Robust Solutions to Stackelberg Games: Addressing Bounded Rationality and Limited Observations in Human Cognition

James Pita, Manish Jain, Milind Tambe[a], Fernando Ordóñez[a,b], Sarit Kraus[c,d]

[a] *University of Southern California, Los Angeles, CA 90089*
[b] *University of Chile, Santiago, Chile*
[c] *Bar-llan University, Ramat-Gan 52900, Israel*
[d] *Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742*

Abstract

How do we build algorithms for agent interactions with human adversaries? Stackelberg games are natural models for many important applications that involve human interaction, such as oligopolistic markets and security domains. In Stackelberg games, one player, the leader, commits to a strategy and the follower makes her decision with knowledge of the leader's commitment. Existing algorithms for Stackelberg games efficiently find optimal solutions (leader strategy), but they critically assume that the follower plays optimally. Unfortunately, in many applications, agents face human followers (adversaries) who — because of their bounded rationality and limited observation of the leader strategy — may deviate from their expected optimal response. In other words, human adversaries' decisions are biased due to their bounded rationality and limited observations. Not taking into account these likely deviations when dealing with human adversaries may cause an unacceptable degradation in the leader's reward, particularly in security applications where these algorithms have seen deployment. The objective of this paper therefore is to investigate how to build algorithms for agent interactions with human adversaries.

To address this crucial problem, this paper introduces a new mixed-integer linear program (MILP) for Stackelberg games to consider human adversaries, incorporating: (i) novel *anchoring* theories on human perception of probability distributions and (ii) robustness approaches for MILPs to address human imprecision. Since this new approach considers human adversaries, traditional proofs of correctness or optimality are insufficient; instead, it is necessary to rely on empirical validation. To that end, this paper considers four settings based on real

deployed security systems at Los Angeles International Airport [43], and compares 6 different approaches (three based on our new approach and three previous approaches), in 4 different observability conditions, involving 218 human subjects playing 2960 games in total. The final conclusion is that a model which incorporates both the ideas of robustness and anchoring achieves statistically significant higher rewards and also maintains equivalent or faster solution speeds compared to existing approaches[1].

1. Introduction

In Stackelberg games, one player, the leader, commits to a strategy publicly before the remaining players, the followers, make their decision [18]. There are many multiagent security domains, such as attacker-defender scenarios and patrolling, where these types of commitments are necessary by the security agent [3, 8, 26, 40] and it has been shown that Stackelberg games appropriately model these commitments [39, 43]. For example, in an airport setting there may be six terminals serving passengers, but only four bomb sniffing canine units to patrol the terminals. In this scenario the canine units decide on a randomized patrolling strategy over these six terminals first, while their adversaries conduct surveillance and act taking this committed strategy into account. Indeed, Stackelberg games are at the heart of the ARMOR system deployed at the Los Angeles International Airport (LAX) to schedule security personnel since August 2007 [39, 43] and have been deployed for the Federal Air Marshals service since October 2009 [26, 56]. Moreover, these games have potential applications for network routing, pricing in transportation systems and many others [11, 28].

Existing algorithms for Bayesian Stackelberg games find optimal solutions considering an *a priori* probability distribution over possible follower types [12, 39]. Unfortunately, to guarantee optimality, these algorithms make strict assumptions on the underlying games, namely that the players are perfectly rational and that the followers perfectly observe the leader's strategy. However, these assumptions rarely hold in real-world domains, particularly when dealing with humans.

[1] This paper significantly extends our previous conference publication [44]. It provides significant new experimental results and new analysis, additional theoretical results and proofs related to zero sum games, additional explanations and related work. In our previous publication [44] we presented three algorithms named BRASS, GUARD, and COBRA. It can be shown that BRASS and GUARD are special cases of COBRA and so we present all three as modifications of COBRA here.

Of specific interest are the security domains mentioned earlier (e.g. LAX) — even though an automated program may determine an optimal leader (security personnel) strategy, it must take into account a human follower (adversary). Such human adversaries may not be utility maximizers, computing optimal decisions. Instead, their decisions may be governed by their bounded rationality [51] which causes them to deviate from their expected optimal strategy. Humans may also suffer from limited observability of the security personnel's strategy, giving them a false impression of that strategy. In other words, when making decisions based on their own cognitive abilities, humans are biased due to their bounded rationality and inability to obtain complete sets of observations. Thus, a human adversary may not respond with the game theoretic optimal choice, causing the leader to face uncertainty over the gamut of adversary's actions. Therefore, in general, the leader in a Stackelberg game must commit to a strategy considering three different types of uncertainty: (i) adversary response uncertainty due to her bounded rationality where the adversary may not choose the utility maximizing optimal strategy; (ii) adversary response uncertainty due to her limitations in appropriately observing the leader strategy; (iii) adversary reward uncertainty modeled as different reward matrices with a Bayesian *a priori* distribution assumption, i.e. a Bayesian Stackelberg game. While existing algorithms handle the third type of uncertainty [12, 39], these models can give a severely under-performing strategy when the adversary deviates because of the first two types of uncertainty. This degradation in leader rewards may be unacceptable in certain domains.

To overcome this limitation, this paper proposes a new algorithm based on a mixed-integer linear program (MILP). The major contribution of this new MILP is in providing a fundamentally novel integration of key ideas from: (i) previous best known algorithms from the multiagent literature for solving Bayesian Stackelberg games; (ii) robustness approaches for games from robust optimization literature [1, 35]; (iii) anchoring theories on human perception of probability distributions from psychology [15, 16, 47]. While the robustness approach helps address human response imprecision, anchoring, which is an expansion of general support theory [57] on how humans attribute probabilities to a discrete set of events, helps address limited observational capabilities. To the best of our knowledge, the effectiveness of the combination of these ideas has not been explored in the context of Stackelberg games (or any other games). By uniquely incorporating these ideas our goal is to defend against the sub-optimal choices that humans may make due to bounded rationality or observational limitations. This MILP complements the prior algorithms for Bayesian Stackelberg games, handling all three types of un-

certainty mentioned[2].

Since this algorithm is centered on addressing non-optimal and uncertain human responses, traditional proofs of correctness and optimality are insufficient: it is necessary to experimentally test this new approach against existing approaches. Experimental analysis with human subjects allows us to show how this algorithm is expected to perform against human adversaries compared to previous approaches. To that end, we experimentally tested our new algorithm to determine its success by considering four settings based on real deployed security systems at LAX [43]. In all four settings, 6 different approaches were compared (three based on the new algorithm, one existing approach, and two baseline approaches), in 4 different observability conditions. These experiments involved 218 human subjects playing 2960 games in total and yielded statistically significant results showing that our new algorithm substantially outperformed existing methods when dealing with human adversaries. Runtime results were also gathered from our new algorithm against previous approaches showing that its solution speeds are equivalent to or faster than previous approaches. Based on these results we conclude that, while theoretically optimal, existing algorithms for Bayesian Stackelberg games may need to be significantly modified for security domains. They are not only outperformed by our new algorithm, which incorporates both robustness approaches and anchoring theories, but also may be outperformed by simple baseline algorithms when playing against human adversaries in certain cases. Our results also show that the anchoring bias may play a particularly important role in human responses under all observability conditions, and exploiting this bias can lead to significant performance improvements. This is an important conclusion since existing algorithms have seen real deployment such as at Los Angeles International Airport (LAX) and the Federal Air Marshals service [26, 43].

The organization of the paper is as follows: Section 2 describes necessary background information on Bayesian Stackelberg games, an existing algorithm for solving Bayesian Stackelberg games and baseline algorithms that are used in the experimental section. In Section 3 we introduce the concepts behind our new algorithm and the MILP in detail. In Section 4 we present some propositions that show under certain conditions our robust models become equivalent to theoretically optimal solvers. Section 5 presents our experimental analysis of our new MILP against the algorithms presented in Section 2. In Section 6 we provide

[2]Although the MILP presented handles all three types of uncertainty, the focus of this paper is handling the first two types of uncertainty.

related work and finally in Section 7 we provide a summary of our work.

2. Background

2.1. Stackelberg Game

In a Stackelberg game, a leader commits to a strategy first, and then a follower selfishly optimizes her reward, *considering the action chosen by the leader.* To see the advantage of being the leader in a Stackelberg game, consider a simple game with the payoff table as shown in Table 1. The leader is the row player and the follower is the column player. The only pure-strategy Nash equilibrium for this game is when the leader plays *a* and the follower plays *c*, which gives the leader a payoff of 2; in fact, for the leader, playing *b* is strictly dominated. However, if the leader can commit to playing *b* before the follower chooses her strategy, then the leader will obtain a payoff of 3, since the follower would then play *d* to ensure a higher payoff for herself. If the leader commits to a uniform mixed strategy of playing *a* and *b* with equal (0.5) probability, then the follower will play *d*, leading to a payoff for the leader of 3.5.

	c	d
a	2,1	4,0
b	1,0	3,2

Table 1: Payoff table for example Stackelberg game.

In this article we assume that a leader commits to a mixed strategy and the follower may or may not fully observe this strategy before making a decision. Such a commitment to a mixed strategy models a real-world situation where police commit to a randomized patrolling strategy first. Given this commitment, adversaries can choose whether or not and how much surveillance to conduct of this mixed strategy. Even with knowledge of this mixed strategy, the adversaries have no specific knowledge of what the police may do on a particular day however. They only have knowledge of the mixed strategy the police will use to decide their resource allocations for that day.

2.2. Bayesian Stackelberg Game

We now define a Bayesian Stackelberg game. A Bayesian game contains a set of N agents, and each agent n must be one of a given set of types. This paper considers a Bayesian Stackelberg game that was inspired by the security domain

presented by LAX [43]. This game has two agents, the defender (leader) and the attacker (follower). We denote the set of possible leader types by Θ and the set of possible follower types by Ψ. For the security games of interest in this paper, we assume that there is only one leader type (e.g. only one police force), although there can be multiple follower types (e.g. multiple adversary types trying to infiltrate security). Therefore, while Θ contains only one element, there is no such restriction on Ψ. However, the leader does not know the follower's type. For each agent (defender or attacker), there is a set of strategies. We denote the leader's set of pure strategies by $\sigma_i \in \Sigma_\Theta$ and the follower's set of pure strategies by $\sigma_j \in \Sigma_\Psi$. Payoffs for each player (defender and attacker) are defined over all possible joint strategy outcomes: $\Omega_\Theta : \Sigma_\Theta \times \Sigma_\Psi \rightarrow \Re$ for the leader and similarly for the follower. The payoff functions are extended to mixed strategies in the standard way by taking the expectation over pure-strategy outcomes. Our goal is *to find the optimal mixed strategy* for the leader to commit to, given that the follower may know this mixed strategy when choosing her strategy.

2.3. Stackelberg Equilibria

There are two types of unique Stackelberg equilibria we are interested in, first proposed by Leitmann [30], and typically called "strong" and "weak" after Breton et. al. [6]. The strong form assumes that the follower will always choose the best strategy for the leader in cases of indifference (i.e. a tie), while the weak form assumes that the follower will choose the worst strategy for the leader. A strong Stackelberg equilibrium exists in all Stackelberg games, but a weak Stackelberg equilibrium may not. In addition, the leader can often induce the favorable strong equilibrium by selecting a strategy arbitrarily close to the equilibrium that causes the follower to strictly prefer the desired favorable strategy for the leader [59]. We focus on the strong Stackelberg equilibrium, as that is the most commonly adopted concept in the related literature [12, 26, 36, 39] due to the key existence results; however, we will also examine situations where the follower chooses the worst strategy for the leader.

2.4. DOBSS

We now describe the Decomposed Optimal Bayesian Stackelberg Solver (DOBSS) in detail as it provides a starting point for the algorithms we develop in the next section. While the problem of choosing an optimal strategy for the leader in a Stackelberg game is NP-hard for a Bayesian game with multiple follower types [12], researchers have continued to provide practical improvements. DOBSS is

currently the most efficient general Stackelberg solver [39] and is in use for security scheduling at the Los Angeles International Airport. It operates directly on the compact Bayesian representation, giving speedups over the multiple linear programs method [12] which requires conversion of the Bayesian game into a normal-form game by the Harsanyi transformation [21]. In particular, DOBSS obtains a decomposition scheme by exploiting the property that follower types are independent of each other. The key to the DOBSS decomposition is the observation that evaluating the leader strategy against a Harsanyi-transformed game matrix is equivalent to evaluating against each of the game matrices for the individual follower types and then obtaining a weighted sum.

We first present DOBSS in its most intuitive form as a Mixed-Integer Quadratic Program (MIQP); we then present a linearized equivalent Mixed-Integer Linear Program (MILP). The DOBSS model explicitly represents the actions by the leader and the *optimal* actions for the follower types in the problem solved by the leader. Note that we need to consider only the reward-maximizing pure strategies of the follower types, since for a given fixed mixed strategy x of the leader, each follower type faces a problem with fixed linear rewards. If a mixed strategy is optimal for the follower, then so are all the pure strategies in support of that mixed strategy.

Thus, we denote by x the leader's policy, which consists of a probability distribution over the leader's pure strategies $\sigma_i \in \Sigma_\Theta$. Hence, the value x_i is the proportion of times in which pure strategy $\sigma_i \in \Sigma_\Theta$ is used in the policy. Similarly, we denote by q^l a probability distribution for follower type $l \in \Psi$ over the possible pure strategies where q^l_j is the probability of taking strategy $\sigma_j \in \Sigma_\Psi$ for follower type l. We denote by X and Q the index sets of the leader and follower l's pure strategies, respectively. We also index the payoff matrices of the leader and each of the follower types $l \in \Psi$ by the matrices R^l and C^l where R^l_{ij} and C^l_{ij} are the rewards obtained if the leader takes strategy $\sigma_i \in \Sigma_\Theta$ and the follower of type l takes strategy $\sigma_j \in \Sigma_\Psi$. Let M be a large positive number; constraint 3 in the MIQP below requires that the variable a^l be set to the maximum reward follower type l can obtain given the current policy x taken by the leader. Given prior probabilities p^l, with $l \in \Psi$, of facing each follower type, the leader solves the following:

$$\max_{x,q,a} \quad \sum_{i\in X}\sum_{l\in\Psi}\sum_{j\in Q} p^l R^l_{ij} x_i q^l_j \tag{1}$$

$$\text{s.t.} \quad \sum_{i\in X} x_i = 1 \tag{2}$$

$$\sum_{j\in Q} q^l_j = 1 \qquad \forall l \in \Psi \tag{3}$$

$$0 \leq (a^l - \sum_{i\in X} C^l_{ij} x_i) \leq (1 - q^l_j)M \qquad \forall l \in \Psi, j \in Q \tag{4}$$

$$x_i \in [0 \dots 1] \qquad \forall i \in X \tag{5}$$

$$q^l_j \in \{0, 1\} \qquad \forall l \in \Psi, j \in Q \tag{6}$$

$$a^l \in \Re \qquad \forall l \in \Psi \tag{7}$$

Here for a leader strategy x and a strategy q^l for each follower type, the objective represents the expected reward for the leader considering the *a priori* distribution over different follower types p^l. The first and the fourth constraints define the set of feasible solutions $x \in X$ as a probability distribution over the set of strategies $\sigma_i \in \Sigma_\Theta$. Constraints 2 and 5 limit the vector of strategies of follower type l, q^l, to be a pure strategy over the set Q (that is each q^l has exactly one coordinate equal to one and the rest equal to zero). The two inequalities in constraint 3 ensure that $q^l_j = 1$ only for a strategy j that is optimal for follower type l. Indeed this is a linearized form of the optimality conditions for the linear programming problem solved by each follower type. We explain constraint 3 as follows: note that the leftmost inequality ensures that $\forall l \in \Psi, j \in Q, a^l \geq \sum_{i\in X} C^l_{ij} x_i$. This means that given the leader's policy x, a^l is an upper bound on follower type l's reward for any strategy. The rightmost inequality is inactive for every strategy where $q^l_j = 0$, since M is a large positive quantity. For the strategy that has $q^l_j = 1$ this inequality states $a^l \leq \sum_{i\in X} C^l_{ij} x_i$, which combined with the previous inequality shows that this strategy must be optimal for follower type l.

