Artificial Intelligence 240 (2016) 65–103

Contents lists available at ScienceDirect

Artificial Intelligence

www.elsevier.com/locate/artint

Comparing human behavior models in repeated Stackelberg security games: An extended study ☆

Debarun Kar [a,*], Fei Fang [a,**], Francesco M. Delle Fave [b,**], Nicole Sintov [a,**], Milind Tambe [a,**], Arnaud Lyet [c,**]

[a] University of Southern California, SAL 300, 941 Bloom Walk, Los Angeles, CA 90089, USA
[b] Disney Research, 222 Third Street, Cambridge, MA 02142, USA
[c] World Wildlife Fund, 1250 24th Street, N.W. Washington, DC 20037, USA

ARTICLE INFO

Article history:
Received 5 July 2015
Received in revised form 5 August 2016
Accepted 10 August 2016
Available online 16 August 2016

Keywords:
Game theory
Repeated Stackelberg games
Human behavior modeling

ABSTRACT

Several competing human behavior models have been proposed to model boundedly rational adversaries in repeated Stackelberg Security Games (SSG). However, these existing models fail to address three main issues which are detrimental to defender performance. First, while they attempt to learn adversary behavior models from adversaries' past actions ("attacks on targets"), they fail to take into account adversaries' future adaptation based on successes or failures of these past actions. Second, existing algorithms fail to learn a reliable model of the adversary unless there exists sufficient data collected by exposing enough of the attack surface – a situation that often arises in initial rounds of the repeated SSG. Third, current leading models have failed to include probability weighting functions, even though it is well known that human beings' weighting of probability is typically nonlinear.

To address these limitations of existing models, this article provides three main contributions. Our first contribution is a new human behavior model, SHARP, which mitigates these three limitations as follows: (i) SHARP reasons based on success or failure of the adversary's past actions on exposed portions of the attack surface to model adversary adaptivity; (ii) SHARP reasons about similarity between exposed and unexposed areas of

☆ This journal article extends a full paper that appeared in AAMAS 2015 [49] with the following new contributions. First, we test our model SHARP in human subjects experiments at the Bukit Barisan Seletan National Park in Indonesia against wildlife security experts and provide results and analysis of the data (Section 13.1). Second, we conduct new human subjects experiments on Amazon Mechanical Turk (AMT) to show the extent to which past successes and failures affect the adversary's future decisions in repeated Stackelberg games (Section 8.1). Third, we conduct new analysis on our human subjects data and illustrate the effectiveness of SHARP's modeling considerations and also the robustness of our experimental results by: (i) showing how SHARP based strategies adapt due to past successes and failures of the adversary, while existing competing models like P-SUQR converge to one particular strategy (Section 12.4); (ii) comparing a popular probability weighting function in the literature (Prelec's model) against the one used in SHARP and showing how the probability weighting function used in SHARP is superior in terms of prediction performances, even though the shape of the learned curves are the same (Sections 3.2 and 12.2.1); (iii) comparing an alternative subjective utility function based on prospect theory where the values of outcomes are weighted by the transformed probabilities, against the weighted-sum-of-features approach used in SHARP – the alternative model yields the same surprising S-shaped probability weighting curves as the weighted-sum-of-features functional form used in SHARP but the weighted-sum-of-features model yields better prediction accuracy than the prospect theoretic subjective utility function (Sections 7.2 and 12.2.2); and (iv) proposing and comparing a new descriptive reinforcement learning (rl) model for SSGs which is based on a popular reinforcement learning model for simultaneous move games against SHARP – although the rl model learns based on feedback from past actions, it performs poorly as compared to SHARP (Sections 11 and 12.1). Fourth, in this article we also provide methodological contributions towards conducting repeated measures experiments on AMT and show the effects of various strategies on the participant retention rates in such repeated experiment settings (Section 6). Fifth, we discuss additional related work (Section 3), directions for future work (Section 14), and provide additional detailed explanations, proofs of theorems and feedback from participants who played our games.

* Principal corresponding author.
** Corresponding authors.
E-mail addresses: dkar@usc.edu (D. Kar), feifang@usc.edu (F. Fang), francesco.dellefave@disneyresearch.com (F.M. Delle Fave), sintov@usc.edu (N. Sintov), tambe@usc.edu (M. Tambe), Arnaud.Lyet@wwfus.org (A. Lyet).

http://dx.doi.org/10.1016/j.artint.2016.08.002

the attack surface, and also incorporates a discounting parameter to mitigate adversary's lack of exposure to enough of the attack surface; and (iii) SHARP integrates a non-linear probability weighting function to capture the adversary's true weighting of probability. Our second contribution is a first "repeated measures study" – at least in the context of SSGs – of competing human behavior models. This study, where each experiment lasted a period of multiple weeks with individual sets of human subjects on the Amazon Mechanical Turk platform, illustrates the strengths and weaknesses of different models and shows the advantages of SHARP. Our third major contribution is to demonstrate SHARP's superiority by conducting real-world human subjects experiments at the Bukit Barisan Seletan National Park in Indonesia against wildlife security experts.

1. Introduction

Whereas previous real-world deployments of Stackelberg Security Games (SSGs) to protect airports, ports or flights have been one-shot game models [75], recent work has focused on domains involving repeated interactions between defenders and adversaries. These domains include "Green Security Game" domains [31], i.e. security of wildlife (repeated interactions between rangers and poachers) [80], security of fisheries (repeated interactions between coast guard and illegal fishermen) [40] and forest protection. The above domains as well as other domains such as drug interdiction are modeled via repeated SSGs. In a repeated SSG model, the defender periodically deploys new patrol strategies (in "rounds" of the game) and the adversaries observe these strategies and act accordingly.

Research in repeated SSGs has produced different approaches to address uncertainty in key dimensions of the game such as payoff uncertainty (but assuming a perfectly rational adversary) [14,54,56] or uncertainty in adversary behavior model (but assuming known payoffs) [40,80]. Our work follows the second approach, learning a model of boundedly rational adversaries with known adversary payoffs, as (arguably) it provides a better fit for domains of interest in this work. This is because (i) in real-world settings such as wildlife or fishery protection, it is feasible to model adversary payoffs via animal or fish density in different locations and we can get a reasonable estimation of these densities from animal census data collected by patrolling rangers or conservation drones; and (ii) there is significant support in the literature for bounded rationality of human adversaries [81,62].

Unfortunately, despite the promise of Bounded Rationality models in Repeated Stackelberg Games (henceforth referred to as BR-RSG models), existing work in this area [40,80] suffers from three key limitations which are extremely detrimental to defender performance. First, existing models reason about the adversary's future actions based on past actions taken but *not* the associated successes and failures. Our analysis reveals that the adversaries adapt in future rounds based on past successes and failures. Hence, failing to consider an adaptive adversary leads to erroneous predictions about his[1] future behavior, and thus significantly flawed defender strategies.

Second, existing approaches for learning BR-RSG models require significant amount of data to be collected in the initial rounds to learn a reliable adversary model. Furthermore, our analysis reveals that the issue is not just the amount of data, but insufficient exposure of *attack surface* [45,55] in the initial rounds which prevents the defender from collecting sufficient information about adversary responses to various strategies and learn a reliable model. Here we define an *attack surface* as the n-dimensional feature space used to model the attacker's behavior (detailed definition in Section 8). This issue of limited attack surface exposure leads to erroneous learned results as the learning is biased towards the limited available information and hence significant losses are incurred by the defender until enough of the *right kind of data* becomes available. This degraded performance in initial rounds may have severe consequences for three reasons: (i) In domains like wildlife crime or fisheries protection, each round lasts for weeks or potentially months and so initial round losses (if massive) could imply irrecoverable losses of resources (e.g., animal populations). (ii) Following heavy losses, human defenders may lose confidence in recommendations provided by the game-theoretic approach. (iii) Given the nature of these domains, re-initialization of the game may periodically be necessary and thus initial rounds may be repeated; in domains such as wildlife crime, re-initialization can stem from man-made or natural changes in parks, e.g., changes in vegetation, water bodies, or possible developmental activities. The construction of an oil-refinery [1] and the simultaneous re-introduction of rhinos in the Murchison Falls National Park in Uganda is an example. In other words, re-initializing the game after a year or so would mean repeating the initial rounds after four to five rounds, adding to the importance of addressing initial rounds.

Finally, BR-RSG models have failed to include probability weighting functions (how humans "perceive" probabilities), even though it is well known that probability weighting curves for humans – e.g., in prospect theory [77] – are typically nonlinear. In light of this, we show that direct application of existing models such as SUQR [62] which assume a linear probability model, provide results that would be extremely detrimental to defender performance.

The primary contribution of this article is a new model called SHARP (**S**tochastic **H**uman behavior model with **A**tt**R**activeness and **P**robability weighting) that mitigates these three limitations: (i) Modeling the adversary's adaptive deci-

[1] By convention in security games literature, the defender is referred to as "she" and the adversary as "he".

sion making process, SHARP reasons based on success or failure of the adversary's past actions on exposed portions of the attack surface; (ii) Addressing limited exposure to significant portions of the attack surface in initial rounds, SHARP reasons about similarity between exposed and unexposed areas of the attack surface, and also incorporates a discounting parameter to mitigate adversary's lack of exposure to enough of the attack surface; (iii) Addressing shortcomings of existing models in learning the adversaries' weighting of probabilities, we incorporate a two parameter probability weighting function in existing human behavior models.

Our second main contribution is to provide evidence from the first "repeated measures study" of competing models in repeated SSGs. A repeated measures study is an observational research method in which data is gathered for the same subjects repeatedly over a period of time, sometimes spanning years or even decades. In our study, a suite of well-established models and SHARP are compared in human subjects experiments on the Amazon Mechanical Turk (AMT). In conducting these repeated measures experiments with AMT, we also provide a procedural contribution – specifically an empirically validated study of the process of conducting repeated measures experiments on AMT. We address challenges faced while conducting several 2.3 week long (on average) repeated measures studies, 46 weeks in total, with various behavioral models and show that our methods help maintain high retention rates of the participants throughout the course of the study and therefore enables valid comparison of the various models.

Our third contribution is to analyze and show results from our human subjects experiments on repeated SSGs. We show that: (i) SHARP outperforms existing approaches consistently over all rounds, most notably in initial rounds. (ii) As discussed earlier, existing approaches perform poorly in initial rounds with some performing poorly throughout all rounds. (iii) Surprisingly, simpler models which were originally proposed for single-shot games performed better than recent advances which are geared specifically towards addressing repeated SSGs. We also conducted a different set of human subjects experiments on AMT to examine the extent to which past successes and failures affect the adversary's future decisions in repeated Stackelberg game settings.

In order to validate the findings from our AMT experiments, we conducted one repeated measures study for SHARP in the real world: with wildlife security experts from the provinces of Lampung and Riau, Sumatra, Indonesia. Participants were from the local government and from the following NGOs: Yayasan Badak Indonesia (YABI), World Wildlife Fund (WWF) and Wildlife Conservation Society (WCS). The experiments were conducted at the Bukit Barisan Seletan National Park in Indonesia. The results are consistent with the findings from our experiments on AMT. In fact, the performance for SHARP is better against experts as compared to those obtained from AMT. Taken together, given the critical importance of the repeated 'initial rounds' as discussed above, these results indicate that SHARP should be the model of choice in repeated SSGs.

The rest of the article is organized as follows. In Section 2, we first provide background for our work, followed by related work in Section 3. In Section 4, we discuss our wildlife poaching game which was used to conduct the repeated measures experiments. This is followed by a description of our payoff structures for the wildlife game in Section 5. In Section 6, we briefly highlight the results of our methodological contributions towards conducting our repeated measures experiments on AMT. In Sections 7, 8, 9 and 10 we discuss in detail the three modeling considerations of SHARP, the process of generating the optimal defender strategy against SHARP and an analysis of various modeling considerations of SHARP. In Section 11 we first discuss the results of our AMT experiments and then in Section 12 we provide real-world experimental results against security experts to validate our AMT findings. Finally, we conclude with a summary and directions for future work in Section 13.

2. Background

In this section we first introduce Stackelberg Security Games (SSG) and key solution concepts and existing human behavioral models used to model the boundedly rational adversaries in SSGs.

In an SSG, the defender plays the role of a leader who protects a set of targets from the adversary, who acts as the follower [20,65,51]. The defender's pure strategy is an assignment of a limited number of security resources M to the set of targets T. An assignment of a resource to a target is also referred to as covering a target. A defender's mixed-strategy $\hat{x}$ ($0 \leq \hat{x}_j \leq 1;\ \forall \hat{x}_j,\ j \in P;\ \sum_{j=1}^{P} \hat{x}_j = 1$) is then defined as a probability distribution over the set of all possible pure strategies P. An equivalent description [51,81] of these mixed strategies is a probability distribution over the set of targets: x ($0 \leq x_i \leq 1;\ \forall x_i,\ i \in T;\ \sum_{i=1}^{T} x_i = M$). In the rest of this article, we will refer to this latter description as the mixed strategy of the defender.

A pure strategy of an adversary is defined as attacking a single target. The adversary receives a reward R_i^a for selecting i if it is not covered and a penalty P_i^a for selecting i if it is covered. Similarly, the defender receives a reward R_i^d for covering i if it is selected by the adversary and penalty P_i^d for not covering i if it is selected. Then, the expected utility for the defender (while playing mixed strategy x) when target i is selected by the adversary to attack is:

$$U_i^d(x) = x_i R_i^d + (1 - x_i) P_i^d \tag{1}$$

Similarly, the expected utility for the adversary for attacking target i is:

$$U_i^a(x) = (1 - x_i) R_i^a + x_i P_i^a \tag{2}$$

Although a perfectly rational adversary would choose to attack the target with the highest expected utility, more recent work has focused on modeling boundedly rational adversaries in SSGs [62,80,40,81,34,22], thus developing models, some of which are discussed in Sections 2.2, 2.3 and 2.4.

2.1. Repeated Stackelberg security games

As introduced earlier in Section 1, in this article we focus on a repeated Stackelberg Security Game (SSG) consisting of multiple rounds. A repeated SSG setting is different from the traditional repeated game setting [63] in the following ways. First, in a repeated SSG, in one round, one player acts first by deploying a mixed strategy and then the other player responds. Intuitively, one round in a repeated SSG corresponds to several ($>> 1$) consecutive rounds in a repeated game. Second, in a repeated SSG, the mixed strategy of the defender may change at the end of one round leading to a new mixed strategy, while such a concept of a change of mixed strategy is not part of a traditional repeated game [63]. In other words, just as a Stackelberg Security Game focuses on commitment to a mixed strategy in a round, rather than commitment to a pure strategy as done in earlier literature on Stackelberg games [8], repeated SSG focuses on mixed strategies in each round.

2.2. Subjective utility quantal response (SUQR)

SUQR [62] builds upon prior work on quantal response [58] according to which rather than strictly maximizing utility, an adversary stochastically chooses to attack targets, i.e., the adversary attacks a target with higher expected utility with a higher probability. SUQR proposes a new utility function called Subjective Utility, which is a linear combination of key features that are considered to be the most important in each adversary decision-making step. This is based on the Lens model in psychology which is a framework for modeling prediction based on observable cues [15,39]. Usually these observable cues are combined in a weighted fashion to get the utility of the decision maker. Nguyen et al. [62] experimented with three features: defender's coverage probability, adversary's reward and penalty at each target. Thus, according to this model, the probability that the adversary will attack target i is given by:

$$q_i(\omega|x) = \frac{e^{SU_i^a(x)}}{\sum_{j\in T} e^{SU_j^a(x)}} \tag{3}$$

where $SU_i^a(x)$ is the Subjective Utility of an adversary for attacking target i when defender employs strategy x and is given by:

$$SU_i^a(x) = \omega_1 x_i + \omega_2 R_i^a + \omega_3 P_i^a \tag{4}$$

The vector $\omega = (\omega_1, \omega_2, \omega_3)$ encodes information about the adversary's behavior and each component of ω indicates the relative importance the adversary gives to each attribute in the decision making process. The weights are computed by performing Maximum Likelihood Estimation (MLE) on available attack data.

