conversely, we suppose that $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}) < lb$ for all $\boldsymbol{\beta} \in S(\overline{\mathbf{x}}^*, lb)$. As $(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*)$ and lb is in turn the final solution and final lower bound in the binary search of CUBIS, the following condition holds: $U_i^d(\overline{x}_i^*) + \overline{\beta}_i^* \geq lb, \forall i$ (due to the feasibility of (**P1**)). This indicates that $\overline{\boldsymbol{\beta}}^* \in S(\overline{\mathbf{x}}^*, lb)$.

Since $H(\overline{\mathbf{x}}^*, \boldsymbol{\beta}) < lb$ for all $\boldsymbol{\beta} \in S(\overline{\mathbf{x}}^*, lb)$ and the fact that $\overline{\boldsymbol{\beta}}^* \in S(\overline{\mathbf{x}}^*, lb)$, we obtain the condition for the objective $H(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) < lb \leq U_i^d(\overline{x}_i^*) + \overline{\beta}_i^*, \forall i$. Therefore, $(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*)$ is a feasible solution of (15 – 17) and thus $H_\beta(\overline{\mathbf{x}}^*) \geq H(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*)$. Finally, by using the Lemma 1 and the fact that $\overline{H}(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) \geq lb$ (due to the feasibility of (**P1**)), we obtain a lower bound on $H_\beta(\overline{\mathbf{x}}^*) \geq H(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) = H(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) - \overline{H}(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) + \overline{H}(\overline{\mathbf{x}}^*, \overline{\boldsymbol{\beta}}^*) \geq lb + O\left(\frac{1}{K}\right)$. ■

Lemma 2 shows that CUBIS provides an approximated solution for (15 – 17) which has a lower bound of $lb + O\left(\frac{1}{K}\right)$ where lb is the final lower bound of CUBIS's binary search.

Lemma 3. *Suppose that $(\mathbf{x}^*, \boldsymbol{\beta}^*)$ is the ϵ-optimal solution of (15 – 17) which is obtained by following CUBIS 's procedure but without the piecewise linear approximation step. Then $H(\mathbf{x}^*, \boldsymbol{\beta}^*)$ has an upper bound of $ub + O\left(\frac{1}{K}\right)$.*

Proof: If $H(\mathbf{x}^*, \boldsymbol{\beta}^*) \leq ub$, obviously, the statement is true. Otherwise, if $H(\mathbf{x}^*, \boldsymbol{\beta}^*) > ub$, this indicates that $U_i^d(x_i^*) + \beta_i^* \geq ub$ for all i (due to the constraint (16)). Since ub is the final upper bound of CUBIS, the problem (**P1**) is infeasible for $c = ub$, meaning that $\overline{H}(\mathbf{x}, \boldsymbol{\beta}) \leq ub$ for all $(\mathbf{x}, \boldsymbol{\beta})$ that satisfy the constraints $U_i^d(x_i) + \beta_i \geq ub, \forall i$. Therefore, we have: $\overline{H}(\mathbf{x}^*, \boldsymbol{\beta}^*) \leq ub$. Finally, based on Lemma 1, we obtain the upper bound on $H(\mathbf{x}^*, \boldsymbol{\beta}^*) = H(\mathbf{x}^*, \boldsymbol{\beta}^*) - \overline{H}(\mathbf{x}^*, \boldsymbol{\beta}^*) + \overline{H}(\mathbf{x}^*, \boldsymbol{\beta}^*) \leq ub + O(\frac{1}{K})$. ■

Finally, based on Lemmas 2 and 3, we obtain the following theorem which provides a bound on CUBIS's approximated solution of computing the defender's optimal strategy against the worst case of uncertainty in (5).