We can linearize the quadratic programming problem 1 through the change of variables $z^l_{ij} = x_i q^l_j$, thus obtaining the following mixed integer linear programming problem:

$$\max_{q,z,a} \quad \sum_{i\in X}\sum_{l\in L}\sum_{j\in Q} p^l R^l_{ij} z^l_{ij} \tag{8}$$

$$\text{s.t.} \quad \sum_{i\in X}\sum_{j\in Q} z^l_{ij} = 1 \qquad \forall l \in \Psi \tag{9}$$

$$\sum_{j\in Q} z^l_{ij} \leq 1 \qquad \forall l \in \Psi, i \in X \tag{10}$$

$$q^l_j \leq \sum_{i\in X} z^l_{ij} \leq 1 \qquad \forall l \in \Psi, j \in Q \tag{11}$$

$$\sum_{j\in Q} q^l_j = 1 \qquad \forall l \in \Psi \tag{12}$$

$$0 \leq (a^l - \sum_{i\in X} C^l_{ij}(\sum_{h\in Q} z^l_{ih})) \leq (1 - q^l_j)M \qquad \forall l \in \Psi, j \in Q \tag{13}$$

$$\sum_{j\in Q} z^l_{ij} = \sum_{j\in Q} z^1_{ij} \qquad \forall l \in \Psi, i \in X \tag{14}$$

$$z^l_{ij} \in [0 \ldots 1] \qquad \forall l \in \Psi, i \in X, j \in Q \tag{15}$$

$$q^l_j \in \{0, 1\} \qquad \forall l \in \Psi, j \in Q \tag{16}$$

$$a^l \in \Re \qquad \forall l \in \Psi \tag{17}$$

Our implementation of DOBSS solves this mixed integer linear program, which was shown to be equivalent to 1 and the equivalent Harsanyi transformed Stackelberg game in [39]. Problem 8 can be solved using efficient integer programming packages. For a more in depth explanation of DOBSS please see [39].

2.5. Baseline Algorithms

For completeness this paper includes both a uniformly random strategy and a MAXIMIN strategy against human opponents as a baseline against the performance of both existing algorithms, such as DOBSS, and our new algorithm. Proposed algorithms must outperform the two baseline algorithms to provide benefits.

2.5.1. UNIFORM

UNIFORM is the most basic method of randomization which just assigns an equal probability of taking each strategy $\sigma_i \in \Sigma_\Theta$ (a uniform distribution).

2.5.2. MAXIMIN

MAXIMIN is a traditional approach which assumes the follower may take any of the available actions. The objective of the following LP is to maximize the minimum reward, γ, the leader will obtain irrespective of the follower's action:

$$\max \quad \sum_{l \in L} p^l \gamma_l \tag{18}$$

$$\text{s.t.} \quad \sum_{i \in X} x_i = 1 \tag{19}$$

$$\sum_{i \in X} R^l_{ij} x_i \geq \gamma_l \quad \forall l \in \Psi, j \in Q \tag{20}$$

$$x_i \in [0 \dots 1] \quad \forall i \in X \tag{21}$$

$$\tag{22}$$

3. Robust Algorithm

There are two fundamental assumptions underlying current algorithms for Stackelberg games, including DOBSS. First, the follower is assumed to act with infallible utility maximizing rationality, choosing the absolute optimal among her strategies. Second, if the follower faces a tie in her strategies' rewards, she will break it in favor of the leader, choosing the one that gives a higher reward to the leader (a strong Stackelberg equilibrium). This standard assumption is also shown to follow from the follower's rationality and optimal response under some conditions [59]. Unfortunately, in many real-world domains, agents can face human followers who may not respond optimally: this may be caused by their bounded rationality or their uncertainty regarding the leader strategy. In essence, the leader faces uncertainty over follower responses — the follower may not choose the optimal response, but from a range of possible responses — potentially significantly degrading leader rewards. No *a priori* probability distributions are available or assumed for this follower response uncertainty.

To remedy this situation, we draw inspiration from robust optimization methodology, in which the decision maker optimizes against the worst outcome over some uncertainty [1, 34]. We also draw inspiration from psychological support theory for human decision making when humans are given a discrete set of actions and an unknown probability function over those actions [47, 57]. In the presented Stackelberg problem, the leader will make a robust decision by considering that the follower, who may not follow the utility maximizing rationality, could choose a strategy from her range of possible responses that degrades the leader reward the most or that she could choose a strategy that is based on her limited observations.

To that end, this paper introduces a mixed-integer linear program (MILP), COBRA (Combined Observability and Rationality Assumption), that builds on the Bayesian Stackelberg game model in DOBSS. This MILP continues to handle

adversary reward uncertainty in the same fashion as DOBSS. Along with handling reward uncertainty, it also addresses the uncertainty that may arise from human imprecision in choosing the expected optimal strategy due to bounded rationality and limited observations. Namely it introduces the idea of robust responses to ε-optimal follower responses into DOBSS and Stackelberg games in general. It also utilizes the concept of anchoring biases to protect against limited observation conditions, handling observational uncertainty. We first describe in depth the key ideas behind our new approach and then incrementally define the MILP that uses them.

3.1. Key Ideas

The two main ideas in our new algorithm is addressing boundedly rational opponents and anchoring biases those opponents may have.

3.1.1. Bounded Rationality

Our new algorithm assumes that the follower is boundedly rational and may not strictly maximize utility. As a result, the follower may select an ε-optimal response strategy, i.e. the follower may choose any of the responses within ε-reward of her optimal strategy. This choice may be caused by a variety of reasons, but we attempt to guard against the choices that fall within this ε-bound of the optimal response. Similar approaches have been applied in simultaneous move games and are known as ε-equilibria [55], but these ε-equilibria have not been considered in Stackelberg settings and our method differs in that we only consider ε-deviations of the follower while the leader remains perfectly rational. More specifically, given multiple possible ε-optimal responses, the robust approach is to assume that the follower could choose the one that provides the leader the worst reward — not necessarily because the follower attends to the leader reward, but to robustly guard against the worst-case outcome. In an adversarial setting, such as the one we face, handling the worst case outcome may be in the best interest of the leader. This worst case assumption contrasts with those of other Stackelberg solvers which assume the follower will play a strong Stackelberg equilibrium (choosing a strategy that favors the leader in the case of a tie) [12, 39], making our approach novel to address human followers.

3.1.2. Anchoring Theory

Support theory is a theory of subjective probability [57] and has been used to introduce anchoring biases [15, 16, 47]. An anchoring bias is when, given no information about the occurrence of a discrete set of events, humans will tend to

assign an equal weight to the occurrence of each event (a uniform distribution). This is also referred to as giving full support to the ignorance prior [16]. It has been shown through extensive experimentation that humans are particularly susceptible to giving full support to the ignorance prior before they are given any information and that, once given information, they are slow to update away from this assumption [15, 16, 47]. Thus they leave some support, $\alpha \in [0 \dots 1]$, on the ignorance prior and the rest, $1 - \alpha$, on the occurrence they have actually viewed. As humans become more confident in what they are viewing, this bias begins to diminish, decreasing the value of α.

Models have been proposed to address this bias and predict what probability a human will assign to a particular event x from a set of events X based on the evaluative assessment (i.e. assessment based on events actually viewed) they have made for the occurrence of that event. Let x represent a particular event, $X \backslash x$ represent the remaining events possible, let $P(x), P(X \backslash x)$ be the real probabilities and $P(x'), P((X \backslash x)')$ represent the probability a human assigns to event x and $X \backslash x$ respectively. One model, [16], defines the human estimated probabilities with the following ratios $P(x')/P((X \backslash x)') = (|x|/|X \backslash x|)^{\alpha}(P(x)/P(X \backslash x))^{1-\alpha}$. Here, $|x|/|X \backslash x|$ is the ratio in the case of uniform probabilities and represents the ignorance prior. The α value indicates the relative contribution of these two sources of information. Note that as α approaches 1, the estimated probability converges on the uniform assumption, while when α approaches 0, it is closer to the true probability distribution. Research suggests that as people gain more relevant knowledge they will give less support to the ignorance prior and more support to evaluative assessment thus decreasing the value of α [15, 16, 47].

An alternative model assumes the estimated probabilities are directly calculated using a simple linear model [14, 57]: $P(x') = \alpha(1/|X|) + (1 - \alpha)P(x)$. We commandeer this anchoring bias for Stackelberg games to determine how a human follower may perceive the leader strategy. For example, in the game shown in Table 1, suppose the leader strategy was to play *a* with a probability of 0.8 and *b* with 0.2. Anchoring bias would predict that in the absence of any information ($\alpha = 1$), humans will assign a probability of 0.5 to each of *a* and *b*, and will only update this belief (alter the value of α) after observing the leader strategy for some time. Although these may not be the only possible models for determining anchoring bias, they are standard in the related literature [16, 57] and the linear model is ideal since the odds form model is not easily representable in an MILP.

As an alternative approach we could use Bayesian updating to predict how humans will perceive the probability of each event from a set of events after obtaining some observations. However, there is more support in the literature that

humans act according to subjective probability and anchoring biases rather than performing Bayesian updating when evaluating evidence [15, 16, 24, 47, 57]. Also, using Bayesian updates requires tracking which observations the humans have received while in the real world, security forces will not be aware of which days adversaries take observations and which days they do not. Anchoring bias is more general in that it allows us to work with just an estimate of how many observations we believe an adversary will take rather than which specific observations she takes. Specifically we assign a value to α based on how much evidence we think the human will receive. If the human adversary is expected to observe our policy frequently and carefully then α will be low while if we suspect she will not have many observations of our policy, α will be high.

3.2. COBRA(α,ε)

COBRA(α,ε) is the new algorithm we introduce in this article. To introduce it in steps, we will first introduce two simplified versions of our algorithm which we will refer to as COBRA(0,ε) and COBRA(α,0). COBRA(0,ε) deals only with bounded rationality and COBRA(α,0) deals only with observational uncertainty. After introducing each of these pieces individually we will combine them into a single algorithm that can handle both types of uncertainty which we refer to as COBRA(α,ε). For each of these algorithms α and ε represent two parameters that can be adjusted.

3.2.1. COBRA(0,ε)

COBRA(0,ε) considers the case of a boundedly rational follower, where it maximizes the minimum reward it obtains from any ε-optimal response from the follower. In the following MILP, we use the same variable notation as in DOBSS. In addition, the variables h_j^l identify the optimal strategy for follower type l with a value of a^l in the third and fourth constraints. Variables q_j^l represent all ε-optimal strategies for follower type l; the second constraint now allows selection of more than one ε-optimal strategy per follower type. The fifth constraint ensures that $q_j^l = 1$ for every action j such that $a^l - \sum_{i \in X} C_{ij}^l x_i < \varepsilon$, since in this case the middle term in the inequality is less than ε and the left inequality is then only satisfied if $q_j^l = 1$. This robust approach required the design of a new objective and an additional constraint. The sixth constraint helps define the objective value against follower type l, γ_l, which must be lower than any leader reward for all actions $q_j^l = 1$, as opposed to the DOBSS formulation which has only one action $q_j^l = 1$. Setting γ_l to the minimum leader reward allows COBRA(0,ε) to robustly guard against the worst case scenario. The new MILP is as follows:

$$\max_{x,q,h,a,\gamma} \quad \sum_{l \in L} p^l \gamma_l \tag{23}$$

$$\text{s.t.} \quad \sum_{i \in X} x_i = 1 \tag{24}$$

$$\sum_{j \in Q} q_j^l \geq 1 \qquad \forall l \in \Psi \tag{25}$$

$$\sum_{j \in Q} h_j^l = 1 \qquad \forall l \in \Psi \tag{26}$$

$$0 \leq (a^l - \sum_{i \in X} C_{ij}^l x_i) \leq (1 - h_j^l)M \qquad \forall l \in \Psi, j \in Q \tag{27}$$

$$\varepsilon(1 - q_j^l) \leq a^l - \sum_{i \in X} C_{ij}^l x_i \leq \varepsilon + (1 - q_j^l)M \qquad \forall l \in \Psi, j \in Q \tag{28}$$

$$M(1 - q_j^l) + \sum_{i \in X} R_{ij}^l x_i \geq \gamma_l \qquad \forall l \in \Psi, j \in Q \tag{29}$$

$$h_j^l \leq q_j^l \qquad \forall l \in \Psi, j \in Q \tag{30}$$

$$x_i \in [0 \dots 1] \qquad \forall i \in X \tag{31}$$

$$q_j^l, h_j^l \in \{0, 1\} \qquad \forall l \in \Psi, j \in Q \tag{32}$$

$$a^l \in \Re \qquad \forall l \in \Psi \tag{33}$$

3.3. COBRA(α,0)

COBRA(α,0) considers the case where the human follower is perfectly rational, but faces limited observations. COBRA(α,0) draws upon the theory of anchoring biases mentioned previously to help address the human uncertainty that arises from such limited observation. It deals with two strategies: (i) the real leader strategy (x) and (ii) the perceived strategy by the follower (x'), where x' is defined by the linear model presented earlier. Thus, x_i is replaced in the third constraint with x'_i and x'_i is accordingly defined as $x'_i = \alpha(1/|X|) + (1 - \alpha)x_i$. The justification for this replacement is as follows. First, this particular constraint ensures that the follower maximizes her reward. Since the follower believes x' to be the leader strategy then she will choose her strategy according to x' and not x. Second, given this knowledge, the leader can find the follower's responses based on x' and optimize its actual strategy x against this strategy. Since x' is a combination of the support for x and the support toward the ignorance prior, COBRA(α,0) is able to find a strategy x that will maximize the leader's reward based on the relative contribution of these two sources of support[3]. For consistency among our

[3]In an alternative generalized approach we could allow for an α^l for each adversary type $l \in \Psi$. However, there are two considerations to take into account. First, this generalization comes at a cost: it would introduce additional parameters into COBRA(α,0), potentially increasing the burden on future designers to select an appropriate α for each follower type. Second, given that we are

new approaches we use the same objective introduced in MILP 23. The new MILP then is as follows:

$$\max_{l} \quad \sum_{l \in L} p^l \gamma_l \tag{34}$$

$$\text{s.t.} \quad \sum_{i \in X} x_i = 1 \tag{35}$$

$$\sum_{j \in Q} q_j^l = 1 \qquad \forall l \in \Psi \tag{36}$$

$$0 \leq (a^l - \sum_{i \in X} C_{ij}^l x_i') \leq (1 - q_j^l)M \qquad \forall l \in \Psi, j \in Q \tag{37}$$

$$M(1 - q_j^l) + \sum_{i \in X} R_{ij}^l x_i \geq \gamma_l \qquad \forall l \in \Psi, j \in Q \tag{38}$$

$$x_i \in [0 \dots 1] \qquad \forall i \in X \tag{39}$$

$$q_j^l \in \{0, 1\} \qquad \forall l \in \Psi, j \in Q \tag{40}$$

$$a^l \in \Re \qquad \forall l \in \Psi \tag{41}$$

$$x_i' = \alpha(1/|X|) + (1 - \alpha)x_i \qquad \forall i \in X \tag{42}$$

3.4. COBRA(α,ε)

COBRA(α,ε) is an MILP that combines both a bounded rationality assumption and an observational uncertainty assumption. This is achieved by incorporating the alterations made in COBRA(α,0) and COBRA(0,ε) into a single MILP. Namely, COBRA(α,ε) includes both the ε parameter and the α parameter from MILP (23) and MILP (34) respectively. The MILP that follows is identical to MILP (23) except that in the fourth and fifth constraints, x_i is replaced with x_i' as it is in MILP (34). The justification for this replacement is the same as in MILP (34). The new MILP then is as follows:

addressing human follower types, presumably they may share a similar anchoring bias even if they differ in their type, i.e. the benefits of this generalization are not clear-cut. Thus, it is not completely clear that the benefits of such a generalization would be offset by its cost. Exploring these trade-offs is left for future work.

$$\max_{x,q,h,a,\gamma} \quad \sum_{l \in L} p^l \gamma_l \tag{43}$$

$$\text{s.t.} \quad \sum_{i \in X} x_i = 1 \tag{44}$$

$$\sum_{j \in Q} q_j^l \geq 1 \qquad \forall l \in \Psi \tag{45}$$

$$\sum_{j \in Q} h_j^l = 1 \qquad \forall l \in \Psi \tag{46}$$

$$0 \leq (a^l - \sum_{i \in X} C_{ij}^l x_i') \leq (1 - h_j^l)M \qquad \forall l \in \Psi, j \in Q \tag{47}$$

$$\varepsilon(1 - q_j^l) \leq a^l - \sum_{i \in X} C_{ij}^l x_i' \leq \varepsilon + (1 - q_j^l)M \qquad \forall l \in \Psi, j \in Q \tag{48}$$

$$M(1 - q_j^l) + \sum_{i \in X} R_{ij}^l x_i \geq \gamma_l \qquad \forall l \in \Psi, j \in Q \tag{49}$$

$$h_j^l \leq q_j^l \qquad \forall l \in \Psi, j \in Q \tag{50}$$

$$x_i \in [0 \dots 1] \qquad \forall i \in X \tag{51}$$

$$q_j^l, h_j^l \in \{0, 1\} \qquad \forall l \in \Psi, j \in Q \tag{52}$$

$$a^l \in \Re \qquad \forall l \in \Psi \tag{53}$$

$$x_i' = \alpha(1/|X|) + (1 - \alpha)x_i \qquad \forall i \in X \tag{54}$$

3.5. Complexity

It has been shown that finding an optimal solution in a Bayesian Stackelberg game is NP-hard [12] and thus DOBSS, COBRA(α,0), COBRA(0,ε), and COBRA(α,ε) are MILPs that face an NP-hard problem. A number of effective solution packages for MILPs can be used, but their performance depends on the number of integer variables. DOBSS and COBRA(α,0) consider $|Q|\,|L|$ integer variables, while COBRA(0,ε) and COBRA(α,ε) double that. MAXIMIN on the other hand is a linear programming problem that can be solved in polynomial time. Thus we anticipated MAXIMIN will have the lowest running time per problem instance, followed by DOBSS and COBRA(α,0) with COBRA(0,ε) and COBRA(α,ε) close behind. However, as shown in runtime results, this was not observed in practice.