2.3. Bayesian SUQR

SUQR assumes that there is a homogeneous population of adversaries, i.e., a single ω is used to represent an adversary in [62]. However, in the real-world we face an entire population of heterogeneous adversaries. So Yang et al. [80] introduces a set $\Omega \subset \mathbb{R}^3$ to represent the range of all possible ω, i.e. the entire set of adversaries. Therefore Bayesian SUQR is proposed to learn a particular value of ω for each attack. It assumes that there is a prior distribution F over Ω. Bayesian updates are performed on F as more data becomes available. Then the following stochastic optimization problem is solved to obtain the optimal strategy x:

$$\max_{x\in\mathbb{X}} \int_{\Omega} \left[\sum_{t\in T} U_t^d(x)\, q_t(\omega \mid x) \right] F(d\omega) \tag{5}$$

Protection Assistant for Wildlife Security (PAWS) is an application which was originally created using Bayesian SUQR. Recent work by Fang et al. [31] has also used this notion of a heterogeneous population of boundedly rational adversaries and applied Bayesian updating based algorithms to learn models of these adversaries.

2.4. Robust SUQR

Robust SUQR [40] combines data-driven learning and robust optimization to address settings where not enough data is available to provide a reasonable hypothesis about the distribution of ω. It does not require a specific distribution F over the adversary population parameters. Given an uncertainty set $\widehat{\Omega}$, Robust SUQR solves the following robust optimization problem:

$$\max_{x\in\mathbb{X}} \min_{\omega\in\widehat{\Omega}} \sum_{t\in\mathbb{T}} U_t(x)\, q_t(\omega \mid x) \tag{6}$$

We now explain Eqn. (6) starting with the uncertainty set $\widehat{\Omega}$. There are various ways to construct the uncertainty set $\widehat{\Omega}$. Haskell et al. [40] suggests combining the robust optimization in Eqn. (6) with a data-driven approach by using the set of all ω learned from each attack by the adversary as the uncertainty set $\widehat{\Omega}$. Therefore, Robust SUQR computes the worst-case expected utility over all previously seen SUQR models of the adversary and deploys the optimal strategy against the adversary type that reduces the defender's utility the most. Robust SUQR has been applied to fisheries protection domain [40].

3. Related work

We have already discussed related work in SSGs in the previous section, including key behavioral models. Here we discuss six additional areas of related work:

3.1. Related research in repeated games

Here we discuss past work on repeated games which are relevant to our setting.

3.1.1. Learning in repeated Stackelberg games

The problem of learning the adversary's payoffs to alleviate uncertainty in an SSG by launching a minimum number of games against a perfectly rational adversary is studied in [54,14]. Letchford et al. [54] propose an approach to learn a single attacker's payoffs by making a number of best-response queries which is polynomial in the number of pure strategies. A query here refers to the defender's execution of a mixed strategy, and letting an adversary respond, thereby providing information about adversary's payoffs. This work was the first in the security game context for learning adversary payoffs. They extend their results to Bayesian Stackelberg games with a known distribution over attacker types by running the single-attacker learning algorithm, where they repeat each best response query until the response of the desired attacker type is observed.

Noticing that Letchford et al. [54] may still lead to a large number of queries, particularly given that number of pure strategies may grow exponentially, Blum et al. [14] design an algorithm that learns an ϵ-optimal strategy for the defender with a certain probability by asking a significantly lower number of queries. However, Blum et al. [14] only study the interaction between the defender and a single attacker.

Building upon previous work [14,54] as described above, Balcan et al. [9] provides two contributions in terms of learning the randomized defender strategy to commit to in each round against perfectly rational adversaries: (i) an online learning algorithm where the defender observes the adversary type that is attacking a particular target (full-information); and (ii) an online learning algorithm where the defender only observes a particular target being attacked in each round (partial information). In each interaction, the attacker is assumed to be adversarially chosen from a set of known attacker types.

Additionally, Marecki et al. [56] focused on optimizing the defender's overall utility during the learning process when faced with a perfectly rational adversary with unknown payoffs. Their analysis is focused on the repeated interaction between the defender and a single attacker type drawn initially from a distribution. Although their algorithm is shown to converge in the long-term, they do not provide any guarantees for the convergence of their algorithm.

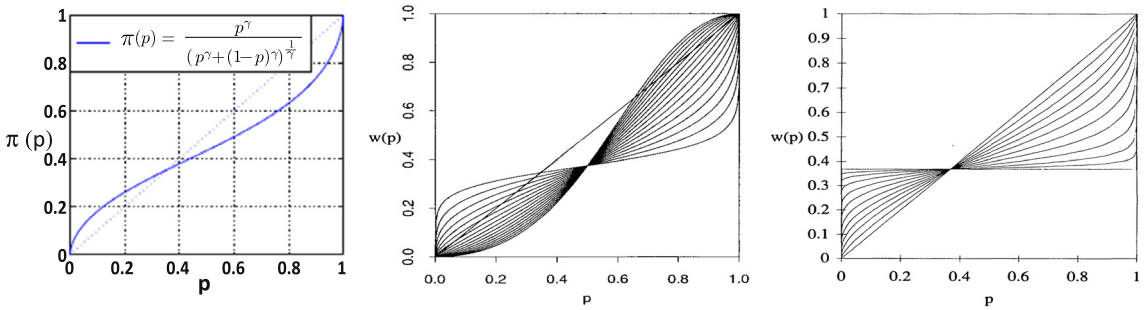
3.1.2. Robust strategies in repeated games

In cases where the opponent cannot be successfully modeled, McCracken et al. [57] proposed techniques to generate ϵ-safe strategies which bound the loss from a safe value by ϵ. Johanson et al. [47,46] studied the problem of generating robust strategies in a repeated zero-sum game while exploiting the tendency in the adversary's decision making and evaluated their technique in a game of two-player, Limit Texas Hold'em. Following up on this work, Johanson et al. [46] proposed methods to minimize losses due to limited data while also exploiting an unknown opponent's weaknesses and evaluated their technique in a game of two-player, Limit Texas Hold'em. Recently, Ponsen et al. [68] proposed techniques to compute robust best responses in partially observable stochastic games using sampling methods.

All of the above work differs from ours in three ways: (i) They do not model bounded rationality in human behavior; (ii) They do not consider how humans weigh probabilities; and (iii) None of these existing work address the important problem of significant initial round losses. Initial round losses is a critical problem in domains such as wildlife security as explained above; requiring a fundamental shift at least in the learning paradigm for SSGs. Work on learning in SSGs differ because in our game, the payoffs are known but we are faced with boundedly rational adversaries whose parameters in their behavioral model are to be learned.

3.2. Probability weighting functions

Probability weighting functions model human perceptions of probability. Perhaps the most notable is the weighting function in Tversky and Kahneman's Nobel Prize-winning work on Prospect Theory [48,77], which suggests that people weigh probability non-uniformly, The empirical form of the probability weighting function $\pi(p_i)$, where p_i is the actual probability, from [48] is shown in Fig. 1(a). It indicates that people tend to overweight low probabilities and underweight high probabilities. The diagonal straight line in the figure indicates the linear weighting of probability. However, other works in this domain propose and experiment with parametric models which capture both inverse S-shaped as well as S-shaped



(a) Prospect Theory (b) Gonzalez & Wu, 99 (c) Prelec, 98

Fig. 1. Probability weighting functions.

probability curves [3,37] (Fig. 1(b)). We build on this research, incorporating probability weighting functions in SHARP that allow for both S-shaped and inverse S-shaped curves; however, in our work, data supports S-shaped probability curves. Further discussions about this function are in Section 7.

There are other popular probability weighting functions in the literature, such as Prelec's one-parameter model [69], where the weighted probability is

$$w(p) = exp(-(-\ln p)^{\alpha}); 0 < \alpha < 1 \tag{7}$$

Although this model has been shown to perform well in the literature, the functional form does not allow for an S-shaped curve to be learned given the allowed range of parameter values – it is only capable of learning an inverse S-shaped curve as shown in Fig. 1(c) when $0 < \alpha < 1$. This parameter range of α is due to the necessity that the function satisfies certain properties such as sub-proportionality and compound invariance, which will get violated if $\alpha > 1$. However, it can account for S-shaped curves if we allow $\alpha > 1$. Later in Section 12.2.1, we allow α to be greater than 1 so as to allow learning both an S-shaped as well as an inverse S-shaped curve with this function – our results show that an S-shaped curve is learned on our data. In other words, no matter whether we use Prelec's function or Gonzalez and Wu's function, if we allow for learning both S-shaped as well as inverse S-shaped curves, our data fits an S-shaped probability weighting curve. We conduct further analysis to show in Section 12.2.1 that, even though both generate S-shaped curves on our data, using the probability weighting function by Gonzalez and Wu [37] in our model gives us better prediction accuracy as compared to the case when we use Prelec's function, thus justifying our choice of the probability weighting function in Section 7.

3.3. Repeated measures studies

Repeated measures studies are conducted to measure a set of variables over a period of time. Repeated measures studies are usually conducted in psychology, political sciences and social sciences and the duration of the experiments can span from a few weeks [71] to even a few years [16].[2] Recently, AMT has become a more favorable choice of conducting these experiments due to the ease of collecting data from a huge and diverse subject pool [64,12]. One of the most important problems in conducting repeated measures studies is handling participant attrition (i.e., people dropping out). While researchers often use imputation and sampling techniques to fill missing data due to participant attrition [78,36,23], for our repeated measures study of comparing human behavior models this may result in extremely biased estimates of the modeling parameters due to the influence of the retained participants' game plays and therefore may generate biased defender strategies.

3.4. Learning from reinforcements

Skinner [72–74] first proposed the theory of operant conditioning in which he explained through experiments that behavior which is reinforced tends to be repeated (i.e. strengthened) and behavior which is not reinforced tends to die out-or be extinguished (i.e. weakened). This reinforcement or lack thereof, happens due to actions and its associated consequences. More specifically, Skinner identified three types of responses or operants that can alter behavior, the most relevant among them are: (i) Reinforcers: Responses from the environment that increase the probability of a behavior being repeated; and (ii) Punishers: Responses from the environment that decrease the likelihood of a behavior being repeated. Since this early research, such behavior where the subject learns based on "superstitious" beliefs due to past actions and consequences has come to be known as superstitious learning [24,84] in psychology. The influence of superstitious learning on the adaptive behavior of humans has also been studied in the literature [10]. We will show later how superstitious learning, induced

[2] Whereas "repeated measures study" is often used to describe research that spans years – in which measurement occasions are conducted every X years – we use the term repeated measures study because our study included multiple (5) measurement points with a single population.

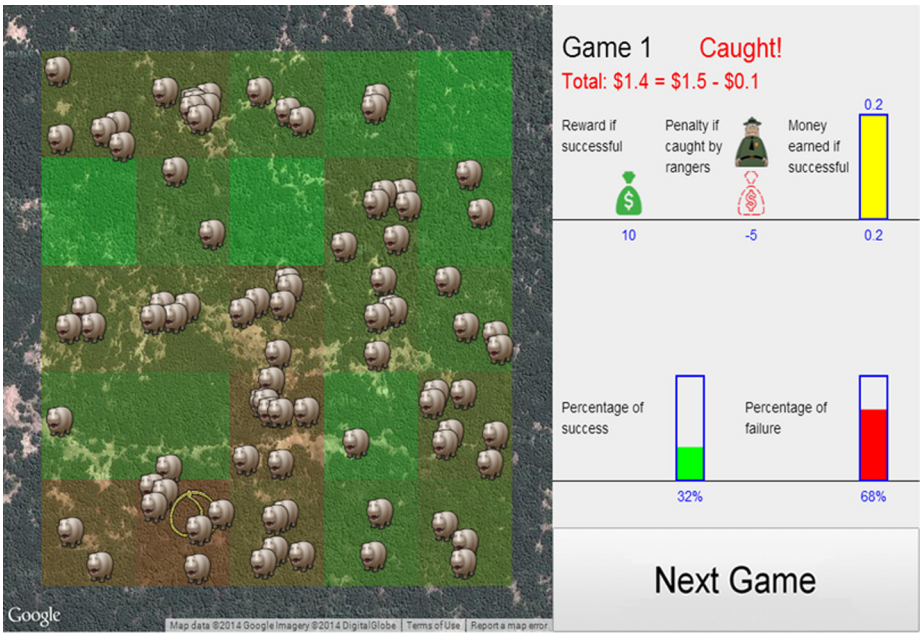


Fig. 2. Game interface for our simulated online repeated SSG (reward, penalty and coverage probability for a selected cell are shown). (For interpretation of the references to color in this figure, the reader is referred to the web version of this article.)

by these reinforced/punished responses plays a crucial role in predicting human subject behavior in repeated Stackelberg security games. More specifically, SHARP models this phenomenon and that leads to a significant improvement in SHARP's performance.

"Superstitious Learning", as it is widely known in the psychology literature, is closely related to "Reinforcement Learning" in the computer science literature. Reinforcement learning has been widely studied in the context of game theory [28,33,43,11,19,17]. Erev and Roth [28] describes one such popular Reinforcement Learning (RL) model that predicts people's behaviors while playing repeated, simultaneous move games. However, the paper has major differences as compared to our game setting. First, the models in Erev and Roth's paper were developed for simultaneous move games without any notion of prior commitment to a mixed strategy by any player. This is different from our leader–follower setting where one player moves first by playing a mixed strategy and the other player moves next by playing a pure strategy after having observed the first player's mixed strategy. Therefore, the notion of surveillance and reacting to a mixed strategy by the adversary is missing in [28]. Second, in our games, the defender responds after each round by playing an optimal mixed strategy based on the learned adversary model from past rounds' data. Assuming an adversary who follows an RL model as described in [28], the optimal defender response would then be to play a pure strategy in each round given the fixed adversary strategy in each round. This will have significantly detrimental effects in terms of the defender utilities obtained in each round, as shown in Section 7.1. Therefore, while not directly applicable, we attempt to translate the main concept of the RL model to our setting of leader–followers with mixed strategies. To that end, in Section 11, we have adapted the basic RL model in [28] to compute optimal mixed strategies for the defender in repeated SSG settings. In Section 12, we also show results of conducting human subjects experiments with this model. Specifically, the RL based approach performs poorly as compared to other models in our experiments. Therefore, significant new work would need to be done to understand how the RL models in the literature [28,33,43,11,19] could be adapted more efficiently for our Stackelberg Security Game setting.

4. Wildlife poaching game

We conducted *repeated measures experiments* with human subjects to test the effectiveness of existing behavioral models and algorithms against our proposed approach on repeated SSGs. In order to conduct these experiments, we developed an online simulated game. Below is an overview of our experimental game and its properties.