Theorem 1. *CUBIS provides an $O\left(\epsilon + \frac{1}{K}\right)$-optimal solution of the maximin problem (5).*

Proof: By combining Lemma 2 and 3 and the fact that $|ub - lb| \leq \epsilon$, we obtain the difference, $|H(\mathbf{x}^*, \boldsymbol{\beta}^*) - H_\beta(\overline{\mathbf{x}}^*)|$, to be bounded by $O(\epsilon + \frac{1}{K})$. Since $H_\beta(\overline{\mathbf{x}}^*)$ is CUBIS's solution and $H(\mathbf{x}^*, \boldsymbol{\beta}^*)$ is the ϵ-optimal solution of (15 – 17), CUBIS provides an $O\left(\epsilon + \frac{1}{K}\right)$-optimal solution of (15 – 17) which is equivalent to the maximin problem (5). ■

V. SUMMARY

In summary, modeling the attacker's behavior is critical in security games in order to generating effective patrolling strategies for the defender. However, access to real-world data (used for learning an accurate behavioral model) is often limited, leading to uncertainty in attacker's behaviors while modeling. In this paper, we attempt to address the uncertainty in behavioral modeling of the attacker while providing the following key contributions: (i) we present a new uncertainty game model in which we consider a single behavioral model to capture decision making of the whole attacker population (instead of multiple behavioral models), while uncertainty intervals are integrated with the chosen model to capture behavioral uncertainty; and (ii) based on this game model, we propose a new efficient robust algorithm, CUBIS, that approximately computes the defender's optimal strategy which is robust to the uncertainty. We show that CUBIS provides an $O(\epsilon + \frac{1}{K})$-optimal solution where ϵ is the convergence threshold for the binary search and K is the number of segments in the piecewise linear approximation.

Acknowledgements: This research was supported by MURI Grant W911NF-11-1-0332 and by CREATE under grant number 2010-ST-061-RE0001.

REFERENCES

[1] M. Aghassi and D. Bertsimas. Robust game theory. *Mathematical Programming*, 107(1-2):231–273, 2006.

[2] N. Basilico, N. Gatti, and F. Amigoni. Leader-follower strategies for robotic patrolling in environments with arbitrary topologies. In *AAMAS*, pages 57–64, 2009.

[3] M. Brown, W. B. Haskell, and M. Tambe. Addressing scalability and robustness in security games with multiple boundedly rational adversaries. In *GameSec*, 2014.

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

[5] J. Cui and R. John. Adversary modeling in stackelberg security games. In *GameSec*, 2014.

[6] C. Kiekintveld, T. Islam, and V. Kreinovich. Security games with interval uncertainty. In *AAMAS*, 2013.

[7] D. Korzhyk, V. Conitzer, and R. Parr. Complexity of computing optimal stackelberg strategies in security resource allocation games. In *AAAI*, 2010.

[8] A. M. Lemieux. *Situational Prevention of Poaching*. Routledge, 2014.

[9] J. Letchford and Y. Vorobeychik. Computing randomized security strategies in networked domains. In *Applied Adversarial Reasoning and Risk Modeling*, 2011.

[10] D. McFadden. Conditional logit analysis of qualitative choice behavior. Technical report, 1972.

[11] D. McFadden. The measurement of urban travel demand. *Journal of public economics*, 3(4):303–328, 1974.

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

[13] T. H. Nguyen, A. X. Jiang, and M. Tambe. Stop the compartmentalization: unified robust algorithms for handling uncertainties in security games. In *AAMAS*, 2014.

[14] T. H. Nguyen, A. Sinha, and M. Tambe. Conquering adversary behavioral uncertainty in security games: An efficient modeling robust based algorithm. In *AAAI (Student Abstract)*, 2016.

[15] T. H. Nguyen, R. Yang, A. Azaria, S. Kraus, and M. Tambe. Analyzing the effectiveness of adversary modeling in security games. In *AAAI*, 2013.

[16] J. Pita, M. Jain, C. Western, C. Portway, M. Tambe, F. Ordonez, S. Kraus, and P. Paruchuri. Deployed armor protection: The application of a game theoretic model for security at the los angeles international airport. In *AAMAS*, 2008.

[17] E. Shieh, B. An, R. Yang, M. Tambe, C. Baldwin, J. DiRenzo, B. Maule, and G. Meyer. Protect: A deployed game theoretic system to protect the ports of the united states. In *AAMAS*, 2012.

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

[19] K. E. Train. *Discrete choice methods with simulation*. Cambridge university press, 2009.

[20] R. Yang, B. Ford, M. Tambe, and A. Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *AAMAS*, 2014.

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