4. Equivalences Between Models

In this section we suggest and prove equivalencies between our robust model and DOBSS under certain conditions. First we will demonstrate how DOBSS can be reformulated with the same objective as COBRA(α,0). After this reformulation we will show how altering the parameters α and ε can cause COBRA(α,ε), COBRA(α,0), COBRA(0,ε), and DOBSS to produce identical results (i.e. identical mixed strategies).

Observation 1. *When $\varepsilon = 0$ and $\alpha = 0$ then MILPs (8) and (43) are equivalent.*

Proof 1. *It follows from the definition of x'_i that when $\alpha = 0$ then $x'_i = x_i$ since the follower is assumed to once again perfectly observe and believe the leader strategy x_i. Note that if $\varepsilon = 0$ the inequality in the fifth constraint of MILP (43) is the same expression as the inequality in the fourth constraint with q^l_j substituted for h^l_j. Since the objective of MILP (43) is to maximize the leader reward this means that h^l_j will be selected as the follower's optimal response that maximizes the leader reward (i.e. a strong Stackelberg equilibrium) and q^l_j will be set to the same. Introducing additional $q^l_k = 1$ where $k \neq j$ would only serve to reduce the reward and thus would not be an optimal solution to the MILP. We will show that the two MILPs (8) and (43) attain the same optimal objective function value.*

To show that solution to MILP (43) $\geq$ solution to MILP (8), consider (q, z, a) a feasible solution for MILP (8). We define $\bar{x}_i = \sum_{j \in Q} z^l_{ij}$, $\bar{q} = \bar{h} = q$, $\bar{a} = a$, and $\bar{\gamma}_l = \sum_{i \in X} \sum_{j \in Q} R^l_{ij} z^l_{ij}$. From the first through third constraints and the sixth constraint in MILP (8) we can show that $z^l_{ij} = 0$ for all j such that $q^l_j = 0$ and thus that $\bar{x}_i = z^l_{ij}$ for all j such that $q^l_j = 1$. This implies that $\bar{\gamma}_l = \sum_{i \in X} R^l_{ij} \bar{x}_i$ for the j such that $q^l_j = 1$ in MILP (8) and it is then easy to verify that $(\bar{x}, \bar{q}, \bar{h}, \bar{a}, \bar{\gamma})$ is feasible for MILP (43) with the same objective function value of (q, z, a) in MILP (8).

For solution to MILP (8) $\geq$ solution to MILP (43), consider (x, q, h, a, γ) feasible for MILP (43). Define $\bar{q} = h$, $\bar{z}^l_{ij} = x_i h^l_j$, and $\bar{a} = a$. Then we can show that $(\bar{q}, \bar{z}, \bar{a})$ is feasible for MILP (8) by construction. Since $h^l_j \leq q^l_j$ in constraint seven and as explained in the optimal solution q^l_j will equal h^l_j it follows that $\gamma_l \leq \sum_{i \in X} R^l_{ij} x_i$ for the j such that $h^l_j = 1$. This implies that $\gamma_l \leq \sum_{i \in X} \sum_{j \in Q} R^l_{ij} \bar{z}^l_{ij}$ and that the objective function value of $(\bar{q}, \bar{z}, \bar{a})$ in MILP (8) greater than or equal to the objective value of (x, q, h, a, γ) in MILP (43).

The key implication of the above observation is that when $\varepsilon = 0$, COBRA(α,ε) loses its robustness feature, so that once again when the follower faces a tie, it selects a strategy favoring the leader, as in DOBSS.

Based on this observation, the remaining observations presented in this paper about COBRA(α,ε) can be generalized to DOBSS accordingly.

Observation 2. *When α is held constant, the optimal reward COBRA(α,ε) can obtain is decreasing in ε.*

Proof 2. *Since the fifth constraint in MILP (43) makes $q^l_j = 1$ when that action has a follower reward between $(a^l - \varepsilon, a^l]$, increasing ε would increase the number*

of follower strategies set to 1. Having more active follower actions in the sixth constraint can only decrease the minimum value γ_l.

Observation 3. *Regardless of α, if $\frac{1}{2}\varepsilon > K \geq |C_{ij}^l|$ for all i, j, l, where K is the greatest absolute opponent payoff, then COBRA(α,ε) is equivalent to MAXIMIN.*

Proof 3. *Note that $|a^l|$ in MILP (43) $\leq K$. The proof simply needs to show that the leftmost inequality of the fifth constraint in MILP (43) implies that all q_j^l must equal 1. This would make COBRA(α,ε) equivalent to MAXIMIN. Suppose some $q_j^l = 0$, then that inequality states that $-K \leq \sum_{i \in X} C_{ij}^l x_i \leq a^l - \varepsilon < K - 2K = -K$ a contradiction.*

Although DOBSS and COBRA(0,ε) make different assumptions about the follower's responses, it can be shown that in a zero-sum game their optimal solutions become equivalent. This of course is not true of general sum games. In a zero-sum game the rewards of the leader and follower are related by

$$R_{ij}^l = -C_{ij}^l \quad \forall i \in X, j \in Q, l \in L \; .$$

It is well known that minimax strategies constitute the only natural solution concept for two player zero-sum games [33]. As DOBSS is an optimal Stackelberg solver it follows that it is equivalent to a minimax strategy for zero-sum games.

Observation 4. *COBRA(0,ε) is equivalent to a minimax strategy for zero-sum games.*

Proof 4. *Given the minimax solution to a zero-sum game we define the expected reward for any strategy of the follower, $\sigma_j \in \Sigma_\Psi$, to be V_j^Θ for the leader and V_j^Ψ for the follower. Assuming that the follower's strategies are ordered in descending order in terms of expected reward for the follower and there are m such strategies, the minimax solution yields $V_1^\Psi \geq V_2^\Psi \geq \ldots \geq V_m^\Psi$ for the follower and $V_1^\Theta \leq V_2^\Theta \leq \ldots \leq V_m^\Theta$ for the leader. COBRA(0,ε) is an algorithm that assumes the attacker will choose a strategy within ε of her maximum expected reward (in this case V_1^Ψ) and of these ε-optimal strategies it attempts to maximize the expected reward for the worst case outcome. As seen by $V_1^\Theta \leq \cdots \leq V_m^\Theta$ the minimum expected reward possible for the leader is the follower's optimal strategy and any ε-deviation will only result in a higher expected reward for the leader. It follows that the minimax strategy already maximizes the expected reward for the worst case outcome of any ε-optimal response by the follower. Thus, COBRA(0,ε) yields a strategy that is equivalent to minimax in zero-sum games.*

5. Experiments

We now present results comparing the quality and runtime of COBRA(α,ε) with previous approaches and baseline approaches. The goal of our new algorithm was to increase the expected reward obtained by a security agent against human adversaries by addressing the bounded rationality that humans may exhibit and the limited observations they may experience in many settings. To that end, experiments were set up where human subjects would play as followers (adversaries) against each strategy with varying observability conditions. As noted earlier we cannot prove optimality against human adversaries who may deviate from the expected optimal responses and thus we rely on empirical validation through experimentation.

5.1. Experimental Design

Our experiments were setup to examine three crucial variables of real-world domains. The first variable is the reward structure of a particular domain. Depending on the reward structure human choices could vary vastly. For instance, in some reward structures there may be only a single action that obtains the highest reward possible, while in a different reward structure there may be multiple different actions that obtain the highest reward possible. We are particularly interested in reward structures similar to those used in security domains such as that at LAX and the Federal Air Marshals Service (FAMS) [43, 26]. Our second variable is different observability conditions for human subjects. Security officials, such as those at LAX and FAMS, are interested in the observational capabilities of their adversaries and how this will affect their decisions. In the real-world, some adversaries may be able to take many observations before deciding to act while others may end up having to act with very little information. It is important to understand how this can affect the decisions and actions humans may take. Finally, we want to examine the algorithm we have presented and test its performance against human subjects given adjustments to its α and ε parameters as well as adjustments to reward structure and observation conditions to understand how we can better deal with human adversaries.

In order to examine these three variables (reward structure, observation condition, algorithm parameters) we constructed a game inspired by the security domain at LAX [39, 43], but converted it into a pirate-and-treasure theme. The domain had three pirates — jointly acting as the leader — guarding 8 doors, and each individual subject acted as an adversary. We selected 8 doors to model the 8 terminals found at LAX. Although our algorithm allows for multiple follower types

(a Bayesian Stackelberg game) we modeled each subject as a single follower type defined by the reward structure. The subject's goal was to steal a treasure from behind a door without getting caught. Each of the 8 doors would have a different reward and penalty associated with it for both the subjects as well as the pirates. For instance, as shown in Figure 1, door 4 has a reward of 3 and a penalty of -3 for the subject and a reward of 1 and a penalty of -5 for the pirate. If a subject chose a door that a pirate was guarding, the subject would incur the subject penalty for that door and the pirate would receive the pirate reward for that door, else vice-versa. Going back to our previous example if the subject chose door 4 and a pirate was guarding that door then the subject would receive -3 and the pirate would receive 1. This setup led to a Stackelberg game with $\binom{8}{3} = 56$ leader actions, and 8 follower actions.

5.1.1. Reward Structure

We constructed four different reward structures for the 8-door 3-pirate domain described that are similar to hose used in security domains such as that at LAX [43]. These reward structures can be found in the Appendix, Section A. There are three key features in these reward structures. First, we use a reward scale similar to that used at LAX to determine the payoffs for both the leader and the follower. Namely, rewards range from 1 to 10 and penalties range from -10 and -1. Second, these reward structures meet the model criteria of what are known as security games [62, 26] and they are used in domains such as LAX and FAMS. The main features of these games are a set of targets $T = \{t_1, \ldots, t_n\}$, a number (N) of resources to protect these targets, and a reward and penalty to the attacker and defender based on whether a target is covered by a resource or not. If a target is covered the attacker receives her penalty and the defender its reward else vice versa. Finally, in addition to the reward scale and to ensuring that we followed the requirements of a security game, we also wanted to examine reward structures where the follower's small ε-deviations from the strong Stackelberg equilibrium (SSE) were significantly harmful to the leader (such arbitrarily small ε-deviations model cases where the follower does not break ties in favor of the leader, see Section 2.3). These reward structures are particularly interesting for our new robust method since our method specifically guards against such potentially harmful deviations. Reward structure four is our baseline case, a zero-sum reward structure, where deviations from the follower's optimal strategy based on a SSE assumption are only better for the leader. In the other three reward structures the average expected reward for an ε-deviation — for arbitrarily small ε — from a SSE is held between -1.30 and -1.96; however, the worst case expected reward for a deviation

becomes progressively worse for each reward structure. In reward structure three the worst case deviation is -3.16, in reward structure two it is -4.21, and finally in reward structure one it is -4.56. Thus, for each of these three reward structures, follower's deviations from optimal play based on a SSE assumption can lead to potentially large degradations in the leader's expected reward.

Figure 1 shows the interface used to convey the reward structures to the subjects. The worst case outcome for deviation from SSE is progressively better from reward structure one to reward structure three. The key question is whether there is any systematicity in human subject's deviations from SSE under different conditions and if COBRA(α,ε) can capture them sufficiently to mitigate their impact. If so, then we expect our robust model to provide the largest benefit in reward structure one and the least benefit in reward structure three. In reward structure four we have shown that, given no observational uncertainty, our robust model is equivalent to the optimal minimax strategy and thus there will be no benefit due to our robust approach. However, in all four reward structures we expect our method for handling observational uncertainty to provide benefits when observational capabilities are low.

5.1.2. Observability Conditions

There are four separate observability conditions that we examined. First we explain what an observation is exactly. We imagine that time can be discretized into rounds and on each round the pirates will choose three doors to guard according to their current mixed strategy. Specifically, if we examine the UNIFORM strategy then on each round the guards will choose three doors uniformly at random to guard for that round. A single observation then consists of seeing where the guards have stationed themselves for a single round and then moving on to the next round where the guards will once again reposition themselves according to their mixed strategy. An example of an observation can be seen in Figure 2. We also printed on a separate window the observations the subjects had received in the current game for them to refer back to. For instance, after five observations the separate window may list something like [1,2,4][2,4,5][2,6,7][3,7,8][1,3,6] where each triplet is the three doors the pirates were stationed at in each of the five rounds. On each round the subject will either receive an observation or will be asked to choose a door to attack.

The four different observation conditions tested were: (i) The subject does

not get any observations (a 1 round game)[4]; (ii) the subject gets 5 observations (a 6 round game); (iii) the subject gets 20 observations (a 21 round game); (iv) the subject gets unlimited observations — simulated by revealing the exact mixed strategy of the pirate to the subject. This final condition is also a 1 round game, but we display the mixed strategy the pirates used to select their 3 doors and the subjects are then allowed to choose which door they will attack on this first round. The mixed strategy is revealed as the marginal distribution of guards over the 8 doors. Specifically, for each door the subject is given the probability that any guard will appear on that door for any given round. As shown in Figures 1 and 2, subjects were given full knowledge of their rewards and penalties and those of the pirates in all situations. In each game the subject observed the pirates' strategy under the current observability condition, reward structure, and strategy and then was allowed to make her decision on the final round of the game. After making her decision the subject was informed whether she was successful or whether she was caught.

5.1.3. Algorithms and Parameters

These experiments compare DOBSS, COBRA(0,ε), COBRA(α,ε), MAXIMIN, and UNIFORM. We chose to include COBRA(0,ε) in these experiments to demonstrate the value of having both observational uncertainty and bounded rationality assumptions in an algorithm over having just one. It is important to include either COBRA(α,0) or COBRA(0,ε) to examine whether observational uncertainty, bounded rationality, or the combination of both results has the largest affect on our results. Based on the results in all four conditions for COBRA(0,ε) we already find the need to address both boundedly rationality and observational uncertainty important. Furthermore, we later conduct detailed experiments exploring the impact of α.

The value of ε in an application should reflect the assumptions regarding the uncertainty in adversary reward and the precision with which the human adversary decides between the different actions. While determining the human adversary's precision is difficult in general, in our experiments the value of ε was determined

[4]Although this conditions more closely resembles a simultaneous move game, where a Nash equilibrium strategy would appear to be the appropriate one rather than the Stackelberg equilibrium strategy, we were interested in exploring how our algorithms could perform under these circumstances. In the real world security forces may be more concerned with intelligent adversaries who are taking observations, but there may still be instances of adversaries who attack without the ability to make observations.