4.1. Game overview

In our game, human subjects play the role of poachers looking to place a snare to hunt hippopotamus in a protected park. The game interface is shown in Fig. 2. In the game, the portion of the park shown in the map is divided into a $5 * 5$ grid, i.e. 25 distinct cells. Overlaid on the Google Maps view of the park is a heat-map, which represents the rangers' mixed strategy x – a cell i with higher coverage probability x_i is shown more in red, while a cell with lower coverage probability is shown more in green. As the subjects play the game, they are given the following detailed information: R_i^a, P_i^a and x_i

for each target i. However, they do not know the exact location of the rangers, i.e., they do not know the pure strategy that will be played by the rangers, which is drawn randomly from the mixed strategy x shown on the game interface. Thus, we model the real-world situation whereby poachers have knowledge of past patterns (mixed strategies) of ranger deployment but not the exact location of ranger patrols when they set out to lay snares.

In our game, there were $M = 9$ rangers protecting this park, with each ranger protecting one grid cell. Therefore, at any point in time, only 9 out of the 25 distinct regions in the park are protected. The players know before they play (place a snare) that only 9 out of the 25 regions will be protected, but as mentioned earlier, they do not know which 9 beforehand. In other words, in a particular round, a player can only know about the presence or absence of a ranger at the location he attacks only after he attacks. A player succeeds if he places a snare in a region which is not protected by a ranger and hence captures a hippo, else he is unsuccessful.

We make two modeling choices in this game. First, we focus on situations where there is no collusion or coordination between poachers in placing of snares as is true in most cases in a large forest area; accordingly, a player in our game can only see his/her snare, but not that of other players and thus cannot coordinate with other players. Second, we assume in our experiments that the defender is able to observe all the attacks conducted by the poachers and hence learn the adversaries' preferences from the complete attack data set. However, later in this article we also provide analysis of defender strategies generated when this assumption is violated, i.e., when the defender can only observe fractions of the entire attack dataset.

4.2. Computation of poacher reward

A key factor in this game is determination of rewards for poachers and rangers. For poachers animal density is a key factor determining their rewards. In addition to animal density, which is strongly correlated with high-risk areas of poaching [61,60,38], distance is another important factor in poaching, e.g., recent snare-density studies have found that significant poaching happens within 5 kilometers of South Africa's Kruger National Park border [53] and significantly decreases more than 4 kilometers away from the international border in Ksavo West National Park in Kenya [79]. Therefore, the reward obtained by a poacher in successfully placing a snare at target i is calculated by discounting the animal density by a factor of the distance traveled and is calculated as follows:

$$R_i^a = int(\phi_i - \zeta * \frac{D_i}{\max_j(D_j)}) \tag{8}$$

Here, ϕ_i and D_i refer to, respectively, the animal density at target i and the distance to target i from the poacher's starting location. For simplicity, we consider the adversary's reward as an integer. So, $int(y)$ rounds the value y to the closest integer value. The parameter ζ is the importance given to the distance factor in the reward computation and may vary based on the domain. Intuitively, the reward for successfully placing a snare in a region i near the starting location and which has animal density ϕ_i, is higher than the reward obtained by successfully placing a snare in a region with the same animal density but which is at a greater distance from the starting location as compared to i. We used $\zeta = 2$ in our experiments because the use of $\zeta = 1$ did not introduce substantial impact of distance while computing the actual rewards and $\zeta = 3$ was not used to prevent the overwhelming impact distance had on the actual rewards computed.

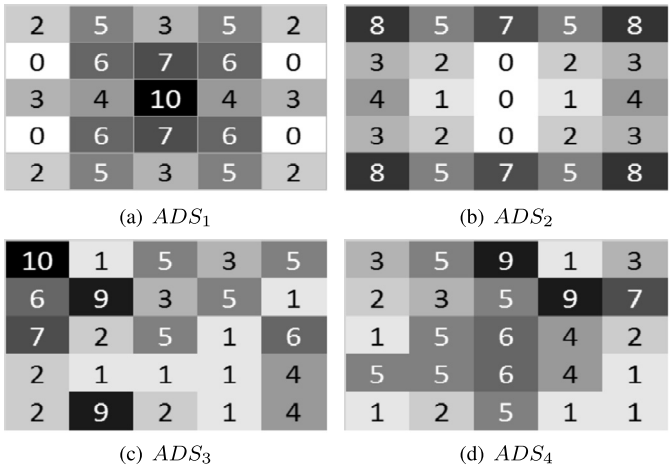
4.3. Non-zero sum game

In our games, the minimum and maximum animal density at each cell were 0 and 10 units respectively. The poacher received a flat penalty of -1 if he was caught at any target. We vary the adversary's actual reward based on the amount of distance traveled because he has to carry the captured animal back to the edge of the forest. However, there is no burden of carrying the animal back when the poacher is caught by the ranger (or equivalently his snare is confiscated), and therefore, in our games we do not vary the penalty based on distance and assume a constant value of -1. Also in our game, when the poacher successfully poaches, he may obtain a reward that is less than the animal density (Eqn. (8)), but the defender loses a value equal to that of the animal density, i.e., the game is non-zero-sum.[3]

5. Payoff structures

The payoff structures used in our human subject experiments vary in terms of the animal densities and hence the adversary rewards. We henceforth refer to payoff structures and animal density structures interchangeably in this article. The total number of animals in all the payoffs we generate is the same (=96). However, the variation in these payoffs is in the way the animals are spread out in the park. In payoff structure 1 (i.e., Animal Density structure 1 or ADS_1), the animal density is concentrated towards the center of the park, whereas the animal density is higher towards the edges of the park

[3] Note that in terms of real-world interpretation of the payoff to the adversary in this game, it is to be interpreted as taking into account the probability of catching a hippo. Therefore, higher density leads to a higher payoff. That is, it is the expected reward (in the absence of the defender) in attacking a particular cell – the expected number of hippos captured without the defender.



(a) ADS_1 (b) ADS_2

(c) ADS_3 (d) ADS_4

Fig. 3. Animal density structures (ADS).

in payoff structure 2. These represent scenarios that might happen in the real world. The animal density for both payoffs is symmetric, thus eliminating any bias due to the participant's starting point in the game.

Contrary to the above, animals in the park may be randomly scattered without any particular orientation. So, we randomly generate two additional animal density structures (payoffs 3 and 4) and test our proposed model on these payoffs. To generate a random structure, one out of 25 cells was chosen uniformly at random and then an animal was allocated to it until the total number of animals in all the cells was 96, keeping in mind that if a cell total reached 10 (maximum animal density in a cell), that cell was not chosen again. Figs. 3(a)–3(d) show heatmaps of four animal density structures, denoted as ADS_1, ADS_2, ADS_3 and ADS_4 respectively.

6. Online repeated measures experiments

Repeated measures studies are research studies which are typically conducted to observe and understand the changes in and effects of a particular set of variables over a period of time [59,32,42]. In our work the key variable is the adversary's strategy and we show that the adversary's strategy does indeed change over time due to his adaptive nature (as explained later in Section 8) and hence we model such behavior with a novel model called SHARP. Such studies can be conducted with a subject pool at a University lab or by recruiting participants in an online setting like AMT. We conducted our experiments on AMT. We tested a variation (Section 7) of the set of behavioral models introduced in Section 2 and our new model SHARP by deploying the mixed strategy generated based on each of these models repeatedly over a set of five rounds [49]. We observed the strategies employed by the participants in each round, i.e., where they attacked and whether they succeeded or failed, and used that to determine the optimal ranger strategy for the next round. For each model, we recruited a new set of participants to eliminate any learning bias.

We took necessary steps to ensure that participants completely remember the game details and the procedures to play the game during each round of the experiment, as otherwise we may lose significant time and effort in collecting poor quality data, especially because each setting would take more than two weeks to be completed. This was done by setting up proper validation and trial games in each round of the experiment, while not over-burdening the participants with many games and thus keeping their cognitive overload at a minimum. This is discussed next in Sec. 6.1, followed by a discussion of participant retention rates in our study in Sec. 6.2.

6.1. Validation and trial games

After viewing the instructions at the beginning, the participants were first asked to play two trial games in round 1, with an option to view the instructions again after each game. After the trial games, they played one validation game, and finally the actual game. The players could choose to stop playing at any point during this process. The validation game consisted of a cell with maximum animal density (=10) and the coverage probability of that cell was zero, while other cells had an animal density of 1 and non-zero but equal coverage probability. The participants were expected to select the target with the maximum animal density and zero coverage. Data from subjects who played the validation game incorrectly were discarded and they were not allowed to participate in future rounds of the experiment.

From second round onwards, participants were only asked to play one trial game and then the actual game. The trial game was kept in order to remind them of the game and its details and the playing procedures. Showing only the actual game without any trial games might have resulted in the participants not playing the game properly due to forgetfulness about the game details.

Table 1
Experiment details.

Average time taken per model per payoff structure (all 5 rounds)	Average time taken for a set of participants to play each round	Number of participants who played round 1 (minimum, maximum)	Average number of participants who played all 5 rounds	Average retention rate from round 2 to round 5
2.3 weeks	4 days	(42, 49)	38	83.69%

6.2. Participant retention rate

For our repeated measures experiments, due to unavailability of data, the strategy shown for each first round of the real game to the participants who passed the validation game was Maximin. We then learned the model parameters based on previous rounds' data, recomputed and redeployed strategies, and asked the *same* players to play again in the subsequent rounds. For each model, all five rounds were deployed over a span of weeks. When we started conducting the experiments, we observed that there were very high attrition rates (i.e. people dropped out) for the number of participants between rounds of the game. We observed that a delayed compensation scheme along with prior participant commitment and repeated reminders throughout the course of the experiment helped in achieving a high average retention rate of 83.69%. This is also shown in Table 1 along with other experiment details. For interested readers, a detailed discussion of the set of challenges that we faced during our experiments and our methodological contributions towards mitigating those challenges are presented in Appendix D.

7. SHARP: probability weighting

This article contributes a novel human behavior model called SHARP for BR-RSG settings. SHARP has three key novelties: (i) SHARP reasons based on success or failure of the adversary's past actions on exposed portions of the attack surface to model adversary adaptivity; (ii) SHARP reasons about similarity between exposed and unexposed areas of the attack surface, and also incorporates a discounting parameter to mitigate adversary's lack of exposure to enough of the attack surface; and (iii) SHARP integrates a non-linear probability weighting function to capture the adversary's true weighting of probability. In this section we cover the probability weighting aspect of SHARP and also discuss possible causes for the surprising results of incorporating probability weighting in our models. Other aspects are covered in Section 8.

7.1. Probability weighting mechanism

The need for probability weighting became apparent after our initial experiments. In particular, initially following up on the approach used in previous work [62,82,80,40], we applied MLE to learn the weights of the SUQR model based on data collected from our human subject experiments and found that the weights on coverage probability were positive for all the experiments. That is, counter-intuitively, humans were modeled as being attracted to cells with high coverage probability, even though they were *not* attacking targets with very high coverage but they were going after targets with moderate to very low coverage probability. Examples of the learned weights for SUQR from data collected from the first round deployment of the game for 48 human subjects on ADS_1 and ADS_2 are: $(\omega_1, \omega_2, \omega_3) = (2.876, -0.186, 0.3)$ and $(\omega_1, \omega_2, \omega_3) = (1.435, -0.21, 0.3)$. Here w_1 provides the SUQR code on coverage probability.

We prove Theorem 7.1 to show that, when the weight on the coverage probability in the SUQR model (ω_1) is found to be positive, the optimal defender strategy is a pure strategy. The proof of the theorem can be found in Appendix B.

Theorem 7.1. *When $\omega_1 > 0$, the optimal defender strategy is a pure strategy.*

Employing a pure strategy means that there will be no uncertainty about the defender's presence. Several cells will always be left unprotected and in those cells, the attackers will always succeed. In our example domains, even if the top-valued cells are covered by a pure strategy, we can show that such a strategy would lead to significantly worse defender expected utility than what results from the simplest of our defender mixed strategies deployed. For example, if cells of value 4 are left unprotected, the defender expected value will be -4, which is much lower than what we achieve even with a simple strategy like Maximin. In repeated SSG domains like wildlife crime, this would mean that the poachers successfully kill animals in each round without any uncertainty of capture by rangers. In order to show that playing a pure strategy does indeed lead to poor defender utility, we conducted an experiment with human subjects by deploying a SUQR based pure strategy on ADS_1. Results and comparisons with other models that are introduced later in the paper are shown in Section 12.1.

We hypothesize that this counter-intuitive result of a model with $\omega_1 > 0$ may be because the SUQR model may not be considering people's *actual* weighting of probability. SUQR assumes that people weigh probabilities of events in a linear fashion, while existing work on probability weighting (Section 3.2) suggest otherwise. To address this issue, we augment the Subjective Utility function (Eqn. (4)) with a two-parameter probability weighting function (Eqn. (9)) proposed by Gonzalez

and Wu [37], that can be either inverse S-shaped (concave near probability zero and convex near probability one) or S-shaped.

$$f(p) = \frac{\delta p^{\gamma}}{\delta p^{\gamma} + (1-p)^{\gamma}} \tag{9}$$

The SU of an adversary denoted by 'a' can then be computed as:

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 R_i^a + \omega_3 P_i^a \tag{10}$$

where $f(x_i)$ for coverage probability x_i is computed as per Eqn. (9). The two parameters δ and γ control the elevation and curvature of the function respectively. $\gamma < 1$ results in an inverse S-shaped curve while $\gamma > 1$ results in an S-shaped curve. We will henceforth refer to this as the PSU (Probability weighted Subjective Utility) function and the models (SUQR, Bayesian SUQR and Robust SUQR) augmented with PSU will be referred to as P-SUQR, P-BSUQR and P-RSUQR respectively. *Our SHARP model will also use PSU*. We will use these PSU-based models in our experiments. We did not explicitly deploy and compare models without probability weighting (for example, SUQR) against PSU-based models (for example, P-SUQR) because we have already shown in Theorem 7.1 that models without probability weighting may result in pure defender strategies being generated for subsequent rounds and would thus perform poorly in repeated SSG experiments.

One of our key findings, based on experiments with the PSU function is that the curve representing human weights for probability is *S-shaped in nature, and not inverse S-shaped* as prospect theory suggests. The S-shaped curve indicates that people would overweight high probabilities and underweight low to medium probabilities. Some learned curves will be shown in Section 12.2. Recent studies in economics [5,44,30] have also found S-shaped probability curves which contradict the inverse S-shaped observation of prospect theory. In addition, other recent work on security games, specifically Opportunistic Crime Security Games, has also found the existence of S-shaped probability weighting curves [2]. Furthermore, in previous literature [48,77] where they experimented with insurance and lotteries, they dealt with smaller number of alternatives (2 or 3 alternatives). In addition to the domain, one possible reason for observing S-shaped curves in our games could be that the participants are shown larger number of alternatives, i.e. they have to choose one from a set of 25 targets. To the best of our knowledge, participants' weighting of probabilities in such games with larger number of alternatives has not been studied before.

Given S-shaped probability weighting functions, the learned ω_1 was negative as it accurately captured the trend that a significantly higher number of people were attacking targets with low to medium coverage probabilities and *not* attacking high coverage targets.

Feature Selection and Weight Learning: In Section 4.1, we introduced a new feature – distance – that affected the reward and hence the obvious question for us was to investigate the effect of this new feature in predicting adversary behavior. We considered several variations of PSU with different combinations of features. Notice that each combination of the features could be used in each of our models, like P-SUQR, P-BSUQR, etc. In addition to Eqn. (10), three more are discussed below (Eqns. (11)–(13)). Recall that ϕ_i denotes the animal density at target i. Now, although it is true that the subjects were explicitly told the values of rewards and penalties at each cell and not the animal densities, the animal densities were still visually observable. ϕ has been used in Eqn. (11) to check if the participants may have been considering animal densities only and ignoring the effect of distance while playing the game, thus not paying attention to the effective reward. ϕ was used in Eqn. (13) to check if participants may have been considering animal density and distance as two separate features and weighting them in a linear fashion instead of the way we provided them the reward values. These models are designed and compared against each other to verify possible ways in which participants may actually have considered the features of the game while making decisions.

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 \phi_i + \omega_3 P_i^a \tag{11}$$

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 R_i^a + \omega_3 P_i^a + \omega_4 D_i \tag{12}$$

$$SU_i^a(x) = \omega_1 f(x_i) + \omega_2 \phi_i + \omega_3 P_i^a + \omega_4 D_i \tag{13}$$

To compare these variations, we need to learn the behavioral parameters for each of the variations (e.g., for Eqn. (13), a 6-tuple $b = <\delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4>$ is to be learned; δ and γ due to inclusion of Eqn. (9)) from available data and evaluate their effectiveness in predicting the attack probability. To learn the behavioral parameters b from available data, we propose an algorithm based on Repeated Random Sub-sampling Validation (Algorithm 2 – see Appendix A). For P-SUQR, we learn a single b, while for P-BSUQR and P-RSUQR we learn a set of $b \in \mathbb{B}$ for each attack. Note that, for our probability weighting function, we use all possible combinations of δ and γ, with values of each ranging from 0 to 4, at an interval of 0.1. Therefore, our analysis also contains $\delta = 1$ and $\gamma = 1$, which correspond to linear weighting of probabilities – the probability weights used in SUQR.

To test the performance of Algorithm 2 against a non-linear solver (Microsoft Excel's Generalized Reduced Gradient (GRG) nonlinear solver function) and also to compare between models with various feature sets, we learned the weights of the four behavioral models (Eqn. (10) to (13) in the article) using both Algorithm 2 and our non-linear solver. In order to do this, we deployed P-SUQR on ADS_1 and then collected participants' responses to the deployed strategy. Then, we performed the following steps, which conform to standard practices in machine learning for splitting data into training-validation-test sets [41,13]:

Table 2
Performance (Squared Errors) of various feature sets. Results accompanied by * imply significant differences (with two-tailed t-tests at confidence = 0.05) in performance of Algorithm 2 as compared to the non-linear solver. Results in **boldface** indicate significant differences in the performance of a particular feature combination with respect to other feature combinations.

	Eqn. (10)	Eqn. (11)	Eqn. (12)	Eqn. (13)
P-SUQR ADS_1 Algorithm 2	0.1965*	0.2031*	0.1985	**0.1025***
P-SUQR ADS_1 Non-linear Solver	0.2545	0.2589	0.2362	**0.1865**
P-SUQR ADS_2 Algorithm 2	0.2065*	0.2156*	0.2625	**0.1136***
P-SUQR ADS_2 Non-linear Solver	0.2546	0.2935	0.3062	**0.1945**

Table 3
Mean of the weights learned for the 10 training sets for the model in Eqn. (13) and all algorithm and payoff combinations in Table 2.

	Eqn. (13)
P-SUQR ADS_1 Algorithm 2	$< 2.36, 2.78, -2.3, 0.688, -0.3, -0.286 >$
P-SUQR ADS_1 Non-linear Solver	$< 3.88, 2.86, -4.3, 0.38, -0.3, -0.4 >$
P-SUQR ADS_2 Algorithm 2	$< 2.62, 2.92, -1.57, 0.38, -0.3, -0.34 >$
P-SUQR ADS_2 Non-linear Solver	$< 3.92, 3.02, -4.3, 0.34, -0.3, -0.28 >$

1. We divided the first round data for the experiment with P-SUQR on ADS_1 into 10 random train-test splits.
2. For each of the 10 training sets, we performed 10-fold cross-validation to obtain the best model weights that give the lowest validation error on the corresponding validation sets. That is, each of the 10 training sets was randomly partitioned into 10 equal sized subsamples. Of the 10 subsamples for each training set, a single subsample was retained as the validation data for validating the model, and the remaining 9 subsamples were used as training data. The cross-validation process was then repeated 10 times (the 10 folds), with each of the 10 subsamples used exactly once as the validation data, and the model weights that gave the lowest validation error (out of the 10 validation errors) was chosen after the cross validation process. Since we did this for each of the 10 training splits, we obtained 10 best learned model weights after applying 10-fold cross validation on each of the training splits.
3. Each of these 10 best learned model weights was then tested on the corresponding hold-out test data set by computing the sum of squared errors (SE) of predicting the attack probability over all the targets.
4. Finally, we computed the average of these SE values over the 10 test data sets. We computed this average SE for each of the four behavioral models using the best model weights learned by Algorithm 2 and the non-linear solver.

We report these average SE values for both the weight learning approaches on all the four behavioral models in Table 2. Results in **boldface** indicate significant differences (with two-tailed t-tests at confidence = 0.05) in the performance of Eqn. (13) as compared to all other feature combinations. We can see that Eqn. (13) achieves the lowest average SE as compared to all other feature combinations.

Our approach helps to significantly improve the robustness of our results. Note that, we used 10-fold cross-validation in our approach *not* to compute the average error over all the validation sets and using it to compare against other models (other feature combinations in our case), but instead to select the best parameters for a particular model, and using those model parameters to test on independent test data sets. We do this multiple times and report the average test set error, thus making the process of comparison between different behavioral models more robust. That is, instead of applying 10-fold cross validation once on one random train split of the original dataset, we performed 10-fold cross-validation on 10 separate training data sets randomly constructed from the original dataset. Cross-validation is in itself a well established model validation technique in machine learning and statistics to assess the generalizability of learned models on independent test data sets. The effectiveness of this approach to derive an accurate estimation of model prediction performance is well established in the machine learning literature [50,70,41,13]. Our approach of not just performing 10-fold cross validation once to select the best model weights, but multiple (10) times and then taking an average of the test set errors of the best learned model weights is also similar to what is traditionally adopted in machine learning literature [25] to improve robustness.

Our results show that we can achieve higher accuracy in modeling by generalizing the subjective utility form used in [62] that relied on just three parameters, by adding more features as shown in Eqn. (13). This opens the door to novel subjective utility functions for different domains that exploit different domain features. Results accompanied by * imply significant differences in performance of Algorithm 2 as compared to the non-linear solver. Thus, Algorithm 2 is more efficient in learning model weights as compared to the GRG non-linear solver.

We also present in Tables 3 and 4 the mean and standard deviations of the weights learned on the 10 training datasets for the best model (Eqn. (13)) on both the payoff structures ADS_1 and ADS_2 and for both the learning algorithms (Algorithm 2 and Non-linear Solver). Based on our detailed experiments, in addition to $\omega_1 < 0$, we found that $\omega_2 > 0$, $\omega_3 < 0$ and $\omega_4 < 0$. The rest of the formulations in this article will be based on these observations about the feature weights.

Table 4
Standard deviation of the weights learned for the 10 training sets for the model in Eqn. (13) and all algorithm and payoff combinations in Table 2.

	Eqn. (13)
P-SUQR ADS_1 Algorithm 2	$< 0.279, 0.4, 0.9, 0.41, 0, 0.18 >$
P-SUQR ADS_1 Non-linear Solver	$< 0.139, 0.32, 0.88, 0.09, 0, 0.07 >$
P-SUQR ADS_2 Algorithm 2	$< 0.19, 0.39, 0.69, 0.37, 0, 0.12 >$
P-SUQR ADS_2 Non-linear Solver	$< 0.1, 0.34, 0.88, 0.02, 0, 0.08 >$

7.2. Discussions

Section 7.1 provides an S-shaped probability weighting curve (learned curves are shown in Figs. 12(a) and 12(b) Section 12.2) as one explanation of the human players' behavior data. Given the surprising nature of this result, it is important to discuss other possible hypotheses that may explain why those human behaviors may have been observed. This section shows however that evidence does not support these alternatives to S-shaped probability weighting curve discussed earlier.

One potential hypothesis is that the participants may have misinterpreted aspects of the game interface design shown in Fig. 2. However, we took several steps to guard against such misinterpretations: (i) we asked the participants to play two trial games and one validation game in the first round and one trial game in each subsequent round; and (ii) we explained key facets of the game in the instructions and the participants could switch to the instructions after playing each of the trial and validation games to verify their understanding before they played the actual game. In addition to ensuring that the participants were given clear instructions and provided enough practice through trial games, we also checked the results of the validation game and it showed that 860 out of 1000 participants passed the validation game – indicating an understanding of the game. Note that we then discarded data from 140 out of 1000 participants (an average of 7 participants per group) who played the validation game incorrectly.

Another hypothesis could be that the validation game had introduced some misinterpretations. Specifically, in our validation game the participants had to choose between an option which is good on two scales (highest animal density of 10 and zero coverage) and other options which are bad on both scales (lowest animal density of 1 and non-zero but equal coverage of 0.375). Therefore, this could potentially have caused the participants to incorrectly interpret the scales in the actual games they played and hence they may have misinterpreted the coverage probabilities in the actual games. However, there is little support for this hypothesis as well. Note that the validation game is one of three games being played by each participant before the actual game in the first round. Also, the validation game is only played once in the first round and never played again in future rounds. However, the participants played two trial games in the first round and one trial game in the future rounds before playing the actual game in each round, and these trial games do not have the same "two scales" property as the validation game as discussed earlier.

Another possible hypothesis for such an S-shaped curve for the probability weighting function could potentially be that we use the weighted probabilities as a separate additive feature in our model – P-SUQR implies that we take a weighted sum of the different model features. This is contrary to how the probability weighting function is used in the prospect theory literature [48,77]. In that literature, the weighted probabilities are used to weight the values of outcomes; could that perhaps explain the S-shaped curve in our results? Unfortunately, evidence does not support this hypothesis as well. First, note that there have been existing works in the literature that show learning of S-shaped probability weighting curves even when conforming to the traditional prospect theoretic model, i.e., when the prospect theoretic values of outcomes are weighted by transformed probabilities [2,52]. Thus, there already exists evidence of S-shaped probability curves in other domains even for the traditional prospect theoretic function. Furthermore, to verify the shape of the probability weighting curve in our game setting when we consider values of outcomes to be weighted by the transformed probabilities, we explored an alternate form of our P-SUQR model, called PWV-SUQR (Probability Weighted Values SUQR). In PWV-SUQR, the rewards and penalties are weighted by the transformed coverage probabilities, as shown in Eqn. (14). In Section 12.2.2, we show that even while learning adversary behavior using Eqn. (14), we get S-shaped probability curves. This result indicates that the learned S-shape of the probability curves is not merely the outcome of the additive nature of our P-SUQR model.

$$SU_i^a(x) = \omega_1(1 - f(x_i))R_i^a + \omega_2 f(x_i)P_i^a \tag{14}$$

8. SHARP: adaptive utility model

A second major innovation in SHARP is the adaptive nature of the adversary and addressing the issue of attack surface exposure. First, we define key concepts, present evidence from our experiments, and then present SHARP's innovations.

Definition. The ***attack surface*** α is defined as the n-dimensional space of the features used to model adversary behavior. Formally, $\alpha = < F^1, F^2, \ldots, F^n >$ for features $F^j(\forall j; 1 \leq j \leq n)$.

For example, as per the PSU model in Eqn. (13), this would mean the space represented by the following four features: coverage probability, animal density, adversary penalty and distance from the starting location.

Definition. A ***target profile*** $\beta_k \in \alpha$ is defined as a point on the attack surface α and can be associated with a target. Formally, $\beta_k = < F_k^1, F_k^2, \ldots, F_k^n >$ denotes the kth target profile on the attack surface.

In our example domain, the kth target profile can be represented as $\beta_k = < x_{\beta_k}, \phi_{\beta_k}, P^a_{\beta_k}, D_{\beta_k} >$, where x_{β_k}, ϕ_{β_k}, $P^a_{\beta_k}$ and D_{β_k} denote values for coverage probability, animal density, attacker penalty and distance from starting location respectively.[4] For example, $< 0.25, 2, -1, 4 >$ is the target profile associated with the top-leftmost cell of ADS_1 in round 1. We can add more features like terrain information, vegetation, etc. if available. Exposing the adversary to a lot of different target profiles would therefore mean exposing the adversary to more of the attack surface and gathering valuable information about their behavior. While a particular target location, defined as a distinct cell in the 2-d space, can only be associated with one target profile in a particular round, more than one target may be associated with the same target profile in the same round. β_k^i denotes that target profile β_k is associated with target i in a particular round.

8.1. Observations and evidence

Below is an observation from our human subjects data, based on the above concepts, that reveal interesting trends in attacker behavior in repeated SSGs.

Observation 1. Consider two sets of adversaries: (i) those who have succeeded in attacking a target associated with a particular target profile in one round; and (ii) those who have failed in attacking a target associated with a particular target profile in the same round. In the subsequent round, the first set of adversaries are significantly more likely than the second set of adversaries to attack a target with a target profile which is 'similar' to the one they attacked in the earlier round.

In order to provide evidence in support of Observation 1, we show results from our data highlighting these trends on ADS_1 and ADS_2 in Figs. 4(a)–4(h). In each plot, the y-axis denotes the percentage of (i) attacks on similar targets out of the total successful attacks in the previous round (ζ_{ss}) and (ii) attacks on similar targets out of the total failed attacks in the previous round (ζ_{fs}). Here, by *similar*, we mean the k nearest neighbors to the target profile under consideration and these are determined by computing the Euclidean distances between the target profiles on the attack surface. In this case, we have set $k = 5$, i.e., 5 nearest neighbors. The x-axis denotes pairs of rounds for which we are computing the percentages, for example, in R23, 2 corresponds to round $(r-1)$ and 3 means round r in our claim. Thus, ζ_{ss} corresponding to R23 in ADS_2 is 80%, meaning that out of all the people who succeeded in round 2, 80% attacked similar target profiles in round 3. Similarly, ζ_{fs} corresponding to R23 in ADS_2 is 33.43%, meaning that out of all people who failed in round 2, 33.43% attacked similar target profiles in round 3.

From Figs. 4(a)–4(h), we can observe that, as opposed to the failed attackers, a statistically significant number of successful attackers returned to attack the same or similar targets in the subsequent round. The average (over all four models on two payoffs and for all round pairs) of ζ_{ss} is 75.2% and the average of ζ_{fs} which is 52.45%. This difference is statistically significant (two-tailed t-tests at confidence $= 0.05$), thus supporting Observation 1.

One might however argue that successful poachers return to attack the same or similar targets in future rounds due to some inherent bias towards specific targets and not because they succeeded on such targets in the previous rounds. Therefore, we conducted additional human subjects experiments to test the extent to which successes and failures alone affect their decision making process.