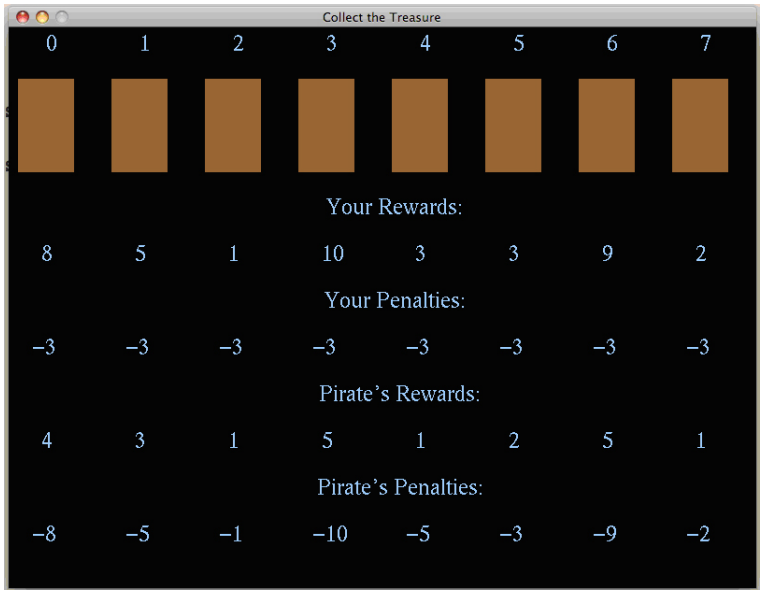


Figure 1: Game Interface

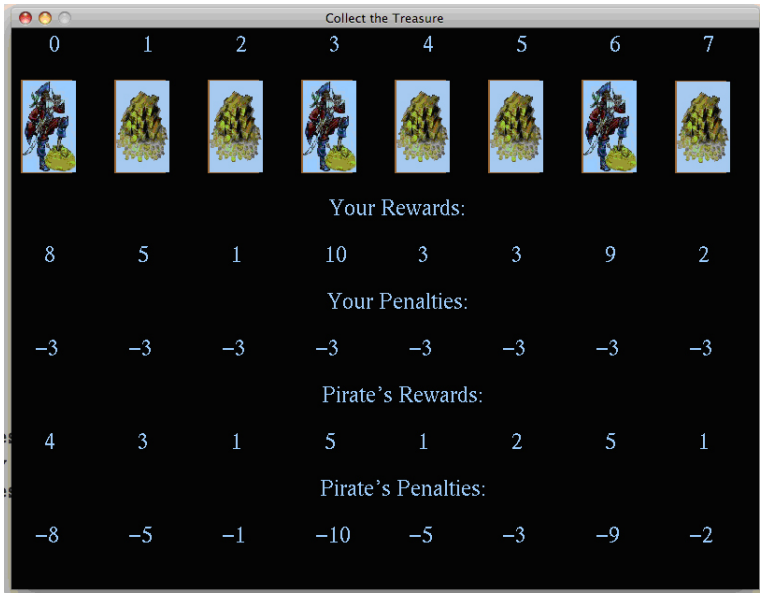


Figure 2: Single Observation

based on three factors: (i) $\varepsilon > 0$ based on the assumption that subjects playing our games were not precisely computing expected utilities; (ii) ε had to be set to a value to produce qualitatively different leader strategies than created by DOBSS and MAXIMIN to help gain a clear understanding of how ε affects the results against human subjects; (iii) ε had to be held constant across our four games. Hence we chose $\varepsilon = 2.5$ that lead to 3 to 4 actions in the subject's ε-optimal set. This is about the halfway point given our experiments had 8 total choices of actions. DOBSS assumes a single action choice to the adversary and MAXIMIN makes a worst-case assumption. In some settings a higher or lower selection of ε may be appropriate. Finding a more precise method for selecting ε is left for future studies.

Unlike deciding ε, a single choice for α will not hold across all reward structures. First, α could be expected to vary with observability conditions. Second, even for a fixed observability condition, identical values of α across reward structures is not appropriate. We include in the Appendix under Section D tables for how COBRA(α,2.5) changes as α changes from 0 to 1. Notice that in some reward structures changing α may not necessarily change the mixed strategy for the leader as in Table 21. For this reason it is necessary to examine each reward structure individually. Within each reward structure, we tried two different techniques to choose α, one with a fixed α and one with a variable α, in order to compare the impact of variable α more clearly.

Since the value of α is used to balance the amount of observed information and a-priori bias information that the adversary incorporates in her assumption of the leader strategy, this parameter should be related to the amount of observations made by the follower. Thus, the technique with variable α is obviously the more standard version. Clearly when the follower has unlimited information $\alpha = 0$ (the follower correctly estimates the leader strategy) and when she has no observations $\alpha = 1$ (the follower uses the a-priori bias). Less straight forward is how to set an α value when the follower has 5 or 20 observations. We follow two methods of adjusting α:

- The first is to conduct r trial experiments with human adversaries using COBRA(0,ε) with 0, unlimited, and n observations. In our experiments n is either 5 or 20. Given the choices made by each subject in the trial experiments we collect r subject expected rewards for each observation condition. Let $corr_{n,0}$ be the correlation between the r subject expected rewards for the n observation condition and the 0 observation condition. Similarly let $corr_{n,u}$ be the same with the unlimited observation condition. We set α

for n observations as $\alpha = corr_{n,0}/(corr_{n,0} + corr_{n,u})$. If the results were more correlated with the unobserved condition this would make α higher and otherwise it would make α lower. We assume that having more observations would lead to results that were more correlated with the unlimited observation condition and having less observations would lead to results that were more correlated with the unobserved condition. We use this form of adjusting α in reward structures 1 and 2.

- The second method to set α uses fixed arbitrary values: $\alpha = .75$ for the 5 observation condition and $\alpha = .25$ for the 20 observation condition. This simple method was created to evaluate the necessity of conducting trial runs to determine the α values experimentally. We use this second method in reward structures 3 and 4.

Finding an exact method to select α in these conditions remains an issue for future work. Before running this study we did not have any evidence to suggest either method we attempted would work best. However, we hoped to gain some insight into choosing α based on the results obtained from both approaches. Section 5.2.6 provides more analysis for choosing α.

The second technique to selecting α is to assume a constant α, leading to a version of COBRA(α,ε) that we will refer to as COBRA(C,ε) (COBRA(α,ε) with constant α). For COBRA(C,ε), α was set to the same α value as the 5 observation condition from the four reward structures with the expectation that it would perform poorly in higher observation conditions since it was not appropriately adjusted. The α values chosen for COBRA(α,ε) in each of the reward structures are summarized in Table 2. In Section 5.2.6 we will present experimental results for many additional settings of the α parameter in both the unobserved and unlimited observation conditions. Based on these settings we will introduce a third, better performing method for determining α that generalizes to all four reward structures and is more practical in the field.

	Unobserved	5	20	Unlimited
Structure One	1	.37	.03	0
Structure Two	1	.54	.41	0
Structure Three	1	.75	.25	0
Structure Four	1	.75	.25	0

Table 2: α values used in each observation condition for a given reward structure.

5.2. Quality Comparison

5.2.1. Setup

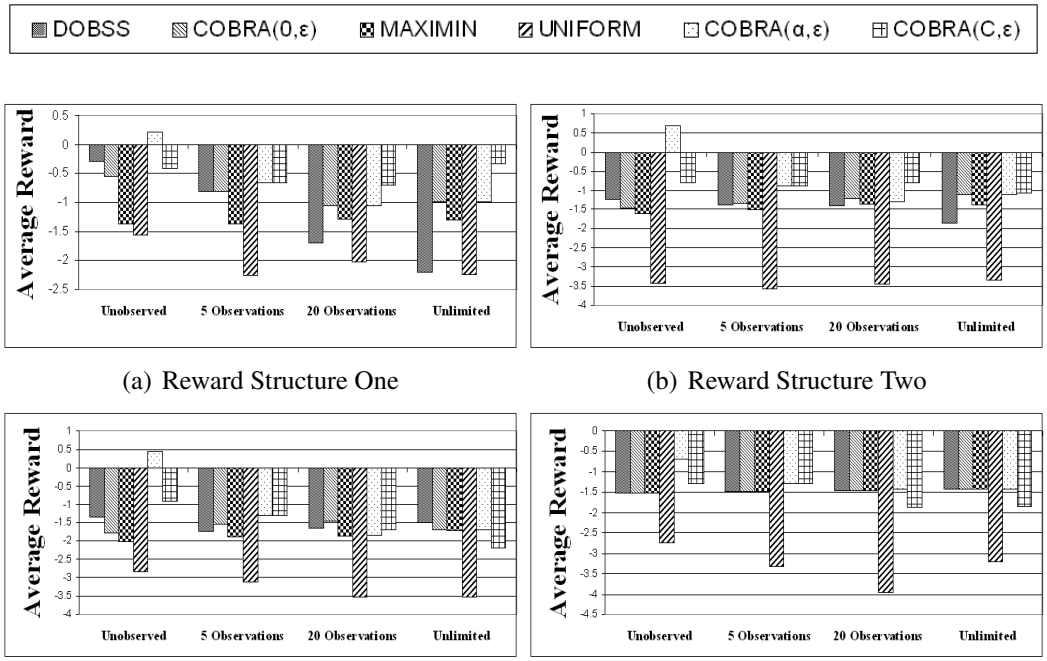
Each of our 96 game settings (four reward structures, six algorithms, and four observability conditions) were played by 40 subjects, i.e. in total there were 2480 total trials. There are fewer than 3840 trials however because some are duplicates. For example, the unobserved condition for a particular reward structure need only be played by 40 subjects as opposed to 240 subjects (one set of 40 for each algorithm). The choices made by those 40 subjects could then be used to compare all six algorithms since the subjects were not given any information about a particular algorithm in advance for this condition. Thus the choices made were irrelevant of the algorithm used. Also notice that in some observation conditions a few of the algorithms yielded the same strategy, i.e. COBRA(α,ε) and COBRA(C,ε) in the five observation condition used the same ε and α parameters. Subjects consisted of Engineering undergraduate and graduate students at the University of Southern California.

Given this setup, each subject played a total of 14 unique games and the games were presented in random orderings to avoid any order bias. In total there were 178 different subjects that played in these experiments. For a given algorithm we computed the expected leader reward for each follower action, i.e. for each choice of door by subject. We then found the average expected reward for a given algorithm using the actual door selections from the 40 subject trials. For each game, the objective of a subject was to earn as many points as possible. The subject was allowed to choose a single door, based on the current reward structure and observation condition, that they believed was unguarded and once a door was chosen that game was over and the subject played the next game. Starting with a base of 8 dollars, each reward point within the game was worth 15 cents for the subject and each penalty point deducted 15 cents. This was incorporated to give the subjects incentive to play as optimally as possible. On average, subjects earned $13.09.

5.2.2. Results

Figure 3(a) shows the average expected leader reward for our first reward structure, with each data-point averaged over 40 human responses. Figures 3(b), 3(c), and 3(d) shows the same for the second, third, and fourth reward structures. Notice that all strategies have a negative average with the exception of

COBRA(α,ε) in the unobserved condition[5]. In both figures, the x-axis shows the observation condition for each strategy and y-axis shows the average expected reward each strategy obtained. For example, examining Figure 3(a) in the unlimited observation condition, COBRA(C,ε) scores an average expected leader reward of -0.33, whereas DOBSS suffers a 663% degradation of reward, obtaining an average score of -2.19.



(a) Reward Structure One

(b) Reward Structure Two

(c) Reward Structure Three

(d) Reward Structure Four

Figure 3: Expected Average Reward

[5]The reason all strategies obtain a negative average is due to the lack of enough resources (or guards) in this setting, but that is where randomization strategies have the most impact. The reason COBRA(α,ε) was able to obtain a positive reward in the unobserved condition is that it placed all its resources on a small subset of doors (a more deterministic strategy) assuming that humans would choose these doors based on their belief of the ignorance prior (uniform distribution). In practice the humans did indeed play according to this expectation and thus the rewards obtained were much higher. Examining Tables 14-16 it is clear that while most rewards are negative, in the unobserved condition for COBRA(α,ε) (seen as COBRA(1,2.5)) there are a few distinct doors that obtain very high rewards, which correspond to the doors chosen by most human subjects in these experiments for this condition.

5.2.3. Key Observations

We provide our key observations first, then provide statistical significance tests in the following section and later provide a deeper analysis. The *main* observation from Figure 3 is that the COBRA(α,ε) algorithm has a performance that is superior to the theoretically optimal DOBSS algorithm and baseline approaches against human subjects. We can breakdown this main observation into observations of the following trends:

1. *Considering observational uncertainty is important*: When creating algorithms for leader strategies in Stackelberg games it is important to address biases that may arise from observational uncertainty. Focusing first on the unobserved and 5 observation conditions, COBRA(α,ε) and COBRA(C,ε) obtain much better results demonstrating the benefit of incorporating an anchoring bias. In the 20 and unlimited observation conditions we see that COBRA(α,ε) and COBRA(C,ε) can still provide benefits, however, in these cases it is assumed that observational uncertainty is low so the effect of anchoring biases begin to diminish. As can be seen in Figure 3, COBRA(α,ε) performs better than DOBSS in all except only two cases (Reward structure three 20 and unlimited observations). Because of its lack of adjustment of α, COBRA(C,ε) is seen to perform better than DOBSS in all conditions except in four total cases confined to reward structures three and four.
2. *Addressing bounded rationality is an important component when designing algorithms that perform against humans*: Examining COBRA(0,ε) specifically we can see in the 20 and unlimited observation conditions for reward structure one and two that addressing bounded rationality provides improvements over DOBSS. In the lower observation conditions we begin to see how observational uncertainty becomes a larger factor making bounded rationality a secondary issue. This follows from our first observation presented. In reward structure four, a zero-sum game, COBRA(0,ε) and DOBSS are exactly equivalent, so no gains are expected. In reward structure three, where deviations from SSE were least harmful of the remaining reward structures, there is some difference between COBRA(0,ε) and DOBSS, but not as much as in reward structures one and two. In general though, our results confirm the expected result that humans do not strictly play the game theoretic optimal. Thus, it is important to address deviations from this theoretic optimal based on bounded rationality to prevent what could potentially be significant losses.
3. *COBRA(C,ε) surprisingly outperforms COBRA(α,ε) under high observa-*

tion conditions in some reward structures: This unexpected result lead to the subsequent experiments reported in Section 5.2.6 and to a more efficient heuristic for selecting α.

5.2.4. Statistical Significance

Our main observation in the previous section critically depends on significant differences between COBRA(α,ε) and the remaining strategies. We chose to employ more robust statistical methods for our tests in order to overcome limitations with our data set. These limitations include a non-normal distribution (due to a very small number of discrete choices as opposes to continous or near continuous choices) and high variance. Having a normal distribution is an important assumption of traditional statistical tests such as the classic T-test.

For our statistical significance tests we used a one-way Brunner-Puri test [9] for repeated observations in the unobserved condition and we used Yuen's test for comparing trimmed means [63] in the 5, 20, and unlimited observation conditions. In the unobserved condition all structures were treated separately, however, in the 5, 20, and unlimited observation conditions reward structures one and two were combined into a single data set. For an in depth discussion of this decision and also why these statistical tests were chosen please see the Appendix, Section B. In general, given the nature of our data — discrete rather than continuous distribution of values and non-normal distributions — it can be difficult to obtain statistical significance without significantly larger data sets. Yet, our results do achieve statistical significance in key cases *demonstrating the effectiveness of our COBRA(α,ε) strategies*, as summarized below.

Conclusions regarding unobserved condition: Looking first at the unobserved condition, COBRA(α,ε) obtained statistical significance against DOBSS in reward structures one, two, and three with a maximum p-value of .04. Since reward structure four is a zero-sum game we reiterate that the strategy space for our robust method is limited. Namely, in all observation conditions the results for the DOBSS, MAXIMIN, and COBRA(0,ε) algorithms are identical. Thus, the only way to alter the strategy based on our robust method is through the use of α. Although the results are in favor of COBRA(α,ε) in this reward structure, in the unobserved condition we were unable to achieve statistical significance against DOBSS/COBRA(0,ε)/MAXIMIN without a larger data set. Against MAXIMIN and UNIFORM, COBRA(α,ε) obtains statistical significance in reward structures one, two, and three with a maximum p-value of .04 except in reward structure one where it obtains a p-value of .098 against MAXIMIN. Overall these results demonstrate the superiority of COBRA(α,ε) over DOBSS and simple baseline

algorithms in the unobserved condition.