We recruited two groups of human subjects and conducted two rounds of repeated experiments with each group. We showed the Maximin strategy to both groups in both rounds of the experiment. We ensured that all the participants of Group 1 succeeded in round 1, i.e., even though there were coverage probabilities shown, no rangers were actually "deployed". In round 2, Maximin strategy was again deployed and the same set of players were asked to play. We observed that 96% of the human subjects attacked the same or similar ($k = 5$) target profiles. We observed that out of the 96%, 70.83% attacked the exact same target profile as they had attacked in round 1. Group 2 was shown Maximin strategy in round 1 and all the participants were made to fail in round 1, i.e., despite the coverage probabilities, there was a "ranger" deployed in every cell. In round 2, Maximin strategy was again deployed and the same set of players were asked to play. We observed that only 36% of the participants attacked the same or similar ($k = 5$) targets in round 2. This shows that successes and failures are important factors that players take into account while deciding on their strategy in subsequent rounds. Similarly, when $k = 6$, we observe that 38% of the participants from Group 2 who failed in round 1, had actually attacked the same or similar target profiles. In Fig. 5, we show for various values of k, the percentage of successful participants in round 1 who returned to attack the same or similar targets in round 2 and the percentage of failed participants in round 1 who returned to attack same or similar targets in round 2.

Notice that failure does not lead all attackers to abandon their target profile (and vice versa with successful attacker). This shows that attackers have some inherent weights for defender coverage, animal density, penalty and distance, as is captured by the PSU weight vectors, but they do adapt their strategies based on their past successes and failures. Therefore,

[4] In our experiments, $\phi_{\beta_i} > 0$, $P^a_{\beta_i} < 0$ and $D_{\beta_i} > 0$.

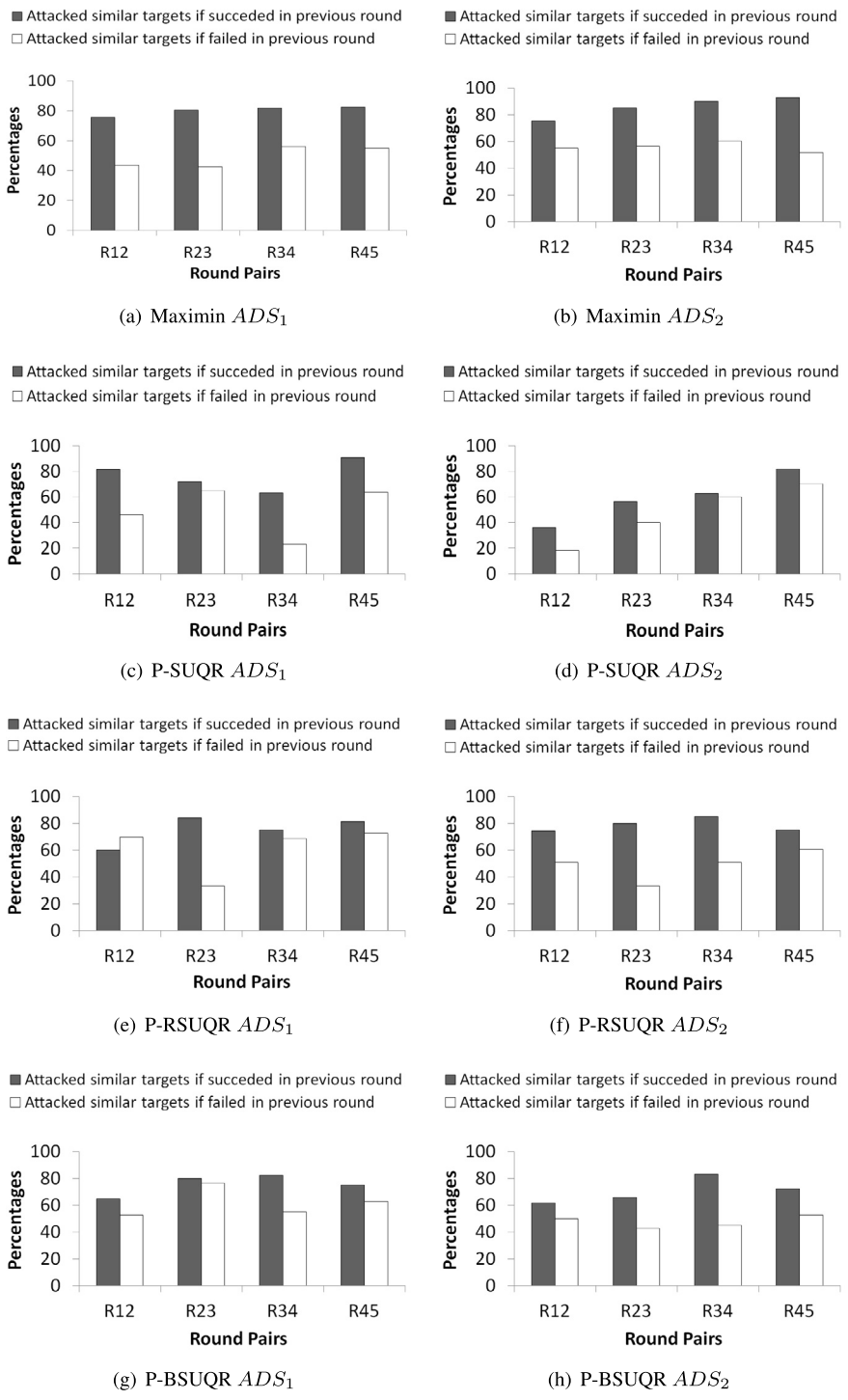


Fig. 4. Evidence for adaptivity of attackers.

we will observe later in Section 12 that even though P-SUQR is outperformed by our model SHARP in the initial rounds, P-SUQR is still a valuable model.

These observations about successes and failures on the adversary's future behavior are also consistent with the "spillover effect" in psychology [27], which in our case suggests that an adversary will tend to associate properties of unexposed target profiles with knowledge about similar target profiles to which he has been exposed, where similarity is expressed in terms of the Euclidean distance between two points on the attack surface. Smaller distance indicates higher similarity. The above

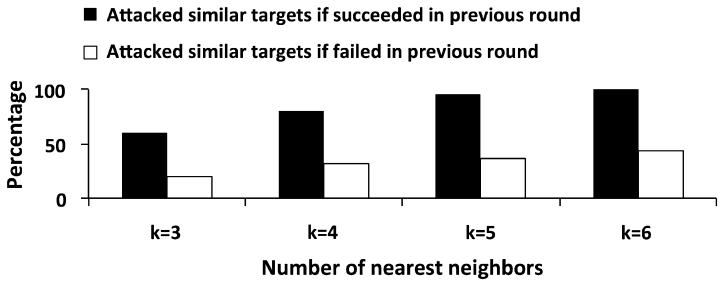


Fig. 5. For various values of k (the number of nearest neighbors), percentage of people who attacked similar targets in round 2 after succeeding or failing in the previous round.

aspects of adversary behavior currently remain unaccounted for, in BR-RSG models. Based on observations above, we define two key properties below to capture the consequences of past successes and failures on the adversary's behavior and reason based on them.

Definition. The ***vulnerability*** associated with a target profile β_i which was shown to the adversary in round r, denoted $V^r_{\beta_i}$, is defined as a function of the total number of successes and failures on the concerned target profile in that round (denoted by $success^r_{\beta_i}$ and $failure^r_{\beta_i}$ respectively). This is shown in Eqn. (15):

$$V^r_{\beta_i} = \frac{success^r_{\beta_i} - failure^r_{\beta_i}}{success^r_{\beta_i} + failure^r_{\beta_i}} \tag{15}$$

Therefore, more successful attacks and few failures on a target profile indicate that it was highly vulnerable in that round. Because multiple targets can be associated with the same target profile and the pure strategy generated based on the mixed strategy x in a particular round may result in a defender being present at some of these targets while not at others, there may be both successes and failures associated with the same target profile in that round.

Definition. The ***attractiveness*** of a target profile β_i at the end of round R, denoted $A^R_{\beta_i}$, is defined as a function of the vulnerabilities for β_i from round 1 to round R. It is computed using Eq. (16).

$$A^R_{\beta_i} = \frac{\sum_{r=1}^{R} V^r_{\beta_i}}{R} \tag{16}$$

Therefore, we model the attractiveness of a target profile as the average of the Vulnerabilities for that target profile over all the rounds till round R. This is consistent with the notion that a target profile which has led to more successful attacks over several rounds will be perceived as more attractive by the adversary.[5]

8.2. SHARP's utility computation

Existing models (such as SUQR, which is based on the subjective utility function (Eqn. (4))) only consider the adversary's actions from round $(r-1)$ to predict their actions in round r. However, based on our observation (Observation 1) it is clear that the adversary's actions in a particular round are dependent on his past successes and failures. The *adaptive* probability weighted subjective utility function proposed in Eq. (17) captures this adaptive nature of the adversary's behavior by capturing the shifting trends in attractiveness of various target profiles over rounds.

$$ASU^R_{\beta_i} = (1 - d * A^R_{\beta_i})\omega_1 f(x_{\beta_i}) + (1 + d * A^R_{\beta_i})\omega_2 \phi_{\beta_i} + (1 + d * A^R_{\beta_i})\omega_3 P^a_{\beta_i} + (1 - d * A^R_{\beta_i})\omega_4 D_{\beta_i} \tag{17}$$

There are three main parts to SHARP's computation: (i) Adapting the subjective utility based on past successes and failures on exposed parts of the attack surface; (ii) Discounting to handle situations where not enough attack surface has been exposed; and (iii) Reasoning about similarity of unexposed portions of the attack surface based on other exposed parts of the attack surface.

The intuition behind the adaptive portion of this model is that, the subjective utility of target profiles which are highly attractive to the adversary should be increased, and that of less attractive target profiles should be decreased, to model the

[5] Although here we give equal weight to the vulnerability values in each round, we can modify this easily to consider the recency effect in human decision making by discounting vulnerability values of earlier rounds and giving more importance to recent rounds. Such models of human discounting of past actions, such as hyperbolic discounting and exponential discounting, have been explored in [6,18,35]. Exploring such models in our formulation would be an interesting area for future work.

adversary's future decision making. Hence, for a highly attractive target profile β_i, the attacker would view the coverage x_{β_i} and distance from starting location D_{β_i} to be of much lower value, but the animal density ϕ_{β_i} to be of higher value, as compared to the actual values. The contribution of the penalty term would also increase the utility (recall that $P^a_{\beta_i} < 0$ and $\omega_3 < 0$).

Let us take an example from our game. Suppose we take the target profile $\beta_i =< 0.25, 2, -1, 4 >$. This profile had $A^1_{\beta_i} = 1$ after round 1, because $success^r_{\beta_i} = 9$ and $failure^r_{\beta_i} = 0$, i.e., the target was highly attractive to the attacker. The weights learned were $b =< \delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4 >=< 2.2, 2.4, -3, 0.9, -0.3, -0.5 >$, P-SUQR would compute the subjective utility as -0.29, while (assuming d (explained later) $= 0.25$, for example) SHARP's adaptive utility function would compute the subjective utility as 0.855. In comparison to the original subjective utility function, this function is adaptive due to the positive or negative boosting of model weights based on the defender's knowledge about the adversary's past experiences. Here, learning the model parameters b has been decoupled from boosting the model parameters for future prediction to ensure simplicity in terms of both the model formulation as well the weight learning process. This also ensures that the linearity in terms of the features of the model (as in the original SUQR model) remains intact. Through an example in Section 10, we show the effect of this design decision on the defender mixed strategy generated.

Now we turn to the next aspect of SHARP's utility computation. Recall the problem that the defender does not necessarily have information about the attacker's preferences for enough of the attack surface in the initial rounds. This is because, the attacker is exposed to only a limited set of target profiles in each round and the defender progressively gathers knowledge about the attacker's preferences for only those target profiles. We provide evidence in support of this observation in Section 12.3.

The parameter d ($0 \leq d \leq 1$) in Eqn. (17) mitigates this attack surface exposure problem. It is a discounting parameter which is based on a measure of the amount of attack surface exposed. d is low in the initial rounds when the defender does not have enough of the right kind of data, but would gradually increase as more information about the attacker's preferences about various regions of the attack surface become available. For our experiments, we varied d based on Eqn. (18):

$$d = \frac{1}{N_r - r} \tag{18}$$

where N_r is the total number of rounds and r is the round whose data is under consideration. For example, $N_r = 5$ and $r = 1$ for data collected in round 1 of an experiment conducted over five rounds. The intuition behind this formulation is that, as more rounds are played, more cumulative information about the adversary' preferences for a lot of the attack surface will be available and hence d will increase from a very small value gradually as rounds progress.

Finally, we look at how we reason about unexposed portions of the attack surface based on the exposed areas. If a target profile β_u was not exposed to attacker response in round r, the defender will not be able to compute the vulnerability $V^r_{\beta_u}$. Therefore, we will also not be able to estimate the attractiveness for β_u and hence the optimal defender strategy. So, in keeping with our analysis on available data and based on the spillover effect introduced earlier, we use the distance-weighted k-nearest neighbors algorithm [26] to obtain the Vulnerability $V^r_{\beta_u}$ of an unexposed target profile β_u in round r, based on the k most similar target profiles which were exposed to the attacker in round r (Eqns. (19) and (20)).

$$V^r_{\beta_u} = \frac{\sum_{i=1}^{k} \theta_i * V^r_{\beta_i}}{\sum_{i=1}^{k} \theta_i} \tag{19}$$

$$\theta_i \equiv \frac{1}{d(\beta_u, \beta_i)^2} \tag{20}$$

where, $d(\beta_u, \beta_i)$ denotes the Euclidean distance between β_u and β_i in the feature space. We use $k = 5$ for our experiments.

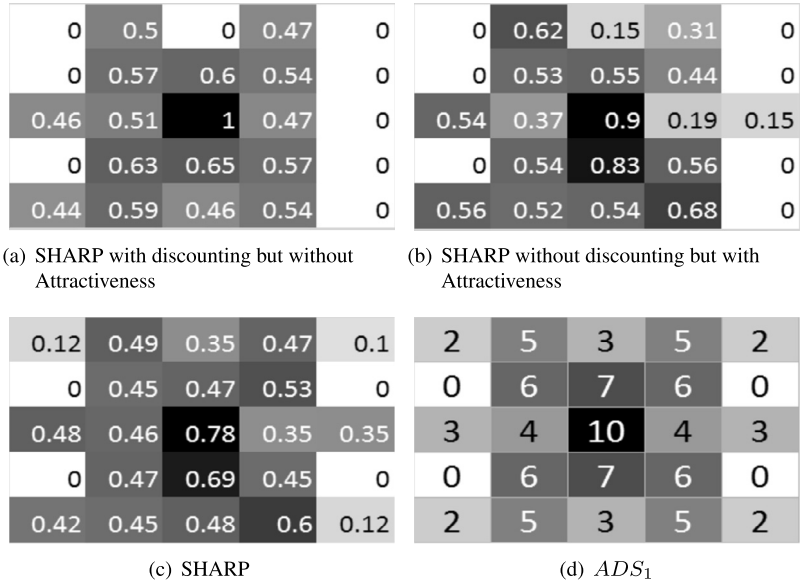
9. Generating defender strategies against SHARP

While SHARP provides an adversary model, we must now generate defender strategies against such a model. To that end, we first learn the parameters of SHARP from available data (see Section 7). We then generate future round strategies against the boundedly rational adversary characterized by the learned model parameters by solving the following optimization problem:

$$\max_{x \in \mathbb{X}} \left[\sum_{i \in \mathbb{T}} U^d_i(x)\, q^R_i(x|\omega) \right] \tag{21}$$

$q^R_i(\omega|x)$ is the probability that the adversary will attack target i in round R and is calculated based on the following equation:

$$q^R_i(\omega|x) = \frac{e^{ASU^R_{\beta^i_k}(x)}}{\sum_{i \in \mathbb{T}} e^{ASU^R_{\beta^i_k}(x)}} \tag{22}$$



(a) SHARP with discounting but without Attractiveness

(b) SHARP without discounting but with Attractiveness

(c) SHARP

(d) ADS_1

Fig. 6. (a, b, c): Round 2 strategies generated by SHARP (with discounting without Attractiveness), SHARP (no discounting but with Attractiveness) and SHARP respectively; (d) ADS_1.