Conclusions regarding combined data from reward structures one and two in remaining observation conditions: In the 5, 20, and unlimited observation cases the maximum p-value obtained for COBRA(C,ε) versus any other strategy was .033. Given that COBRA(C,ε) is shown outperforming every other strategy, including COBRA(α,ε), under these observation conditions, this establishes that COBRA(C,ε) is statistically significantly better than all other strategies in these reward structures under these observation conditions. This in turn demonstrates the superiority of the COBRA(α,ε) algorithm as a whole in these reward structures and observation conditions since COBRA(C,ε) is an instantiation of COBRA(α,ε) with a particular choice of α.

Conclusions regarding reward structure three in remaining observation conditions: In reward structure three for the 5 and 20 observation conditions we do not achieve statistical significance between DOBSS, COBRA(α,ε), and COBRA(C,ε) making the results obtained inconclusive. To a certain extent this is an implication of deviations in reward structure three not being as harmful to the leader (see Section 5.1.1) and thus more difficult to obtain significant differences between the strategies. However, in the unlimited observation condition we find that DOBSS is statistically significantly better than all other strategies with a maximum p-value of .036. Although in the unlimited condition, given the choices made for α and ε, DOBSS outperforms our robust strategy, we will later present an alternative choice for these parameters in Section 5.2.6 that is able to outperform DOBSS.

Conclusions regarding reward structure four in remaining observation conditions: In reward structure four, as presented in Section 4, we know that MAXIMIN, DOBSS, and COBRA(0,ε) are equivalent. In the unlimited and 20 observation conditions, given our choice for α, it is also the case that COBRA(α,ε) and DOBSS/MAXIMIN/COBRA(0,ε) are equivalent. Given these equivalencies, our only concern is COBRA(C,ε). Since COBRA(C,ε) is outperformed in the 20 and unlimited observation conditions it provides no benefits in these cases. However, for the 5 observation condition we find that COBRA(C,ε) (or COBRA(α,ε)) is statistically significantly better than all other strategies with a maximum p-value of .005. Given these results and the statistical significance achieved in the 5 observation condition we find COBRA(α,ε) to be the superior strategy, even in a zero-sum game, due to its ability to handle observational uncertainty.

5.2.5. Analysis of Results

We discuss the key implications of the observations presented in Section 5.2.3 and why they were reached. We include two tables for reference in the following discussion, Tables 4 and 3. Table 3 shows the expected rewards (for a subset of the algorithms tested) the leader should obtain for each door selection by the follower in reward structure one on average. For instance, if the follower selected Door 2 when playing against DOBSS the leader would expect to obtain an average reward of -.97. Obviously depending on whether there was a guard stationed there or not in any particular instance the leader would get the respective reward or penalty associated with that door, but over time the average would converge to the expectation of -.97. We have placed in bold font the expected reward for each of the algorithms. This is the reward an algorithm expects to receive based on the assumptions it has made. For example, DOBSS is an algorithm that expects the follower to play a strong Stackelberg Equilibrium (SSE) strategy. This means that the follower will choose, between her highest reward choices, the door that is also best for the leader. This SSE strategy is the reward value that has been placed in bold font for DOBSS. MAXIMIN on the other hand is an algorithm that makes a worst-case assumption and thus all doors that give the minimum expected reward are placed in bold font.

Table 4 shows the percentage of times the follower chose a response that gives the leader (pirate) a reward equivalent to or higher than the expected reward for the current algorithm under different observation conditions in reward structure one. We will refer to these responses as *expected strategy(s)*. To clarify what we mean by *expected strategy(s)* we will look at COBRA(0,ε) as an example. As seen in Table 3, this algorithm expects to receive a reward of -.36. Thus, in Table 4 under the unobserved observation condition we see that the follower chose a door that gave the leader a reward of -.36 or higher (an *expected strategy*) 65% of the time. We point out that MAXIMIN is a strategy that expects a worst-case outcome and thus all doors are *expected strategies*. DOBSS is on the opposite extreme, since it assumes perfectly rational play, and thus generally results in a single door being the *expected strategy*. COBRA(0,ε) and COBRA(α,ε) fall somewhere in between these lines, where multiple doors within the ε-optimal strategies are *expected strategies*, but less doors than MAXIMIN where all doors are *expected strategies*. Of course as shown in Section 4 this depends on the setting of ε since a setting that is too high will yield the same result as MAXIMIN.

Table 3 shows an important trade-off. MAXIMIN achieved a 100% match with *expected strategies* (Table 4), but it does so by making all leader rewards

low (-1.63). DOBSS achieves low match with *expected strategies*, but its leader expected payoff is higher (.39). COBRA(α,ε) is in the middle of these extremes. We include in the Appendix under Section C: i) tables presenting the actual mixed strategies for each of the reward structures; ii) tables of the expected rewards for each reward structure given the strategies presented in i); iii) tables for the percentage of times the follower chose an *expected strategy* in each reward structure. In each of the expected reward tables we have placed in bold font the expected reward for each of the algorithms. We reiterate that an *expected strategy* is any door selection that gives an expected reward for the leader at least as high as the values in bold font.

	DOBSS	COBRA(0,ε)	MAXIMIN	COBRA(.37,ε) COBRA-5	COBRA(.03,2.5)
Door 1	-5	-4.58	**-1.63**	-5	-4.61
Door 2	-.97	-.42	**-1.63**	**-.30**	**-.37**
Door 3	.36	**-.36**	-1	**-.30**	**-.37**
Door 4	-1.38	-.79	**-1.63**	**-.30**	-.73
Door 5	.06	**-.36**	**-1.63**	**-.30**	**-.37**
Door 6	-1	-.86	-1	-1	-.87
Door 7	**.39**	**-.36**	**-1.63**	**-.30**	**-.37**
Door 8	-4.57	-3.69	**-1.63**	-3.32	-3.67

Table 3: Leader expected rewards for each door selection in reward structure one.

Structure One	Unobserved	5	20	Unlimited
DOBSS	20%	7.5%	17.5%	12.5%
COBRA(0,ε)	65%	65%	65%	70%
COBRA(α,ε)	57.5%	92.5%	72.5%	70%
COBRA(C,ε)	92.5%	92.5%	87.5%	95%
MAXIMIN	100%	100%	100%	100%

Table 4: Percentage of times follower chose an *expected strategy* in reward structure one.

Our first conclusion was that observational uncertainty is important. By accounting for this uncertainty, strategies are able to exploit human perceptions and make more appropriate use of resources. That is why COBRA(α,ε) performs better than DOBSS in most conditions. In fact, examining Table 4 we see that in the unobserved condition against COBRA(α,ε), human subjects played an *expected*

strategy 57.5% of the time, such as door 3 or door 5 as seen in Table 3, while against DOBSS they played an *expected strategy* merely 20% of the time. However, based on MAXIMIN's performance it is clear that getting followers to play *expected strategies* is not the only component. As mentioned earlier, there is a tradeoff in this match with *expected strategies* and leader rewards. MAXIMIN is too loose with its resources making all responses *expected strategies* and thus the benefits begin to diminish because resources are spread too thin. It is important to utilize resources efficiently and not squander them unnecessarily. By more accurately modeling human responses our new strategies are better able to utilize resources to guard against these responses and thus achieve a higher reward, one that is closer to the reward they expect to recieve based on their *expected strategies*. Of course when observation is high we assume that observational uncertainty is low and adjust our strategies accordingly. Thus we find that utilizing a strategy that exploits human anchoring bias, but does not squander resources, provides the benefits we are seeking.

Our second conclusion was that addressing bounded rationality is important when dealing with human adversaries. We reach this conclusion due to COBRA(0,ε)'s superior performance. In fact, examining Table 4 we clearly see that under all observation conditions the assumptions made by DOBSS are a poor model of human choices. Against COBRA(0,ε) on the other hand, which addresses bounded rationality by utilizing the concept of ε-optimal responses, human subjects consistently play *expected strategies* 65-70% of the time under all observation conditions. While this improvement in *expected strategy* match comes at the cost of lower expected reward (Table 3) the overall results indicate that the trade-off leads to an overall better performance of COBRA(0,ε). This is a clear indication of the benefits that can be obtained by addressing bounded rationality. In fact, we can specifically see in the 20 and unlimited observation conditions for reward structure one and two that addressing bounded rationality provides improvements over DOBSS. Indeed, it is necessary to address bounded rationality when dealing with humans [51]. Many times their choices can be guided by their cognitive limitations and thus it is necessary to robustly guard against a spectrum of possible choices rather than optimize against a single optimal choice [51, 46, 52, 10]. By optimizing against the perfectly rational choice DOBSS may make poor use of its resources when dealing with human adversaries. Even COBRA(0,ε) is not a perfect model of human behavior, however, it is at least a step in the right direction since it is able to obtain *expected strategies* more often than algorithms without bounded rationality.

We defer the explanation of our final key observation to Section 5.2.6. How-

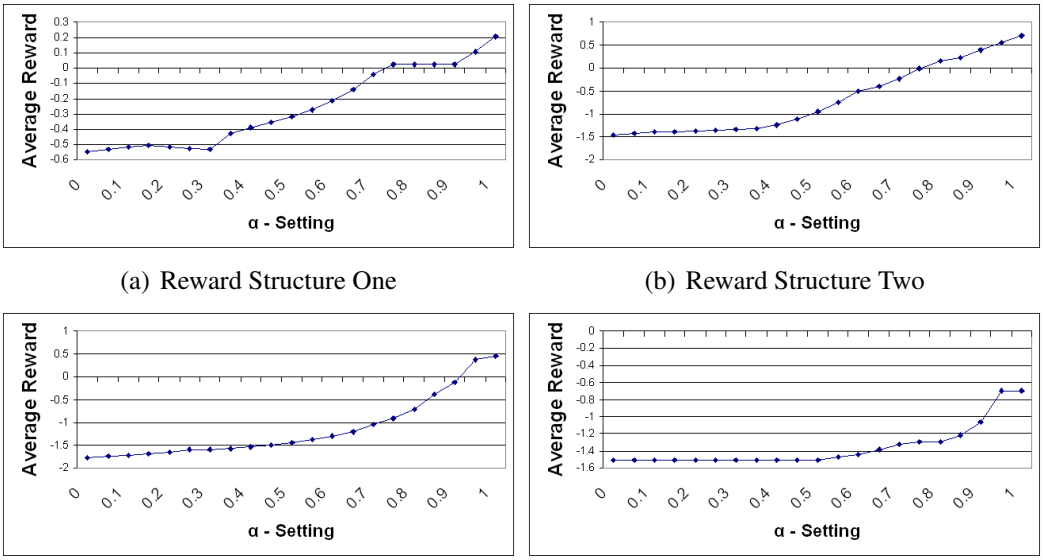
ever, we point out that due to the poor performance of COBRA(α,ε) in reward structure three compared to DOBSS we ran further experiments exploring different α values that are separate from the experiments presented in Section 5.2.6. Using a strategy with $\alpha = .5$ we found through experimentation with 40 new subjects that in the unlimited observation condition, COBRA(α,ε) obtains an average expected reward of -1.35 outperforming DOBSS with an average expected reward of -1.5. In Section 5.2.6 we present 3 alternate choices for α in the unlimited observation condition on top of the three choices presented in these original experiments.

5.2.6. Handling Observational Uncertainty

Given the significant impact of α on our results, this section provides further analysis of the choice of α on performance. Given that human choice under uncertainty remains a key area of research in psychology [15, 16, 24, 27, 54, 57], it is difficult to provide a definitive answer; however, we provide a solid initial grounding and heuristics for choosing α. We focus on the two extreme observability cases — the unobserved observation case and the unlimited observation case — for this initial investigation.

As explained before, since the choices made in the unobserved condition were made irrespective of the strategy used (humans did not have any information on the strategy being employed when they made their decision) we were able to test all α values in this case using the data collected for each reward structure. The results are presented in Figure 4 and demonstrate a clear increasing trend in all four reward structures. On the x-axis we vary the value of α and on the y-axis we show the average expected reward obtained for a particular value of α given the choices made by the subjects in the unobserved condition. For example, looking at Figure 4(b) we see when $\alpha = 0$ the average expected reward is -1.46 while when $\alpha = 1$ the average expected reward is .7. These results suggest that in the absence of observations, humans do appear to be anchoring on the uniform distribution and thus $\alpha = 1$ is the optimal setting in all four reward structures. It follows that if the expectation is for humans not to take any observations of the leader strategy, exploiting their anchoring biases can be important.

Determining an appropriate α value for the unlimited observation condition is more difficult. Since the humans are given the strategy in advance under this condition and normally (with a few exceptions) changing α also alters the strategy used, we were required to test each new value of α with a new set of subjects. To avoid exhaustively testing a large number of α values, we focused on testing three new α values per reward structure in particular. For these experiments we



(a) Reward Structure One

(b) Reward Structure Two

(c) Reward Structure Three

(d) Reward Structure Four

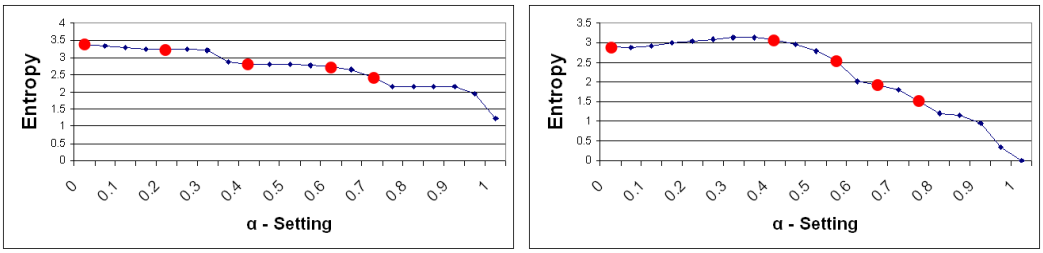
Figure 4: Unobserved Condition - Expected Average Reward

used the exact same setup described previously but the subjects only played 12 games instead of 14 (for the 12 new α values) and each of these games were with unlimited observation.

The new values of α tested against human subjects were selected using two key criteria. Before discussing these criteria, we present Figure 5, which helps ground these criteria. We also define a term which we will call *strategy entropy* as $-\sum_{i=1}^{n}(p_i * log(p_i))$ where p_i is the probability value shown on door i. Although this is the standard equation for entropy it differs in our setting as it is defined over the marginal distribution subjects are shown. This is in contrast to calculating the entropy of the mixed strategy that produced this marginal distribution. As explained previously in Section 5.1.2, in the unlimited observation condition, the strategies are presented to subjects as the marginal probability distribution of guards over the 8 doors, where the sum of the probabilities over all 8 doors will be 3. Specifically, for each door we give the subject a probability p and this is the probability that they will obtain their penalty (there will be a guard on the door) where $1 - p$ is the probability they will obtain their reward (there will not be a guard on the door).

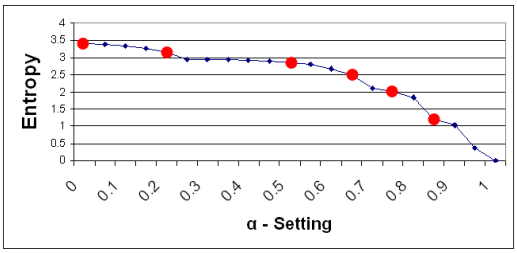
Based on this definition, a higher *strategy entropy* represents a strategy where the probability value of each door is closer to $.375$ (300% divided evenly among

8 doors) and a lower *strategy entropy* represents a strategy where the probability value on each door is closer to 1 or 0 (more specifically the highest entropy possible is 4.245). Figure 5 shows, for each value of α in each reward structure, the *strategy entropy* produced from the corresponding mixed strategy. On the x-axis we list the value of α and on the y-axis we list the *strategy entropy* obtained from the corresponding strategy. We highlight the values used in the additional experiments we ran and the original experiments for the unlimited observation condition with a bold circle. More specifically, for each of these reward structures there are 5 highlighted values corresponding to the three new values selected for these additional experiments, the value used in COBRA(C,ε) in the original experiments, and finally the value used in COBRA(α,ε) in the original experiments which was $\alpha = 0$[6].