β_k^i denotes that target profile β_k is associated with target i. $ASU_{\beta_k^i}^R$ and $U_i^d(x)$ are calculated according to Eqns. (17) and (1) respectively.

To solve the non-linear and non-convex optimization problem in Eqn. (21) and generate the optimal defender strategy, we use PASAQ [83] as it provides an efficient approximated computation of the defender strategy with near-optimal solution quality.

10. SHARP in action: an example

In this section we give an example to show the effectiveness of SHARP in terms of the design of each component: (i) adaptive utility, (ii) similarity learning, and (iii) confidence based discounting. Figs. 6(a), 6(b) and 6(c) show second round strategies generated by SHARP with discounting of Eqn. (18) but without Attractiveness, SHARP without discounting, i.e., $d = 1$ but with Attractiveness, and SHARP, based on parameters learned ($b = < \delta, \gamma, \omega_1, \omega_2, \omega_3, \omega_4 > = < 1.2, 1.6, -3.2791, 0.1952, -0.3, -0.8388 >$) from first round data collected for the experiment with SHARP on ADS_1 (shown in Fig. 6(d)). The strategy generated by SHARP with discounting but without Attractiveness (see Fig. 6(a)) can be easily exploited due to several unprotected cells with positive animal density. SHARP without discounting but with Attractiveness (see Fig. 6(b)) generates a comparatively more robust strategy than SHARP with discounting but without Attractiveness (Fig. 6(a)) due to its adaptive utility function and similarity learning mechanism. SHARP generates the best strategy (see Fig. 6(c)) due to its capability to model all the design parameters together into one single framework.

11. RL-SSG: a descriptive reinforcement learning algorithm for SSGs

In this section, we translate the basic Reinforcement Learning (RL) model proposed by Erev and Roth [28] to our setting; we use their RL approach to compute the optimal mixed strategy for the defender in repeated SSGs. The primary reason for adapting an existing RL based approach for our problem is to compare our models against another popular learning framework which has been used earlier in the context of two-player games. However, as explained in Section 3.4, the main challenge of using the same framework as in [28] is that the models in Erev and Roth's paper were developed for simultaneous move games without any notion of prior commitment to a mixed strategy by any player. Therefore, we developed a new RL model for our leader–follower setting where the defender moves first by playing a *mixed* strategy and the other player moves next by playing a pure strategy after having observed the first player's mixed strategy. Since we deploy and compute optimal mixed strategy responses for the defender per round based on all the attacks observed in the earlier rounds, we do not explicitly require a lot of defender and attacker pure strategy combinations to be deployed. We describe below the RL based algorithm (Algorithm 1) to compute the optimal mixed strategy.

The algorithm first starts with an initial propensity for the defender to play any pure strategy k (Line 1). The probability distribution over all possible pure strategies is then computed by normalizing the propensities (Line 2). The defender then computes the mixed strategy and deploys this strategy (which results in a coverage probability for each target as discussed in Section 2.1), and collects all pure strategy responses of the attacker to the defender's mixed strategy in the corresponding

Algorithm 1 Algorithm to learn RL based defender strategy for repeated Stackelberg games.

INPUT: Set of targets T; Number of security resources M.
OUTPUT: Optimal defender strategy for any round r.

1: In round $r = 1$, the defender has an initial propensity to play pure strategy k, denoted by $q_k(1)$; $\forall k \in P$, where P is the set of all possible pure strategies as described earlier in Sec. 2. (1) denotes round 1.
2: The defender computes the probability of playing a particular pure strategy k in round r based on the propensities as follows:

$$p_k(r) = \frac{q_k(r)}{\sum_{j \in P} q_j(r)} \tag{23}$$

The defender can then directly compute the mixed strategy $x(r)$ for round r from the $p_k(r)$'s, deploy $x(r)$ and collect attack data for round r.
3: The defender uses data collected in round r, i.e. D^r, to compute the expected utility of playing pure strategy k in round $r+1$ as follows:

$$U_k(r+1) = \frac{\sum_{i \in T} D_i(r) * B_{k,i}}{\sum_{i \in T} D_i(r)} \tag{24}$$

Here, $D_i(r)$ denotes the number of attacks on target i in round r, $B_{k,i}$ denotes the payoff to the defender if she plays pure strategy k and the adversary plays pure strategy i (i.e. attacks target i). It is calculated as follows:

$$\mathrm{B_{k,i}} = \begin{cases} \mathrm{R_i^d} & \text{if 'i' is protected in pure strategy 'k'} \\ \mathrm{P_i^d} & \text{if 'i' is not protected in pure strategy 'k'} \end{cases}$$

Here, R_i^d is the defender's reward for covering i if it is selected by the adversary and P_i^d is the penalty for not covering i.
4: The reinforcement of playing pure strategy k is then computed as:

$$I_k = U_k(r+1) - \min_k(U_k(r+1)) \tag{25}$$

Here, $\min_k(U_k(r+1))$ denotes the minimum utility over all the computed utilities.
5: The propensities of the defender for pure strategy $k(\forall k \in P)$ in round $(r+1)$ is then updated as follows:

$$q_k(r+1) = q_k(r) + I_k \tag{26}$$

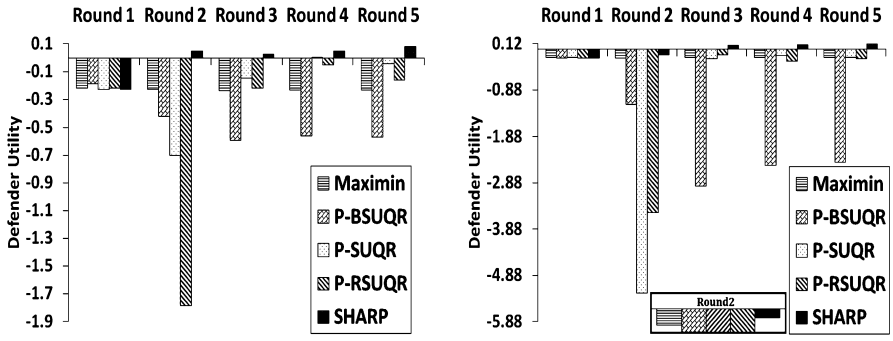
6: Repeat Step 2 to Step 5.

round (Line 2). Using this data collected in a particular round, the defender computes her utility of playing each pure strategy k (Line 3). More specifically, this utility is computed for each pure strategy as the reward that would result to the defender given the observed adversary response if the defender were playing only this pure strategy. The reinforcement of playing each pure strategy is then computed as the difference between the corresponding utility and the minimum possible utility over all the pure strategies (Line 4). The defender then updates her propensities of playing each pure strategy by adding the reinforcements to the propensities computed in the earlier round (Line 5). The mixed strategy to be deployed in the next round is then computed in the same way as Line 2, and the game proceeds to future rounds in this fashion.

In our game, starting from an equal initial propensity for each pure strategy (as proposed in [28]) would result in a mixed strategy with equal coverage probability at each target. Due to the differences in animal densities, if we start from a uniform mixed strategy, it would leave many of the targets of high animal density to be attacked in round 1 (as evidenced from other human subject experiments conducted in the past [66]) and that would result in a defender utility which is much lower than the cumulative utility for any of our models over five rounds (see Section 12 for details). Therefore, for our experiment with the RL approach, we assumed that the defender starts with the robust Maximin strategy in round 1. The use of Maximin as the initial strategy also ensures that we allow the RL model to start from the same starting point as all other models, and that it does not have an initial advantage or disadvantage compared to other models. Thus, we used Maximin to compute the initial propensities for pure strategies in this setup; we used Comb Sampling on the Maximin mixed strategy [76] to compute the probability of playing each pure strategy and considered those to be the initial propensities for each pure strategy. We then updated the propensities based on round 1 attack data for Maximin on ADS_1 and computed the corresponding mixed strategy and deployed that as the round 2 strategy. Based on this experiment, results and comparisons of the RL based approach against other models are shown in Section 12.1.

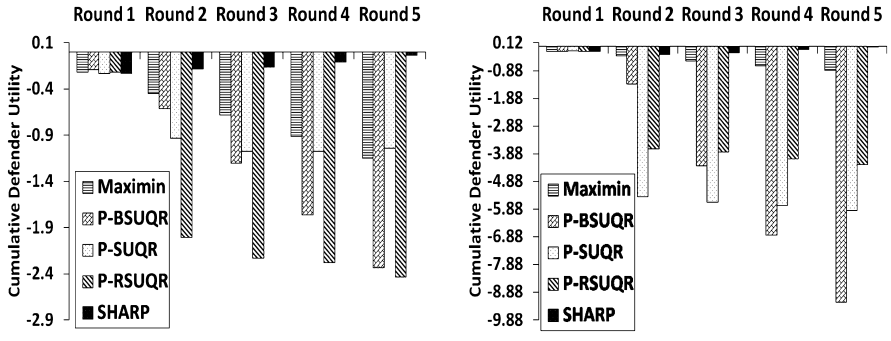
12. Results with human subjects on AMT

This section shows the results of our human subjects experiments on AMT, while the next section shows results against security experts, based on the experimental setting discussed in Sections 4, 5 and 6. In Section 12.1 we show average defender expected utilities for five models (P-SUQR, P-BSUQR, P-RSUQR, SHARP and Maximin) against actual human subjects, for various rounds of our experiment on two payoff structures (ADS_1 and ADS_2). In Section 12.2, we show results of learning the shape of the probability weighting function for the adversaries on ADS_1 and ADS_2 for three different scenarios: (i) when Gonzalez and Wu's probability weighting function Eqn. (9) is used in Eqn. (13); (ii) when Prelec's probability weighting (Eqn. (7)) is used in Eqn. (13); and, (iii) when transformed probabilities with Gonzalez and Wu's function is used to weight the values of outcomes (Eqn. (14)). In the same section, we also show prediction performances of models



(a) Results on ADS_1 (b) Results on ADS_2

Fig. 7. Defender utilities for various models on ADS_1 and ADS_2 respectively.



(a) Results on ADS_1 (b) Results on ADS_2

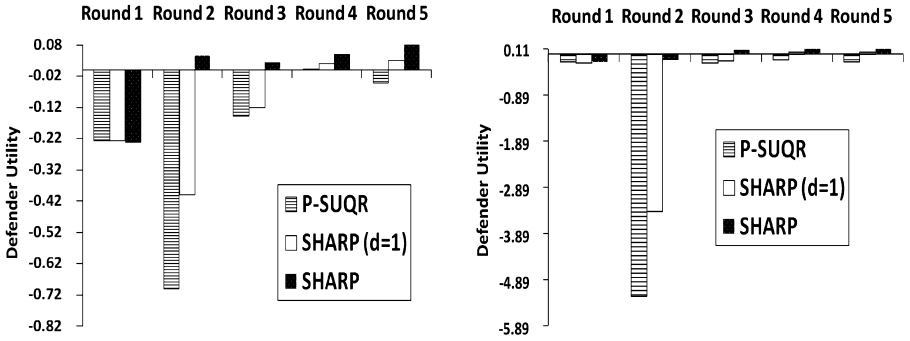
Fig. 8. Cumulative defender utilities for various models on ADS_1 and ADS_2 respectively.

for cases (ii) and (iii) above. Section 12.3 shows the effect of attack surface exposure on the performance of P-SUQR. In Section 12.4 we show the adaptiveness of SHARP based strategies over rounds while P-SUQR based strategies converge to a particular strategy at the end of a few rounds. Plots for all the analysis on payoff structures ADS_3 and ADS_4 are reported in Appendix F.

12.1. Defender utilities

In Figs. 7(a)–7(b) we show actual defender utilities obtained over 5 rounds for P-SUQR, P-BSUQR, P-RSUQR, SHARP and Maximin on ADS_1 and ADS_2 respectively, with an average of 38 human subjects playing per round. Similar to human subjects experiment results in previous work [67,81,62], in the plots, y-axis corresponds to defender expected utility. However, what is different here is that, we are now reporting results on repeated rounds and hence the round numbers are shown on the x-axis. For example, in Fig. 7(b), P-SUQR performs worst in round 2 with a utility of -5.26. In Fig. 7(b), we also show (inset) zoomed in results of the second round to highlight the difference in performance between Maximin ($=-0.18$) and SHARP ($=-0.1$). Figs. 8(a)–8(b) show cumulative defender utility over five rounds on ADS_1 and ADS_2 respectively. Note that the first round utilities for all models are same as Maximin strategy was played due to absence of data. All significance results reported below are computed with bootstrap t-test. Following are key observations from our experiments.

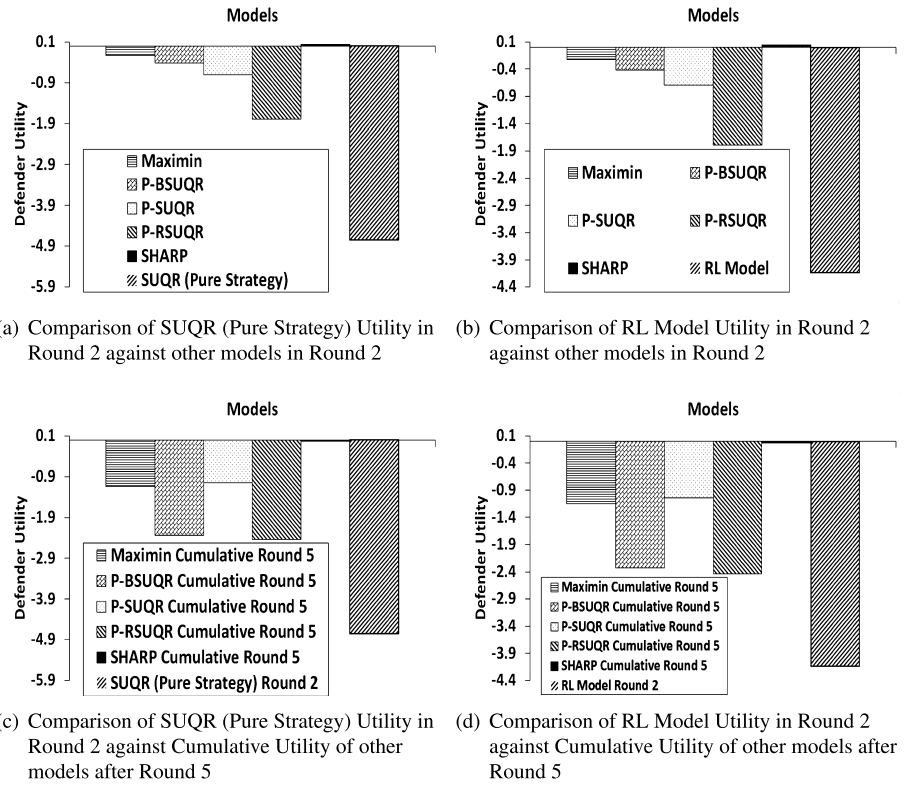
- **Heavy initial round losses**: For all models except SHARP, there is statistically significant ($p = 0.05$) loss in defender utility as compared to Maximin in second round on all the payoffs. P-SUQR recovers from initial round losses and outperforms Maximin in rounds 3, 4 and 5 for ADS_1 (statistically significant at $p = 0.05$), and in round 4 (statistically significant at $p = 0.15$) and round 5 for ADS_2. P-SUQR also outperforms Maximin in rounds 3, 4 and 5 on ADS_3 and ADS_4 (see Appendix F). P-RSUQR, which is a robust model, also outperforms Maximin in rounds 4 and 5 (statistically significant at $p = 0.05$) for ADS_1, ADS_3 and ADS_4 after initial round losses. Surprisingly, P-BSUQR, which is the basis for wildlife security application PAWS [80], performs worst on all payoffs over all rounds.
  From Fig. 8(a), we can observe that it takes five rounds for P-SUQR to recover from initial round losses and outperform Maximin in terms of cumulative defender utility for ADS_1. P-SUQR does not recover from initial round losses and outperform Maximin on any other payoffs. None of the other models recover from the initial round losses on any of the payoffs in five rounds, thus highlighting the impact initial round losses have on model performance for a long period of time.