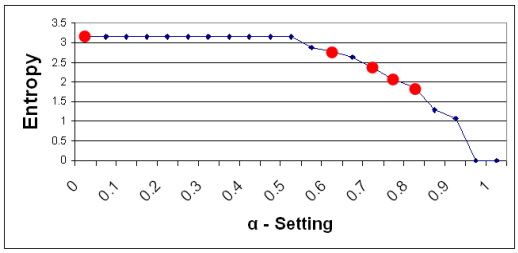


(a) Reward Structure One

(b) Reward Structure Two



(c) Reward Structure Three



(d) Reward Structure Four

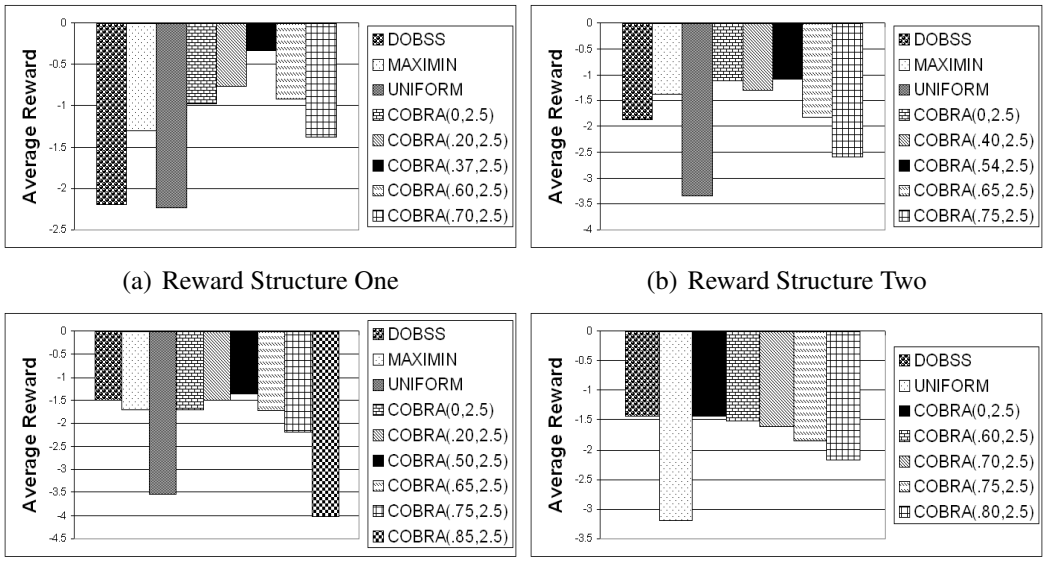
Figure 5: *Strategy Entropy* for varying α values.

Given Figure 5, our two criteria for selecting α values are as follows: i) The *strategy entropy* for the corresponding mixed strategy should not be low as this corresponds to deterministic strategies. Since humans are provided this strategy

[6]In reward structure three there is a sixth value that was used in the original experiments as an alternative value for COBRA(C,ε) in the unlimited observation condition to outperform DOBSS.

in advance, they would certainly exploit a very high α, i.e. low entropy. We selected strategies with entropy 1.19 or higher; ii) The *strategy entropy* for the corresponding mixed strategy should be quantitatively different than other α values already being tested. For example, for reward structure four, strategy entropy is constant over a range of α values as the strategy is constant. Selecting two α values from this range is not useful.

The results of these new experiments along with the results from the original experiments in the unlimited observation condition can be seen in Figure 6. On the x-axis we present the different algorithms and different values of α for COBRA(α,ε) and on the y-axis we present the average reward obtained by each of the corresponding strategies. For example in Figure 6(b), COBRA(.54,2.5) obtains an average reward of -1.08.



(a) Reward Structure One

(b) Reward Structure Two

(c) Reward Structure Three

(d) Reward Structure Four

Figure 6: Expected average rewards for varying α under the unlimited observation condition.

Figure 6 allows us to make the following observations about setting α in COBRA (α, ϵ) for the unlimited observation condition:

- *High values of α lead to poor performance*: Choosing a high α, which leads to a lower *s*trategy entropy performs poorly. Lower *s*trategy entropy implies more determinism, which human followers exploit. In our experiments, *s*trategy entropy below 2.4 appeared to degrade performance.

- *Mid-range values of α may lead to the best performance in reward structures where a follower's deviation from expectation is harmful to the leader*: Previously it was assumed that $\alpha = 0$ would be the best setting for the unlimited observation condition, but our results are clearly contradictory to this. Choosing the lowest values of α (near 0) leads to a high *s*trategy entropy, but that does not provide the best outcome (in three of the four reward structures). One possible explanation for this result is that humans have difficulty reasoning about strategies that are overly complicated. If *s*trategy entropy is too high then humans may have difficulty evaluating alternatives in order to find optimal or even ϵ-optimal strategies and this may cause them to deviate from *expected strategies* which can lead to a degradation in the leaders expected reward — as explained previously, these three reward structures were specifically designed such that deviation by the followers from *expected strategies* were harmful to the leader. In our experiments, a *s*trategy entropy between 2.5 and 2.85 appears to be optimal.

- *Lowest values of α may perform optimally in zero-sum reward structures*: In the fourth reward structure, or zero-sum reward structure, the α value that gives the maximum *s*trategy entropy (i.e. COBRA(0,2.5)) is optimal. This makes intuitive sense given that the optimal strategy in a zero-sum game is equivalent to a minimax strategy as shown by our observations in Section 4. In a zero-sum game, any deviation from optimal play by the follower leads to an expected reward that is strictly higher for the leader. For reward structures one, two, and three however, deviations can lead to severe degradations in the leaders expected reward making it more important to account for deviations from optimal play. Our robust strategies are designed to combat these potentially detrimental deviations, which means in a zero-sum game, as we have explained previously, they are the least useful. However, even in a zero-sum game it is still possible to exploit human anchoring biases in low observation conditions to obtain higher expected leader rewards as shown by our results.

Based on these results we can see why COBRA(C,ε) is seen to outperform COBRA(α,ε) in the unlimited observation condition as noted in Section 5.2.3. Also, our results have demonstrated that the optimal choice for α in the unobserved observation condition appears to be $\alpha = 1$. In the unlimited observation condition our results have shown that, for general sum games, setting α to a value that leads to a strategy between a mid-range to high-range entropy appears to be best. While a precise predictor for optimal α remains an issue, particularly for

other observation conditions, using a *strategy entropy* based technique for selecting α in these conditions appears to be a promising approach.

Given the analysis presented in Section 5.2.5 and these additional experiments, COBRA(α,ε) and COBRA(C,ε), with appropriately chosen α and ε values, appear to be the best performing among the presented algorithms. The performance of DOBSS in some of these experiments also illustrates the need for the novel approaches presented in this paper for dealing with humans. For example, in Figure 3(a) it is clear that under high observation conditions DOBSS performs very poorly in comparison to other strategies. In fact, in this case DOBSS is seen performing even worse than simple baseline algorithms such as MAXIMIN. Indeed, with DOBSS having been deployed for almost three years at Los Angeles International Airport (LAX) [43], these results show that security at LAX could potentially be improved by incorporating our new methods for dealing with human adversaries.

5.3. Runtime Results

For our runtime results, in addition to the original 8-door game, we constructed a 10-door game with $\binom{10}{3} = 120$ leader actions, and 10 follower actions. To average our run-times over multiple instances, we created 19 additional reward structures for each of the 8-door and 10-door games. Furthermore, since our algorithms handle Bayesian games, we created 8 variations of each of the resulting 20 games to test scale-up in number of follower types. For the *a priori* probability distribution of follower types we assume each follower type occurs with a 10% probability except the last which occurs with $1-.10(n-1)$ probability where n is the number of follower types. For example, if there are 5 follower types, the first four types each occur with a 10% probability and the last type occurs with a 60% probability. Experiments were run using CPLEX 8.1 on an Intel(R) Xeon(TM) CPU 3.20GHz processor with 2 GB RDRAM.

In Figure 7, we summarize the runtime results for our Bayesian game using DOBSS, COBRA(0,ε), COBRA(α,0), COBRA(α,ε) and MAXIMIN. We include one graph for the 8-door results and one for the 10 door results. For COBRA(α,ε) we set $\varepsilon = 0$. For both COBRA(α,0) and COBRA(α,ε) we varied the value of α to show the impact on solution speed. We include $\alpha = .25$ and $\alpha = .75$ in the graph, denoted by COBRA(.25,2.5)/COBRA(.75,2.5) for COBRA(α,ε) and COBRA(.25,0)/COBRA(.75,0) for COBRA(α,0) respectively. The *x*-axis in Figure 7 varies the number of follower types from 1 to 8. The *y*-axis of the graph shows the runtime of each algorithm in seconds. All experiments that were not concluded in 20 minutes (1200 seconds) were cut off. As expected, MAXIMIN

is the fastest among the algorithms with a maximum runtime of 0.054 seconds on average in the 10-door case. Not anticipated was the approximately equivalent runtime of DOBSS and COBRA(0,ε) and even more surprising were the significant speedups of COBRA(α,ε) and COBRA(α,0) over DOBSS and COBRA(0,ε) depending on the value of α. As shown in Figure 7 as α increases, the runtime of COBRA(α,ε) and COBRA(α,0) decreases. For example, in the 10-door 8 follower type case when $\alpha = .25$ COBRA(α,ε) is unable to reach a solution within 1200 seconds on average, however, when we increase α to $.75$ COBRA(α,ε) is able to find a solution in 327.5 seconds on average. In fact, excluding MAXIMIN, every strategy except COBRA(α,ε) with $\alpha = .75$ and COBRA(α,0) reached the maximum runtime in the 10-door 8 follower type domain. This speedup could be attributed to the branch-and-bound methods used to find solutions to these MILPs. Since COBRA(α,ε) and COBRA(α,0) distribute some of their weight to the uniform distribution it decreases the number of branch-and-bound nodes necessary to achieve a solution by decreasing the branch space. Our set of results in particular support this theory. These results demonstrate that our algorithms do not incur significant runtime costs for the proposed enhancements to deal with bounded rationality and observational uncertainty and in fact may even provide runtime improvements over DOBSS.

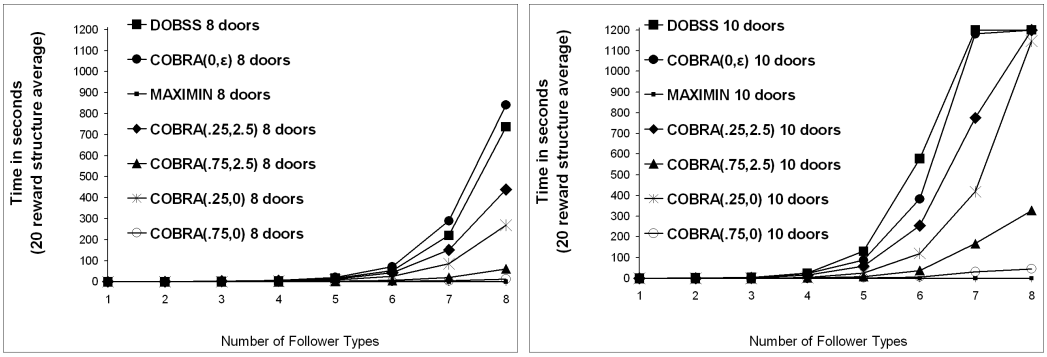


Figure 7: Comparing Runtimes

6. Related Work

We examine three areas of related work. In Section 6.1 we examine related work in game theory and game theoretic approaches. In Section 6.2 we look at non-game theoretic approaches to security and finally in Section 6.3 we address related work in human biases and observational uncertainty in Stackelberg games.

6.1. Game Theoretic Approaches

Addressing deviations from the theoretically optimal choice is a common concern in game theoretic settings [20, 32, 48, 49, 55, 10, 37, 31, 2]. One well known approach is trembling hand perfect equilibrium [48, 49]. While a formal definition of trembling hand perfect equilibrium is provided in [36], the key intuition is that players find an equilibrium given that other players may choose unintended strategies due to a "trembling hand" or error. The key difference between our work and that of the trembling hand approach is that we make specific predictions on how our adversary may deviate, as opposed to assuming she could make uncorrelated mistakes that lead to unexpected events, and then optimize against these predictions. In a security setting this seems more appropriate since many security forces will have some prior knowledge about who their adversaries may be and what their interests are.

An approach that is more closely related to our work is that of ε-equilibrium [55]. This approach has been examined for simultaneous move games with respect to Nash equilibrium, but to the best of our knowledge a similar approach has not been attempted for Stackelberg games. In a simultaneous move game it is assumed that a ε-equilibrium point has been reached when a unilateral deviation from that equilibrium point by one of the players will not increase the payoff of that player by more than ε. Although our approach also examines ε-deviations on behalf of the adversary we still consider a perfectly rational leader. This is an important difference between our approach and ε-equilibrium approaches since we assume only the follower is willing to make unilateral deviations within ε of her optimal choice while the leader continues to play optimally. Addressing these ε-deviations is helpful against human adversaries who may have limited computational abilities and thus may not be able to distinguish between rewards within a ε-bound; indeed we experimentally illustrate the usefulness of this approach.

An alternative approach to accounting for some ε-bound is quantal response equilibrium [32]. Quantal response equilibrium is an approach that assumes the players observe the reward for any particular action with some error. For any particular action there is a vector of possible observation errors and some distribution over this vector. The players will then choose their optimal actions depending on their observed error. Under these circumstances players will play better strategies more often than worse strategies, but best strategies are not always played. This approach has been applied to both normal-form games and extensive-form games. Although quantal response could potentially be better at addressing human behavior in our settings, two of the major pitfalls of this approach are the computational complexity and the necessity for prior data in order to specify the

error structure in advance. Given the data that has been collected in this paper we hope to implement and experiment with a quantal response model in the future.

Robust game theory was first introduced for Nash equilibria [1] and adapted to wardrop network equilibria in [35]. These prior works show existence of an equilibrium and how to compute it, when players act robustly to parameter uncertainty. We draw inspiration from these concepts in our techniques to guard against choice uncertainty, however, the key difference is that instead of robustly guarding against all uncertainty we make predictions about how choices will deviate from the optimal due to *observational uncertainty* and *bounded rationality*. Bounded rationality has itself of course received significant attention in game theory related literature [46] – the key question remains how to precisely model it in game theoretic settings [52].

In addition to all of the key equilibrium concepts, the field of experimental game theory has additionally provided a wealth of contributions and insights on deviations of human play from equilibrium predictions and different models to explain these deviations [10]. In contrast with previous work, our work has focused on Stackelberg security games and provided new MILPs for an automated computer program to play as the leader in such games, taking into account specific human biases and limitations when they act as followers. In this way our work complements the previous work mentioned above by emphasizing the need for efficient computational strategies and understanding their interactions with human responses. As mentioned earlier, specific key novelties of our work are in bringing together (i) previous best known algorithms from the multiagent literature for solving Bayesian Stackelberg games; (ii) robustness approaches from robust optimization literature; (iii) anchoring theories on human perception of probability distributions from psychology.

6.2. Non-Game Theoretic Approaches

Non-game theoretic models have also been explored for security, such as the Hypercube Queuing Model [29], based on queuing theory. This model depicts the detailed spatial operation of urban police departments and emergency medical services and has found application in police beat design, allocation of patrolling time, etc. The patrolling problem itself has received significant attention in multiagent literature due to its wide variety of applications ranging from robot patrol to border patrolling of large areas [3, 26, 40]. We complement these works by applying robust methods for Bayesian Stackelberg games to these domains and taking into account human anchoring biases when dealing with limited observations.

6.3. Human Biases and Observation Errors

Limited observability provides a different challenge which we addressed via support theory [57]. Related work in support theory has shown that people exhibit anchoring biases and that they are slow to update away from these biases [15, 16, 47]. Specifically people anchor on the uniform distribution (ignorance prior) for the occurrence of a discrete set of events until they have obtained information to make an evaluative assessment about the actual occurrence of those events. Once they have obtained information they are slow to update away from the ignorance prior until they become confident on their evaluative assessment. Combining these concepts, such as ε-optimal responses, robust optimization, and anchoring biases, in a novel context (Stackelberg games) we are better able to address human followers who may deviate from the expected optimal choices due to bounded rationality and limited observations.

Related work has also examined other issues surrounding observation in Stackelberg games. It has been shown that if the follower faces any cost for observing the leader's action, irrespective of the size of the cost, the leader will lose his value of commitment in all pure-strategy equilibria. However, there does still exist some mixed-strategy equilibrium that will fully preserve the first-mover advantage [58]. Similarly it has been shown that the first-mover advantage is lost when there is noise in the follower's observation for pure-strategy equilibria [4]. It was later shown that noisy Stackelberg equilibrium can exist in the case of mixed strategies [13]. Our work examines a different issue than either of these findings. Namely, we look at the case where the follower neither incurs a cost to make an observation nor makes noisy observations; rather, the follower must make observations one at a time in order to construct their belief of our policy and because of this constraint may not take the necessary number of observations to construct an accurate representation. In adversarial settings this is can be a reasonable assumption since adversaries will observe the security force for some time in advance and the security force will not know when they will take these observations or for how long. Previous work in the context of Stackelberg games has not addressed this type of observation uncertainty.

7. Summary

Stackelberg games are crucial in many multiagent applications, and particularly for security applications [8, 39]; for instance, these games are applied for security scheduling at the Los Angeles International Airport and the Federal Air Marshal Service [43, 56]. In such applications automated Stackelberg solvers

may create an optimal leader strategy. Unfortunately, the bounded rationality and limited observations of human followers challenge a critical assumption — that followers will act optimally — in existing Stackelberg solvers, which may lead to a severely under-performing strategy when the follower deviates from the optimal strategy. In fact, human decisions are guided by their bounded rationality and limited observations of probabilistic events, as opposed to utility maximizing rationality [16, 22, 47, 51, 52]. To apply Stackelberg games to any setting with people, this limitation must be addressed.

This paper provides the following key contributions. First, it provides a new robust algorithm, COBRA(α,ε), that includes two new key ideas for addressing human adversaries: (i) human anchoring biases drawn from support theory; (ii) robust approaches for MILPs to address human imprecision. To the best of our knowledge, the effectiveness of each of these key ideas against human adversaries had not been explored in the context of Stackelberg games. Furthermore, it was unclear how effective the combination of these ideas, being brought together from different fields, would be against humans. The second contribution is in providing experimental evidence that this new algorithm can perform statistically significantly better than existing algorithms and baseline algorithms when dealing with human adversaries as followers. Thirdly, our detailed experiments provide a solid initial grounding and heuristics for the right parameter settings for the α parameter within our COBRA(α,ε) algorithm. These conclusions are drawn from experiments done on four settings based on real deployed security systems, in 4 different observability conditions, involving 218 human subjects playing 2960 games in total. These results show that COBRA(α,ε) is likely better suited for applications dealing with human adversaries. Lastly, runtime analysis is provided for this algorithms showing that it maintains equivalent solution speeds compared to existing approaches.

8. Acknowledgments

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 grant number 2007-ST-061-000001. However, any opinions, findings, and conclusions or recommendations in this document are those of the authors and do not necessarily reflect views of the United States Department of Homeland Security. This work was also supported in part by the National Science Foundation grant number IIS0705587 and the Israel Science Foundation. F.

Ordóñez would also like to acknowledge the support of Conicyt, through Grant No. ACT87.

References

[1] M. Aghassi and D. Bertsimas. Robust Game Theory. Math, Program., 107(1-2):231-273, 2006.

[2] N. Agmon, S. Kraus, G. Kaminka, and V. Sadov. Adversarial Uncertainty in Multi-Robot Patrol. In IJCAI, 1811-1817, 2009.

[3] N. Agmon, V. Sadov, S. Kraus, and G. Kaminka. The Impact of Adversarial Knowledge on Adversarial Planning in Perimeter Patrol. In AAMAS, 2008.

[4] K. Bagwell. Commitment and Observability in Games. Games and Economic Behavior, 8(2):271-280, 1995.

[5] D. Braziuna and C. Boutilier. Elicitation of Factored Utilities. AI Magazine, 69-79, 2008.

[6] M. Breton, A. Alg, and A. Haurie. Sequential Stackelberg Equilibria in Two-Person Games. Optimization Theory and Applications, 59(1):71-97, 1988.

[7] G. Brewka, I. Niemela, and M. Truszczynski. Preferences and Nonmonotonic Reasoning. AI Magazine, 58-69, 2008.

[8] G. Brown, M. Carlyle, J. Salmerón, and K. Wood. Defending Critical Infrastructure. Interfaces 36(6):530-544, 2006.

[9] E. Brunner, M. Ulrich, M. Puri. Rank-Score Tests in Factorial Designs with Repeated Measures. 70(2):286-317, 1999.

[10] C. Camerer. Behavioral Game Theory: Experiments in Strategic Interaction. Princeton Press, 2003.

[11] J. Cardinal, M. Labbé, S. Langerman, and B. Palop. Pricing of Geometric Transportation Networks. In 17th Canadian Conference on Computational Geometry, 2005.

[12] V. Conitzer and T. Sandholm. Computing the Optimal Strategy to Commit to. In EC, 2006.

[13] E. van Damme and S. Hurkens. Games with Imperfectly Observable Commitment. Games and Economic Behavior, 21:282-308, 1997.

[14] C. Fox. Personal Communication.

[15] C. Fox and R. Clemen. Subjective Probability Assessment in Decision Analysis: Partition Dependence and Bias Toward the Ignorance Prior. Management Science, 51(9):1417-1432, 2005.

[16] C. Fox and Y. Rottenstreich. Partition Priming in Judgment Under Uncertainty. Psychological Science, 14:195-200, 2003.

[17] M. Friedman. The Use of Ranks to Avoid the Assumption of Normality Implicit in the Analysis of Variance. Journal of the American Statistical Association, 32 No. 100:675-701, 1937.

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

[19] Y. Gal, A. Pfeffer, F. Marzo, and B. Grosz. Learning Social Preferences in Games. In AAAI, 2004.

[20] Y. Gal and A. Pfeffer. Predicting People's Bidding Behavior in Negotiation. In AAMAS, 2005.

[21] J. C. Harsanyi and R. Selten. A Generalized Nash Solution for Two-Person Bargaining Games with Incomplete Information. Management Science, 18(5):80-106, 1972.

[22] J. Goldsmith and U. Junker. Preference Handling for Artificial Intelligence. AI Magazine, 9-12, 2008.

[23] S. Jong, K. Tuyls, and K. Verbeeck. Artificial Agents Learning Human Fairness. In AAMAS, 2008.

[24] D. Kahneman and A. Tversky. Subjective Probability: A Judgement of Representativeness. Cognitive Psychology, 3:430-454, 1972.

[25] D. Kahneman and A. Tversky. Prospect Theory: An Analysis of Decision under Risk. Econometrica, XLVII:263-291, 1979.

[26] C. Kiekintveld, M. Jain, J. Tsai, J. Pita, M. Tambe, and F. Ordóñez. Computing Optimal Randomized Resource Allocations for Massive Security Games. In AAMAS, 2009.

[27] D. J. Koehler and G James. Probability matching in choice under uncertainty: Intuition versus deliberation. Cognition, 113:123-127, 2009.

[28] Y. A. Korilis, A. A. Lazar, and A. Orda. Achieving Network Optima Using Stackelberg Routing Strategies. In IEEE/ACM Transactions on Networking, 1997.

[29] R. C. Larson. A Hypercube Queueing Model for Facility Location and Redistricting in Urban Emergency Services. Computers and OR, 1(1):67-95, 1974.

[30] G. Leitmann. On Generalized Stackelberg Strategies. Optimization Theory and Applications, 26(4):637-643, 1978.

[31] R. Lin, S. Kraus, J. Wilkenfeld, and James Barry. Negotiating with Bounded Rational Agents in Environments with Incomplete Information using an Automated Agent. Artificial Intelligence, 172(6-7):823-851, 2008.

[32] R. McKelvey and T. Palfrey. Quantal Response Equilibria for Normal Form Games. Games and Economic Behavior, X:6-38, 1995.

[33] J. von Neumann. Zur Theorie der Gesellschaftsspiele. Mathematische Annalen, 100:295-320, 1927.

[34] A. Nilim and L. E. Ghaoui. Robustness in Markov Decision Problems with Uncertain Transition Matrices. In NIPS, 2004.

[35] F. Ordóñez and N. E. Stier-Moses. Robust Wardrop Equilibrium. In NETCOOP, 2007.

[36] M. J. Osborne and A. Rubinstein. A Course in Game Theory. MIT Press, 1994.

[37] Y. Oshrat, R. Lin, and S. Kraus. Facing the Challenge of Human-Agent Negotiations via Effective General Opponent Modeling. In AAMAS, 377-384, 2009.

[38] C. Pahl-Wostl and E. Ebenhöh. Heuristics to Characterise Human Behavior in Agent Based Models. In iEMSs, 2004.

[39] P. Paruchuri, J. Marecki, J. Pearce, M. Tambe, F. Ordóñez, and S. Kraus. Playing Games for Security: An Efficient Exact Algorithm for Solving Bayesian Stackelberg Games. In AAMAS, 2008.

[40] P. Paruchuri, M. Tambe, F. Ordóñez, and S. Kraus. Security in Multiagent Systems by Policy Randomization. In AAMAS, 2006.

[41] A. Pentland and A. Liu. Modeling and Prediction of Human Behavior. Neural Computation, 11:229-242, 1999.

[42] R. Pew and A. Mavor. Modeling Human and Organizational Behavior: Application to Military Simulations. Washington, DC: National Academy Press, 1998.

[43] J. Pita, M. Jain, J. Marecki, F. Ordóñez, C. Portway, M. Tambe, C. Western, P. Paruchuri, and S. Kraus. Deployed ARMOR Protection: The Application of a Game Theoretic Model for Security at the Los Angeles International Airport. In AAMAS, 2008.

[44] J. Pita, M. Jain, F. Ordóñez, M. Tambe, S. Kraus, and R. Magori-Cohen. Effective Solutions for Real-World Stackelberg Games: When Agents Must Deal with Human Uncertainties. In AAMAS, 2009.

[45] P. Pu and L. Chen. User-Involved Preference Elicitation for Product Search and Recommender Systems. AI Magazine, 79-93, 2008.

[46] A. Rubinstein. Modeling Bounded Rationality. MIT Press, 1998.

[47] K. E. See, C. R. Fox, and Y. S. Rottenstreich. Between Ignorance and Truth: Partition Dependence and Learning in Judgment Under Uncertainty. Journal of Experimental Psychology: Learning, Memory, and Cognition, 32:1385-1402, 2006.

[48] R. Selton. A Reexamination of the Perfectness Concept for Equilibrium Points in Extensive Games. International Journal of Game Theory, 4:25-55, 1975.

[49] R. Selton. Evolutionary Stability in Extensive Two-person Games - Correction and Further Development. Math. Soc. Sci, 16:223-266, 1988.

[50] Y. Shoham, R. Powers, and T. Grenager. If Multi-agent Learning is the Answer, What is the Question?. Artificial Intelligence Journal 171(7):365-377, 2007.

[51] H. Simon. Rational Choice and the Structure of the Environment. Psychological Review, 63:129-138, 1956.

[52] H. Simon. Sciences of the Artificial. MIT Press, 1969.

[53] N. Stanton, P. Salmon, G. Walker, C. Baber, and D. Jenkins. Human Factors Methods: A Practical Guide for Engineering and Design. Ashgate Publiding, Ltd., 2005.

[54] C. Starmer. Developments in Non-Expected Utility Theory: The Hunt for a Descriptive Theory of Choice under Risk. Journal of Economic Literature, XXXVIII:332-382, 2000.

[55] S. Tijs. Nash Equilibria for Noncooperative n-Person Games in Normal Form. SIAM Review, 23(2):225-237, Apr. 1981.

[56] J. Tsai, S. Rathi, C. Kiekintveld, F. Ordóñez, and M. Tambe. IRIS - A Tool for Strategic Security Allocation in Transportation Networks. In AAMAS, 2009.

[57] A. Tversky and D. J. Koehler. Support Theory: A Nonextensional Representation of Subjective Probability. Psychological Review, 101:547-567, 1994.

[58] F. Vardy. The Value of Commitment in Stackelberg Games with Observation Costs. Games and Economic Behavior, 49(2):374-400, 2004.

[59] B. Von Stengel and S. Zamir. Leadership with Commitment to Mixed Strategies. In CDAM Research Report LSE-CDAM-2004-01, London School of Economics, 2004.

[60] R. R. Wilcox. How many discoveries have been lost by ignoring modern statistical methods? American Psychologist, 53(3):300-314, 1998.

[61] R. R. Wilcox. Introduction to Robust Estimation and Hypothesis Testing. Academic Press, 2005.

[62] Z. Yin, D. Korzhyk, C. Kiekintveld, V. Conitzer, and M. Tambe. Stackelberg vs. Nash in Security Games: Interchangeability, Equivalence, and Uniqueness. In AAMAS, 2010.

[63] K. K. Yuen. The Two-Sample Trimmed T for Unequal Population Variances. Biometrika, 61:165-170, 1974.

[64] Y. Rottenstreich and A. Tversky. Unpacking, Repacking, and Anchoring: Advances in Support Theory. Psychological Review, 104:406-415, 1997.

A. Reward Structures

Doors	0	1	2	3	4	5	6	7
Subject Reward	1	9	5	6	7	1	10	3
Subject Penalty	-2	-4	-3	-3	-3	-2	-4	-3
Pirate Reward	1	4	2	3	4	1	5	2
Pirate Penalty	-5	-8	-1	-6	-5	-1	-7	-7

Table 5: Reward structure one

Doors	0	1	2	3	4	5	6	7
Subject Reward	8	5	3	10	1	3	9	4
Subject Penalty	-3	-2	-3	-2	-3	-3	-2	-3
Pirate Reward	4	3	1	5	1	2	5	2
Pirate Penalty	-8	-10	-1	-8	-1	-3	-11	-5

Table 6: Reward structure two

Doors	0	1	2	3	4	5	6	7
Subject Reward	8	5	2	10	1	3	9	4
Subject Penalty	-3	-3	-3	-3	-3	-3	-3	-3
Pirate Reward	4	3	1	5	1	2	5	2
Pirate Penalty	-8	-5	-1	-10	-5	-3	-9	-6

Table 7: Reward structure three

Doors	0	1	2	3	4	5	6	7
Subject Reward	8	5	2	10	1	3	9	4
Subject Penalty	-3	-3	-3	-3	-3	-3	-3	-3
Pirate Reward	3	3	3	3	3	3	3	3
Pirate Penalty	-8	-5	-2	-10	-1	-3	-9	-4

Table 8: Reward structure four

B. Statistical Significance Tests

As noted earlier, we are unable to use classical tests such as the T-test to judge statistical significance and hence different types of tests were run on our data. Among our reward structures, reward structure four was to be treated separately as it is a zero-sum game. We explain the tests we used and the reasons these were chosen in the following.

Reward structure one and reward structure two are structures with higher penalties to the leader when the follower deviates from strong Stackelberg equilibrium (SSE) assumptions. Given this similarity in reward structures one and two and the similarity of the results they produced we first ran a two-way Friedman test for repeated observations [17] in the unobserved observation condition and a two-way Friedman test (not for repeated observations) in the 5, 20, and unlimited observation conditions between them. A two-way test is a test that examines two variables within an experiment. In our case the two variables are reward structure and algorithm. We ran these tests to be certain that reward structure had an impact on the results produced for reward structures one and two. If reward structure did not have an impact on the results produced the data sets can be combined into one large data set.

In the unobserved observation condition the conclusion based on this test was that the reward structures did have an impact on the results, however, in the 5, 20, and unlimited observation conditions we found that they did not. Since reward structure did not influence the results in these cases, the data sets from reward structures one and two were combined into one large data set for the 5, 20, and unlimited observation cases. This arrangement left reward structure three separated by itself (since reward structure four was kept separate as a zero-sum game), but when combining reward structure three with one and two, we find that it had an impact on performance and hence separation was justified. The null hypothesis in all of our tests is that the results produced between any two algorithms are actually identical. This hypothesis is rejected with a p-value of .05 or less. Given

this setup we had the following statistical tests.