(a) Results on ADS_1 (b) Results on ADS_2

Fig. 9. (a) Comparison of defender utilities between P-SUQR, SHARP and SHARP($d = 1$) on ADS_1 and ADS_2 respectively.

- **Performance of SHARP against other models**: SHARP consistently outperforms (statistically significant at $p = 0.05$) all the models over all rounds (Figs. 7(a)–7(b), and 27(a)–27(b)), most notably in initial rounds (round 2) and ends up with significantly high cumulative utility at the end of all rounds (Figs. 8(a)–8(b) and 28(a)–28(b)).
  Therefore, our results on extensive human subjects experiments on repeated SSGs show SHARP's ability to perform well throughout, including the important initial rounds.
- **Performance of SHARP (with and without discounting)**: To test the effectiveness of the design decisions in SHARP, we considered SHARP both with and without discounting. SHARP with $d = 1$ is compared against SHARP and P-SUQR on ADS_1 and ADS_2 in Figs. 9(a) and 9(b). SHARP($d = 1$) outperforms P-SUQR (statistically significant at $p = 0.05$) because it captures the adaptive nature of the adversary. However, it performs worse than SHARP (statistically significant at $p = 0.01$) as SHARP also trusts the data less when we don't have enough information about the adversary's responses to most of the attack surface; in this case the initial rounds.
- **Comparison with SUQR ($w_1 > 0$)**: As mentioned earlier in Section 7.1, we conduct additional human subjects experiments on ADS_1 to show that the performance of SUQR without probability weighting is worse than any of the other models. We deployed an experiment on AMT with the defender strategy computed based on the SUQR model learned from round 1 data of ADS_1. The resulting SUQR weight vector had a positive weight on coverage probability and thus resulted in a defender pure strategy. The game was deployed with this strategy on AMT. 60 people played the game, and out of them 48 participants passed the validation test. For our experimental results, we considered the data from only the participants who passed the validation test. The average expected defender utility obtained was -4.75. This average expected defender utility obtained by deploying a pure strategy based on a learned SUQR model is significantly less than that of all the other models on ADS_1 in Round 2 (Fig. 10(a)). Furthermore, the SUQR (Pure Strategy) model's average expected defender utility in this one round is significantly less than the cumulative average expected defender utility of all the other models after five rounds (Fig. 10(c)). Given that this strategy performs worse in one round than the cumulative average expected defender utility of all the other models, it demonstrates the point that the performance of SUQR without probability weighting is worse than any of the other models that include probability weighting.
  Notice that the reason this SUQR pure strategy performs so poorly is that it leaves 16 out of 25 targets completely exposed, and among these targets are ones with animal densities 4 and 5. Pure strategies for other reward structures similarly leave other targets of high value completely exposed. Also, as mentioned in Section 1, degraded performance in initial rounds may have severe consequences for the reasons outlined there. Thus, the poor performance in this initial round of the pure strategy on ADS_1 and its leaving targets of high value completely exposed illustrates that pure strategy SUQR is completely useless as a strategy for deployment. Therefore, we did not conduct any further experiments for future rounds with this model.
- **Comparison with RL based approach**: We conducted human subjects experiments on ADS_1 with the RL based approach (Algorithm 1) to compare its performance against our behavioral models. We deployed an experiment on AMT with the defender strategy computed based on the RL model learned from round 1 data of Maximin on ADS_1 (as explained earlier in Section 11). 63 participants played the game, out of which 49 participants passed the validation test. For our experimental results, we considered the data from only the participants who passed the validation test. The average expected defender utility obtained was -4.139. This average expected defender utility obtained by deploying the defender strategy based on a learned RL model is significantly less than that of all the other models on ADS_1 in Round 2 (Fig. 10(b)). Furthermore, the RL model's average expected defender utility in this one round is significantly less than the cumulative average expected defender utility of all the other models after five rounds (Fig. 10(d)). Given that this strategy performs worse in one round than the cumulative average expected defender utility of all the other models, it has very little chance of ever recovering and outperforming the other models we have discussed earlier after more rounds.
  In addition, we show the deployed defender strategy for round 2 on ADS_1 in Fig. 11. Based on Fig. 11, notice that the reason the RL based approach performs so poorly is that after learning from attacks in the previous round, it



(a) Comparison of SUQR (Pure Strategy) Utility in Round 2 against other models in Round 2

(b) Comparison of RL Model Utility in Round 2 against other models in Round 2

(c) Comparison of SUQR (Pure Strategy) Utility in Round 2 against Cumulative Utility of other models after Round 5

(d) Comparison of RL Model Utility in Round 2 against Cumulative Utility of other models after Round 5

Fig. 10. Comparison of various models with SUQR (Pure Strategy) and the RL model on ADS_1 respectively.

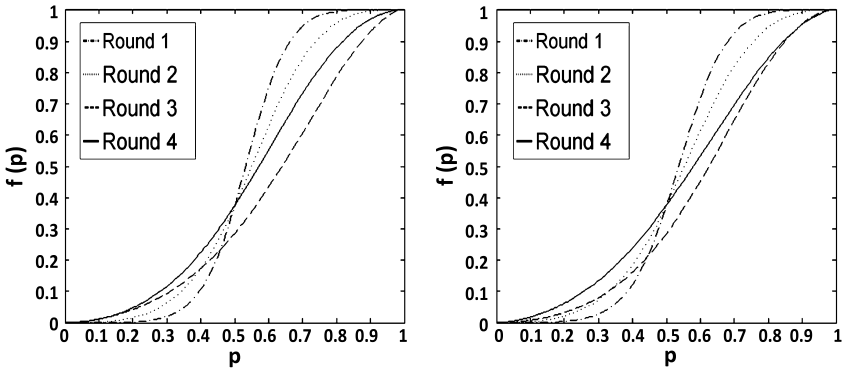
0.333	0.333	0.333	0.333	0.333
0.333	0.333	0.386	0.333	0.333
0.345	0.333	0.371	0.379	0.333
0.333	0.492	0.413	0.356	0.333
0.394	0.428	0.367	0.39	0.341

Fig. 11. RL model based defender strategy for round 2 on ADS_1.

places a significant amount of coverage on target cells with high number of attacks (the cell with a resultant coverage probability of 0.492 in round 2 had an animal density of 6 and was attacked 7 times in round 1), while it reduces the coverage on cells with very few attacks (the middlemost cell with a resultant round 2 coverage probability of 0.371 had an animal density of 10 and only 1 attack in round 1). This is because, for cells with zero or very few attacks, the propensities for playing strategies that correspond to protecting those targets are not updated as much as cells which have been attacked more frequently and have simultaneously resulted in higher gains for the defender. This leaves cells with high rewards but very few attacks in the past rounds less protected in the subsequent round and therefore completely exposed to a lot of attacks. RL based strategies for other reward structures similarly leave other targets of high value (but very attacks) with little protection and therefore open to exploitation by the attacker in the subsequent round. Thus, the poor performance in this initial round of the RL model on ADS_1 and its leaving targets of high value exposed to exploitation in the subsequent round illustrates that significant new work would need to be done to adapt the proposed RL framework (based on the model by Erev and Roth [28]) for our Stackelberg Security Game setting. As mentioned in Section 1, degraded performance in initial rounds may have severe consequences for the reasons outlined there. Therefore, given the poor initial round performance of the RL model on ADS_1, we did not conduct any further experiments for future rounds.

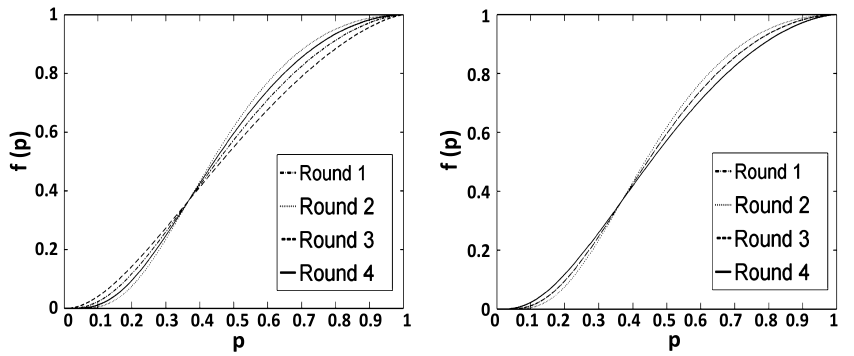
12.2. Learned probability curves

Figs. 12(a)–12(b) and 29(a)–29(b) show human perceptions of probability in rounds 1–4 when the participants were exposed to P-SUQR based strategies on ADS_1, ADS_2, ADS_3 and ADS_4 respectively. Learned curves from P-SUQR on all



(a) ADS_1 (b) ADS_2

Fig. 12. Learned probability curves for P-SUQR on ADS_1 and ADS_2 respectively.



(a) ADS_1 (b) ADS_2

Fig. 13. Learned probability curves with Prelec's probability weighting function for P-SUQR on ADS_1 and ADS_2 respectively.

payoffs have this S-shaped nature, showing that even though there is a little change in the curvature between rounds, it retains the same S-shape throughout all rounds. The curves indicate that people weigh high probabilities to be higher and low to medium probabilities to be lower than the actual values. Even though this is contrary to what Prospect theory [77] suggests, this is an intuitive result for our Stackelberg Security Games domain because we would expect the adversary to be deterred from targets with very high coverage probabilities and that they would prefer to attack targets with low to medium coverage probabilities.

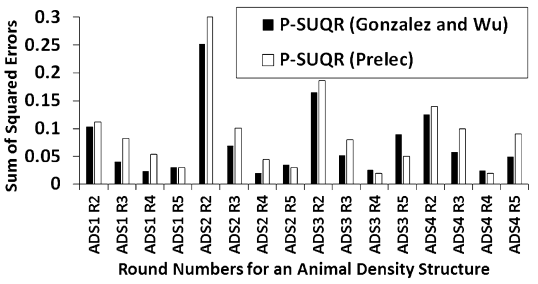
12.2.1. Comparison with Prelec's probability weighting function

As mentioned earlier in Section 3.2, we also experiment with Prelec's one-parameter model while allowing α to be any value greater than zero. In this case too, we learn S-shaped curves on all of our payoff structures as shown in Figs. 13(a)–13(b) and 30(a)–30(b). This indicates that the shape of the learned curve is not dependent on the probability weighting function used, as long as the function allows for learning both an S-shaped and an inverse S-shaped curve. In addition, the prediction performance (average of the sum of squared errors for all rounds and animal density structures) of P-SUQR with Gonzalez and Wu's probability weighting function (Eqn. (9) and Eqn. (13)) and P-SUQR with Prelec's probability weighting function (Eqn. (7) and Eqn. (13)) are 0.072 and 0.09 respectively and this is statistically significant at $p = 0.02$. The sum of squared errors in prediction for each of the four rounds (round 2 to 5) and each animal density structure are shown in Fig. 14(a), where the x-axis shows each possible combination of animal density structures and rounds, and the y-axis shows the sum of squared errors.

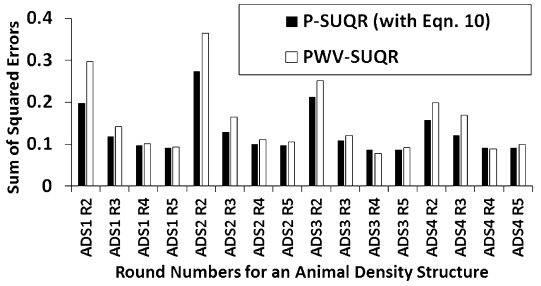
12.2.2. Comparison with PWV-SUQR

As mentioned earlier in Section 7.2, the adversary behavior model PWV-SUQR is one plausible alternative that could be considered for comparison with our models. Therefore, in this section, we first show the probability weighting curves learned (Figs. 15(a)–15(b) and 31(a)–31(b)) when we consider Eqn. (14) (see Section 7.2) as the subjective utility function in our adversary model. We observe that the curves are S-shaped in nature and this indicates that the shape of the probability weighting curves in our domain is not dependent on our use of the P-SUQR model.[6]

[6] Note that, instead of Eqn. (14), even if we use prospects where the transformed probabilities weight the transformed values [48,77], we still get S-shaped curves in our game setting.

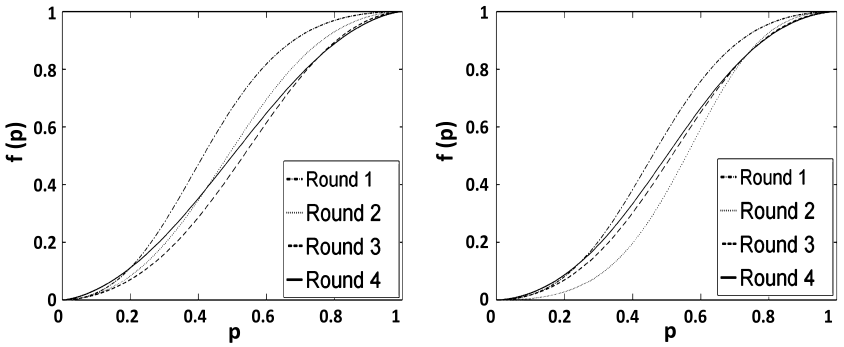


(a) Gonzalez and Wu vs Prelec



(b) P-SUQR vs PWV-SUQR

Fig. 14. (a) Comparison of the sum of squared errors for P-SUQR with Gonzalez and Wu, and P-SUQR with Prelec's probability weighting function respectively; (b) Comparison of the sum of squared errors for P-SUQR and PWV-SUQR respectively.



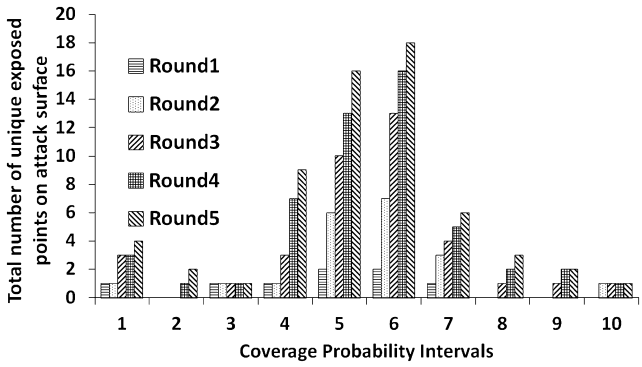
(a) ADS_1 (b) ADS_2

Fig. 15. (a)–(d) Learned probability curves for PWV-SUQR on ADS_1 and ADS_2 respectively.