First, the unobserved case was treated separately from other cases. In the unobserved case, subjects were only given the reward structure and were asked to make a choice. Consequently, the choice they made would be made irrespective of the actual strategy employed. Therefore we were able to take the choice made by an individual subject and employ it across DOBSS, COBRA(0,ε), COBRA(α,ε), COBRA(C,ε), UNIFORM, and MAXIMIN for a particular reward structure. This means that for a particular reward structure in the unobserved case, the door choice made by a single subject was used against all strategies. However, notice that a choice made in a single reward structure was not used for the other three reward structures. For the 5, 20, and unlimited observation cases it was necessary to obtain 40 sample points (subject door choices) for each algorithm in each reward structure. This leads to the following statistical significance tests:

- *Test run for the unobserved case:* We ran the Brunner-Puri test for repeated observations [9] in this case. We chose to run the Brunner-Puri test because the Brunner-Puri method is better suited to non-continuous distributions with discrete data points for one-way designs. It is also necessary to use a test that deals with repeated observations since the choices made by subjects were repeated across algorithms. An alternative and more well known method to the Brunner-Puri method is the Friedman test for repeated measures, however, the Brunner-Puri method is a more robust method.

- *Test run for the 5, 20, and unlimited observation cases:* For these cases we tested each of the algorithms separately (i.e. we did not use one subject's choice across all algorithms as in the unobserved case) so it is necessary to use a different type of test. Once again due to the non-normal distribution of our data we chose a more robust method for this test. In particular we chose to run Yuen's test for comparing trimmed means [63]. For our tests we used a standard 20% trimmed mean. A trimmed mean refers to a situation where a certain proportion of the largest and smallest sample points are removed and the remaining sample points are averaged. This is typically done to help reduce variance in data collections that may have extreme outliers that can skew data sets [60, 61].

C. Strategies, Expected Rewards, and Expected Response Percentages

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.37,2.5)	COBRA(.03,2.5)	MAXIMIN
Door 1	0	.06	0	0	.06	.56
Door 2	.58	.63	.77	.64	.63	.53
Door 3	.45	.21	0	.23	.21	0
Door 4	.51	.58	.81	.63	.58	.48
Door 5	.56	.51	.70	.52	.51	.37
Door 6	0	.06	0	0	.06	0
Door 7	.61	.55	.69	.55	.55	.44
Door 8	.27	.36	0	.40	.36	.59

Table 9: Mixed strategies for reward structure one.

	DOBSS	COBRA(0,ε)	COBRA(1,2.5))	COBRA(.54,2.5)	COBRA(.41,2.5)	MAXIMIN
Door 1	.55	.57	1	.67	.61	.53
Door 2	.44	.55	0	.60	.56	.64
Door 3	.18	0	0	.05	.13	0
Door 4	.67	.53	1	.62	.57	.48
Door 5	0	0	0	0	0	0
Door 6	.18	.31	0	.05	.13	.27
Door 7	.64	.61	1	.69	.65	.58
Door 8	.30	.41	0	.29	.32	.48

Table 10: Mixed strategies for reward structure two.

.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.75,2.5)	COBRA(.25,2.5)	MAXIMIN
Door 1	.58	.52	1	.71	.55	.48
Door 2	.43	.41	0	.60	.46	.35
Door 3	.09	0	0	0	0	0
Door 4	.65	.55	1	.70	.57	.52
Door 5	0	.17	0	0	0	.47
Door 6	.24	.26	0	0	.33	.17
Door 7	.62	.52	1	.68	.54	.49
Door 8	.35	.53	0	.28	.51	.48

Table 11: Mixed strategies for reward structure three.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.75,2.5)	COBRA(.25,2.5)	MAXIMIN
Door 1	.58	.58	1	.68	.58	.58
Door 2	.43	.43	0	.56	.43	.43
Door 3	.09	.09	0	0	.09	.09
Door 4	.65	.65	1	.73	.65	.65
Door 5	0	0	0	0	0	0
Door 6	.24	.24	0	0	.24	.24
Door 7	.62	.62	1	.70	.62	.62
Door 8	.35	.35	0	.32	.35	.35

Table 12: Mixed strategies for reward structure four.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.37,2.5)	COBRA(.03,2.5)	MAXIMIN	UNIFORM
Door 1	-5	-4.58	-5	-5	-4.60	**-1.62**	-2.78
Door 2	-.96	-.42	**1.35**	**-.29**	**-.36**	**-1.62**	-3.56
Door 3	.35	**-.35**	-1	**-.29**	**-.36**	-1	.11
Door 4	-1.37	-.79	**1.35**	**-.29**	-.73	**-1.62**	-2.67
Door 5	.05	**-.35**	**1.35**	**-.29**	**-.36**	**-1.62**	-1.67
Door 6	-1	-.86	-1	-1	-.86	-1	-.26
Door 7	**.38**	**-.35**	**1.35**	**-.29**	**-.36**	**-1.62**	-2.56
Door 8	-4.56	-3.68	-7	-3.32	-3.67	**-1.62**	-3.67

Table 13: Expected rewards for reward structure one.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.54,2.5)	COBRA(.41,2.5)	MAXIMIN	UNIFORM
Door 1	-1.32	**-1.09**	**4**	**.09**	**-.57**	**-1.63**	-3.56
Door 2	-4.21	-2.81	-10	-2.12	-2.69	**-1.63**	-5.19
Door 3	-.62	-1	-1	-.89	-.72	-1	-.26
Door 4	**.79**	**-1.09**	**5**	**.09**	**-.57**	**-1.63**	-3.19
Door 5	-1	-1	-1	-1	-1	-1	-.26
Door 6	-2.06	-1.44	-3	-2.72	-2.31	**-1.63**	-1.15
Door 7	-.64	**-1.09**	**5**	**.09**	**-.57**	**-1.63**	-5.08
Door 8	-2.88	-2.12	-5	-2.93	-2.75	**-1.63**	-2.41

Table 14: Expected rewards for reward structure two.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.75,2.5)	COBRA(.25,2.5)	MAXIMIN	UNIFORM
Door 1	-.92	**-1.66**	**4**	**.62**	**-1.31**	**-2.12**	-3.56
Door 2	-1.51	**-1.66**	-5	-.16	**-1.31**	**-2.12**	-2.04
Door 3	-.80	-1	-1	-1	-1	-1	-.26
Door 4	**-.21**	**-1.66**	**5**	**.62**	**-1.31**	**-2.12**	-4.45
Door 5	-5	-3.92	-5	-5	-5	**-2.12**	-2.78
Door 6	-1.76	**-1.66**	-3	-3	**-1.31**	**-2.12**	-1.15
Door 7	-.26	**-1.66**	**5**	**.62**	**-1.31**	**-2.12**	-3.82
Door 8	-3.16	-1.74	-6	-3.75	-1.87	**-2.12**	-3.04

Table 15: Expected rewards for reward structure three.

	DOBSS	COBRA(0,2.5)	COBRA(1,2.5)	COBRA(.75,2.5)	COBRA(.25,2.5)	MAXIMIN	UNIFORM
Door 1	**-1.51**	**-1.51**	**3**	**-.50**	**-1.51**	**-1.51**	-3.93
Door 2	**-1.51**	**-1.51**	-5	**-.50**	**-1.51**	**-1.51**	-2.04
Door 3	**-1.51**	**-1.51**	-2	-2	**-1.51**	**-1.51**	-.15
Door 4	**-1.51**	**-1.51**	**3**	**-.50**	**-1.51**	**-1.51**	-5.19
Door 5	-1	-1	-1	-1	-1	-1	.48
Door 6	**-1.51**	**-1.51**	-3	-3	**-1.51**	**-1.51**	-.78
Door 7	**-1.51**	**-1.51**	**3**	**-.50**	**-1.51**	**-1.51**	-4.56
Door 8	**-1.51**	**-1.51**	-4	-1.75	**-1.51**	**-1.51**	-1.41

Table 16: Expected rewards for reward structure four.

Structure One	Unobserved	5	20	Unlimited
DOBSS	20%	7.5%	17.5%	12.5%
COBRA(0,ε)	65%	65%	65%	70%
COBRA(α,ε)	57.5%	92.5%	72.5%	70%
COBRA(C,ε)	92.5%	92.5%	87.5%	95%
MAXIMIN	100%	100%	100%	100%
Structure Two				
DOBSS	27.5%	25%	12.5%	10%
COBRA(0,ε)	62.5%	65%	40%	55%
COBRA(α,ε)	62.5%	57.5%	47.5%	55%
COBRA(C,ε)	62.5%	57.5%	55%	47.5%
MAXIMIN	100%	100%	100%	100%
Structure Three				
DOBSS	20%	20%	20%	25%
COBRA(0,ε)	75%	72.5%	62.5%	60%
COBRA(α,ε)	50%	47.5%	67.5%	60%
COBRA(C,ε)	50%	47.5%	20%	25%
MAXIMIN	100%	100%	100%	100%
Structure Four				
DOBSS	100%	92.5%	87.5%	85%
COBRA(0,ε)	100%	92.5%	87.5%	85%
COBRA(α,ε	42.5%	52.5%	82.5%	85%
COBRA(C,ε)	52.5%	52.5%	25%	35%
MAXIMIN	100%	100%	100%	100%

Table 17: Percentage of times follower chose an *expected strategy*.

D. Strategies for varying α in COBRA(α,2.5)

	Door 1	Door 2	Door 3	Door 4	Door 5	Door 6	Door 7	Door 8
COBRA(0.00,2.5)	.069	.631	.213	.578	.515	.069	.553	.368
COBRA(0.05,2.5)	.062	.635	.208	.592	.513	.062	.552	.372
COBRA(0.10,2.5)	.056	.634	.203	.608	.512	.056	.550	.377
COBRA(0.15,2.5)	.051	.632	.198	.621	.510	.051	.549	.384
COBRA(0.20,2.5)	.050	.632	.194	.620	.509	.050	.548	.394
COBRA(0.25,2.5)	.049	.630	.190	.619	.507	.049	.547	.406
COBRA(0.30,2.5)	.046	.630	.187	.618	.506	.046	.546	.418
COBRA(0.35,2.5)	.005	.640	.227	.631	.520	.005	.556	.414
COBRA(0.40,2.5)	.000	.643	.242	.636	.525	.000	.560	.391
COBRA(0.45,2.5)	.000	.647	.256	.641	.529	.000	.564	.361
COBRA(0.50,2.5)	.000	.651	.272	.646	.535	.000	.568	.325
COBRA(0.55,2.5)	.000	.656	.292	.653	.542	.000	.573	.282
COBRA(0.60,2.5)	.000	.662	.318	.661	.550	.000	.579	.227
COBRA(0.65,2.5)	.000	.671	.350	.672	.561	.000	.587	.156
COBRA(0.70,2.5)	.000	.681	.393	.686	.575	.000	.598	.063
COBRA(0.75,2.5)	.000	.689	.423	.696	.585	.000	.605	.000
COBRA(0.80,2.5)	.000	.689	.423	.696	.585	.000	.605	.000
COBRA(0.85,2.5)	.000	.689	.423	.696	.585	.000	.605	.000
COBRA(0.90,2.5)	.000	.689	.423	.696	.585	.000	.605	.000
COBRA(0.95,2.5)	.000	.731	.227	.752	.641	.000	.647	.000
COBRA(1.00,2.5)	.000	.779	.000	.817	.706	.000	.696	.000

Table 18: α-variations for reward structure one.

	Door 1	Door 2	Door 3	Door 4	Door 5	Door 6	Door 7	Door 8
COBRA(0.00,2.5)	.575	.553	.000	.530	.000	.311	.618	.410
COBRA(0.05,2.5)	.579	.555	.000	.535	.000	.300	.622	.405
COBRA(0.10,2.5)	.584	.557	.006	.539	.000	.287	.625	.399
COBRA(0.15,2.5)	.587	.557	.026	.542	.000	.268	.628	.389
COBRA(0.20,2.5)	.591	.556	.048	.546	.000	.248	.631	.377
COBRA(0.25,2.5)	.595	.555	.074	.549	.000	.224	.634	.365
COBRA(0.30,2.5)	.600	.554	.103	.554	.000	.198	.637	.350
COBRA(0.35,2.5)	.606	.554	.136	.559	.000	.167	.642	.334
COBRA(0.40,2.5)	.617	.560	.139	.569	.000	.139	.650	.322
COBRA(0.45,2.5)	.634	.573	.114	.585	.000	.114	.663	.314
COBRA(0.50,2.5)	.654	.589	.084	.604	.000	.084	.678	.304
COBRA(0.55,2.5)	.679	.609	.046	.627	.000	.046	.697	.292
COBRA(0.60,2.5)	.711	.634	.000	.656	.000	.000	.720	.277
COBRA(0.65,2.5)	.730	.633	.000	.674	.000	.000	.735	.225
COBRA(0.70,2.5)	.757	.632	.000	.698	.000	.000	.755	.156
COBRA(0.75,2.5)	.793	.631	.000	.732	.000	.000	.782	.059
COBRA(0.80,2.5)	.828	.597	.000	.765	.000	.000	.809	.000
COBRA(0.85,2.5)	.863	.503	.000	.797	.000	.000	.635	.000
COBRA(0.90,2.5)	.933	.316	.000	.861	.000	.000	.887	.000
COBRA(0.95,2.5)	1.00	.000	.000	1.00	.062	.000	.937	.000
COBRA(1.00,2.5)	1.00	.000	.000	1.00	.000	.000	1.00	.000

Table 19: α-variations for reward structure two.

	Door 1	Door 2	Door 3	Door 4	Door 5	Door 6	Door 7	Door 8
COBRA(0.00,2.5)	.527	.416	.000	.555	.179	.266	.523	.531
COBRA(0.05,2.5)	.532	.423	.000	.559	.151	.277	.527	.529
COBRA(0.10,2.5)	.537	.431	.000	.563	.119	.290	.532	.526
COBRA(0.15,2.5)	.543	.440	.000	.568	.083	.304	.537	.523
COBRA(0.20,2.5)	.550	.450	.000	.573	.043	.320	.542	.520
COBRA(0.25,2.5)	.557	.460	.000	.579	.000	.337	.549	.516
COBRA(0.30,2.5)	.559	.464	.000	.580	.000	.342	.550	.502
COBRA(0.35,2.5)	.562	.468	.000	.583	.000	.345	.553	.487
COBRA(0.40,2.5)	.570	.481	.000	.590	.000	.320	.560	.476
COBRA(0.45,2.5)	.581	.496	.000	.598	.000	.290	.569	.464
COBRA(0.50,2.5)	.593	.514	.000	.607	.000	.254	.579	.450
COBRA(0.55,2.5)	.607	.536	.000	.619	.000	.210	.592	.432
COBRA(0.60,2.5)	.626	.564	.000	.634	.000	.155	.608	.410
COBRA(0.65,2.5)	.650	.600	.000	.653	.000	.085	.628	.381
COBRA(0.70,2.5)	.687	.615	.000	.683	.000	.001	.660	.352
COBRA(0.75,2.5)	.719	.604	.000	.708	.000	.000	.687	.280
COBRA(0.80,2.5)	.766	.587	.000	.746	.000	.000	.728	.171
COBRA(0.85,2.5)	.842	.556	.000	.807	.000	.000	.293	.000
COBRA(0.90,2.5)	.908	.381	.000	.860	.000	.000	.850	.000
COBRA(0.95,2.5)	1.00	.071	.000	1.00	.000	.000	.928	.000
COBRA(1.00,2.5)	1.00	.000	.000	1.00	.000	.000	1.00	.000

Table 20: α-variations for reward structure three.

	Door 1	Door 2	Door 3	Door 4	Door 5	Door 6	Door 7	Door 8
COBRA(0.00,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.05,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.10,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.15,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.20,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.25,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.30,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.35,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.40,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.45,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.50,2.5)	.589	.435	.096	.652	.000	.247	.623	.354
COBRA(0.55,2.5)	.600	.451	.010	.662	.000	.268	.634	.372
COBRA(0.60,2.5)	.609	.462	.000	.669	.000	.231	.641	.285
COBRA(0.65,2.5)	.624	.482	.000	.681	.000	.146	.655	.409
COBRA(0.70,2.5)	.651	.520	.000	.705	.000	.048	.680	.393
COBRA(0.75,2.5)	.681	.561	.000	.730	.000	.000	.707	.320
COBRA(0.80,2.5)	.712	.604	.000	.756	.000	.000	.736	.190
COBRA(0.85,2.5)	.787	.477	.000	.819	.000	.000	.804	.010
COBRA(0.90,2.5)	.852	.406	.000	.875	.000	.000	.865	.000
COBRA(0.95,2.5)	1.00	.000	.000	1.00	.000	.000	1.00	.000
COBRA(1.00,2.5)	1.00	.000	.000	1.00	.000	.000	1.00	.000

Table 21: α-variations for reward structure four.