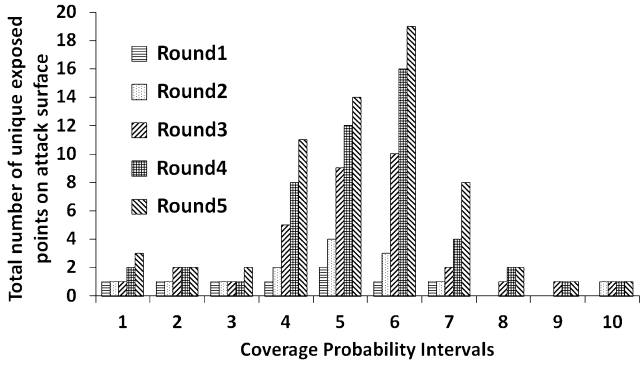
Nonetheless, PWV-SUQR does raise an intriguing possibility as a plausible alternative for P-SUQR and thus the performance of PWV-SUQR should be compared with P-SUQR. Therefore, we compare the performance of P-SUQR (with the PSU function in Eqn. (10)) and PWV-SUQR in terms of predicting future round attacks. We show that P-SUQR (with the PSU function in Eqn. (10)) performs better (with statistical significance) as compared to PWV-SUQR. The sum of squared errors in prediction for each of the four rounds (round 2–5) and each animal density structure are shown in Fig. 14(b), where the x-axis shows each possible combination of animal density structures and rounds, and the y-axis shows the sum of squared errors. The prediction performance (average of the sum of squared errors for all rounds and animal density structures) of P-SUQR (with the PSU function in Eqn. (10)) and PWV-SUQR are 0.128 and 0.155 respectively and this is statistically significant at $p = 0.01$. This justifies the use of P-SUQR and its variants while modeling the adversary.

12.3. Attack surface exposure

In our repeated SSG, the only variation in terms of feature values for our model (Eqn. (17)) from round to round is the mixed strategy x and hence the coverage x_i at each target. Hence, exposure to various regions of the attack surface means exposure to various values of x_i for fixed values of the other model parameters. Fig. 16(a)–16(b) and Fig. 32(a)–32(b) show how the adversary was exposed to more unique values of the coverage probability, and hence attack surface, when conducting experiments with P-SUQR over the five rounds for ADS_1, ADS_2, ADS_3 and ADS_4 respectively. We discretize the range of x_i, i.e. [0,1] into 10 intervals (x-axis) and show the total number of unique coverages exposed till a particular



(a) ADS_1



(b) ADS_2

Fig. 16. Total number of unique exposed target profiles till the end of each round for each coverage probability interval for ADS_1 and ADS_2.

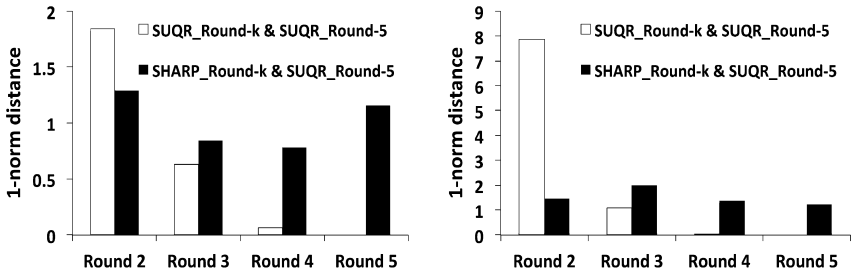
round (y-axis) for each interval. Observe that more interval ranges and more unique coverage probabilities get exposed in rounds 3–5, thus exposing more of the attack surface. Based on our earlier discussion in Section 8, this phenomenon of revealing more of the attack surface would lead to improved gain in information about the adversary and would thus help us to perform better in the future rounds. As we showed in Figs. 7(a)–7(b) and 27(a)–27(b), the defender performance for P-SUQR improves significantly in rounds 4 and 5.

12.4. Adaptiveness of SHARP

Recall that P-SUQR assumes the presence of a homogeneous adversary type and attempts to learn that adversary type from past attack data. So we should expect that as we learn the model parameters over various rounds, the learned parameters and hence the generated defender strategy should converge. On the contrary, SHARP models the adaptive nature of a homogeneous adversary type based on his past successes and failures. Hence the convergence of the defender strategy generated based on SHARP in each round is not guaranteed. Figs. 17(a)–17(b) and 33(a)–33(b) show the 1-norm distance between defender strategies generated by SHARP (and P-SUQR) over rounds with respect to the strategy generated by P-SUQR in round 5. While P-SUQR converges to a particular strategy in round 5 for all four animal density structures, SHARP does not converge to any strategy. To further illustrate that the SHARP based strategy does indeed change over rounds, we show SHARP based strategies on ADS_2 from rounds 2–5 in Figs. 18(a)–18(d). For ADS_2, the 1-norm distances between the defender strategies in rounds 2 and 3, rounds 3 and 4, and rounds 4 and 5 are 2.324, 2.19 and 1.432 respectively, showing that the strategies are changing from round to round. All these results demonstrate the adaptivity of SHARP over rounds based on the successes and failures of the adversaries in the past.

13. Validation and testing robustness of AMT findings

While in general findings from AMT have been validated with human subject experiments in the lab, the first question we ask is whether domain experts would perform similarly to what was observed of human subjects in AMT studies, i.e., we wish to further validate the findings from AMT. To that end, we deploy SHARP-based strategies against security experts at a national park in Indonesia and analyze the results (Section 13.1) by comparing them with our observations on human subjects data from AMT. A second question that may be raised is with regard to our assumption that all attack data is



(a) ADS_1 (b) ADS_2

Fig. 17. Adaptivity of SHARP and Convergence of P-SUQR on payoff structures ADS_1 and ADS_2 respectively.

0.6	0.56	0.55	0.45	0.58
0.35	0.32	0	0.18	0.3
0.36	0.14	0	0	0.5
0.32	0.36	0	0.32	0.32
0.69	0.43	0.58	0.48	0.6

(a) Round 2

0.52	0.43	0.45	0.34	0.4
0.35	0.33	0	0.1	0.1
0.5	0.1	0	0.1	0.35
0.52	0.42	0	0.33	0.37
1	0.67	0.59	0.53	0.52

(b) Round 3

0.64	0.52	0.6	0.5	0.55
0.42	0.38	0	0.1	0.1
0.51	0.19	0	0.1	0.4
0.4	0.1	0	0.38	0.27
0.64	0.53	0.64	0.54	0.64

(c) Round 4

0.62	0.51	0.59	0.5	0.61
0.42	0.24	0	0.1	0.36
0.48	0.19	0	0.1	0.46
0.44	0.37	0	0.06	0.39
0.67	0.53	0.6	0.52	0.62

(d) Round 5

Fig. 18. SHARP based strategy for the defender on payoff structure ADS_2.

perfectly observed in AMT studies. Therefore, we analyze SHARP-based strategies with only a fraction of the entire data (Section 13.2).

13.1. Results with security experts in Indonesia

In order to validate our AMT findings, we also conducted human subjects experiments for SHARP in the real world: with wildlife security experts from the provinces of Lampung and Riau, Sumatra, Indonesia. The 33 participants were from the local government, and from the following NGOs YABI, WWF and WCS. Each of the 33 participants played SHARP based strategies over 4 rounds. As in our AMT experiments, the first round strategy was Maximin.

In Fig. 19 we show actual defender utilities obtained over 4 rounds for SHARP on ADS_3. Interestingly, the defender utility obtained in round 2 was not only significantly higher than other rounds, but is also significantly higher than the utility obtained in round 2 for the same animal density structure for AMT participants (see Fig. 27(a)). This is because 96% of the experts who were successful in round 1 had attacked the same or similar targets in round 2. This is comparatively higher than the number of successful participants on AMT on ADS_3 in round 1 who returned to attack the same or similar targets in round 2: it was 78%. Hence, our model SHARP which captures the adversary's adaptiveness based on their past successes and failures, completely outperforms the experts. The defender's utility drops in round 3 as compared to that in round 2, because the experts, now aware of SHARP's adaptiveness, adjust their strategy. However, SHARP is robust enough to still generate high utility for the defender.

Similarity between AMT and Indonesia experts data: We earlier conducted a set of analyses and made certain observations based on our human subjects experiments data from AMT. We conducted the same analysis on the attack data obtained from real-world experts to validate our AMT results.

First, in our human subjects experiments on AMT we made Observation 1. We conducted analysis on the security experts data to see if we observe the same phenomenon in this data. Fig. 20 shows how the adversaries (security experts in this case) adapted to SHARP based strategy depending on past successes and failures. The x-axis denotes pairs of rounds for which we are computing the percentages; for example, in R23, 2 corresponds to round $(r-1)$ and 3 means round r in our claim. The results obtained are consistent with the ones obtained from our AMT data (see Figs. 4(a)–4(h)), i.e., successful adversaries tend to return to attack the same or similar targets in the subsequent round while failed adversaries will not tend to return to attack the same or similar targets in the subsequent round.

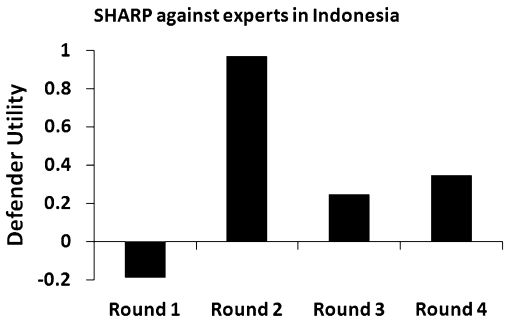


Fig. 19. Defender utility for SHARP against security experts in Indonesia.

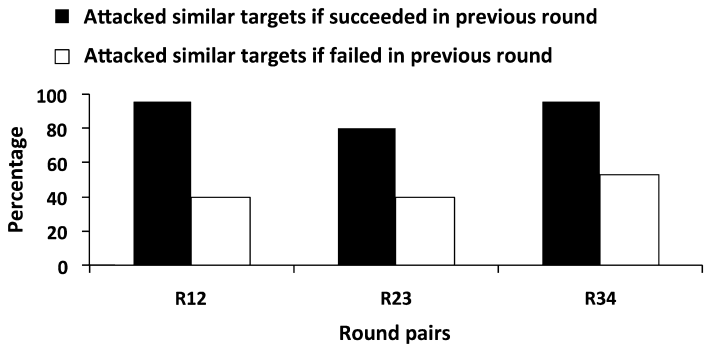


Fig. 20. Evidence for adaptivity of attackers (security experts in Indonesia).

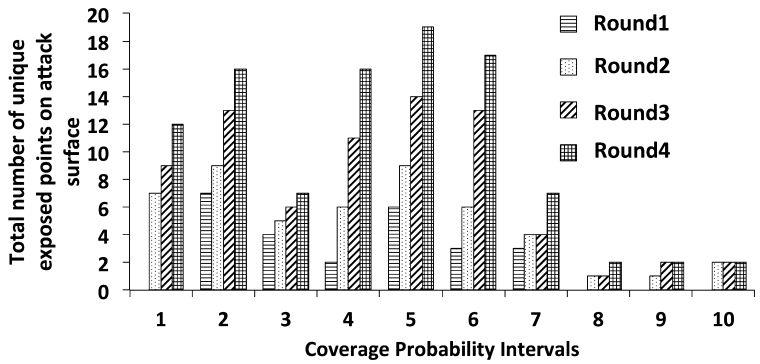


Fig. 21. Total number of unique exposed target profiles till the end of each round for each coverage probability interval for Indonesia experts data.

Second, we conducted analysis to see how the attack surface is exposed to the adversary over various rounds. The amount of attack surface exposed to the adversary over the four rounds in the wildlife experts data is shown in Fig. 21. This is consistent with similar plots obtained from our AMT data (see Fig. 16(a)–16(b) and Fig. 32(a)–32(b)) which show that as rounds progress, more number of coverage probability values from various intervals are exposed to the adversary.

Third, we show in Fig. 22, the human perceptions of probability in rounds 1–4 when the security experts were exposed to SHARP based strategies on ADS_3. The learned curves have an S-shaped nature for each of the rounds, which is consistent with our AMT findings (Section 12.2).

13.2. Results with fraction of complete attack data

In our human subjects experiments, we assume that the defender can observe all the attacks that occurred in each target region of the park at the end of every round. However, this may not be true in reality, i.e., defenders may miss some large fraction of the attacks. Therefore, we conduct additional analysis to understand the effects of considering a fraction of the original dataset on our defender strategy.

We generated round 2 defender strategies for all four payoffs with 50% of the data sampled randomly to test the robustness of our model. Here, by *robustness* we mean that the deviation of the strategy generated will be very similar to the original one, i.e., the 1-norm distance of the strategy generated with a fraction of the data will be very small when com-

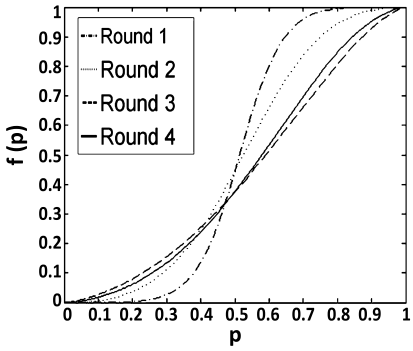
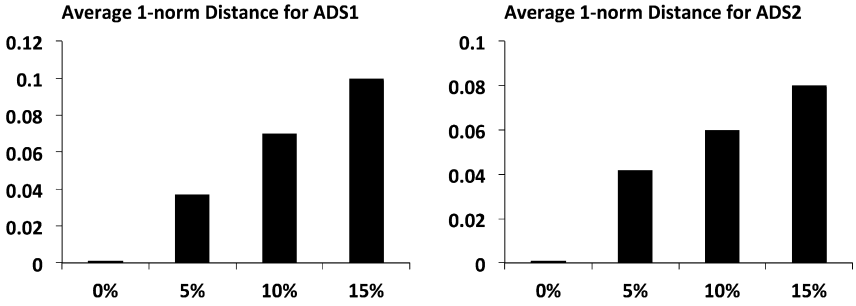


Fig. 22. Learned probability curves for SHARP on ADS_3 on the security experts dataset.



(a) Average 1-norm distance for ADS1 (b) Average 1-norm distance for ADS2

Fig. 23. (a) and (b): Average 1-norm distances between defender strategies generated by SHARP when the model is learned based on randomly sampled 50% data (0%, 5%, 10% and 15% deviation from actual data) and when the model is learned from the complete data set. Results are shown for ADS_1 and ADS_2 respectively.

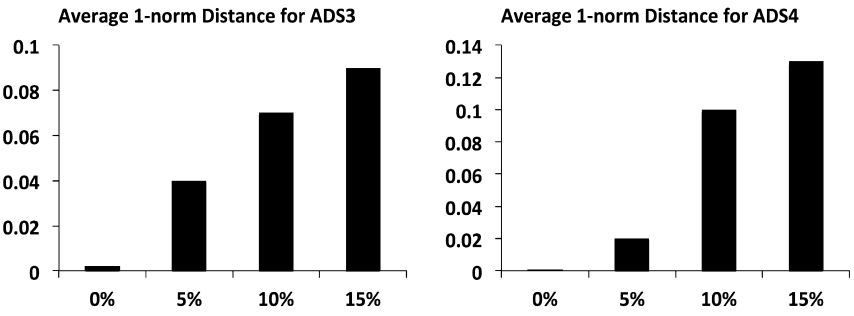
pared with the strategy generated with the full dataset. We randomly sampled several such fractional datasets but show results for four different sampled datasets (0%, 5%, 10% and 15% deviations from original attack data) for each payoff for the fraction size of 50%. By random sampling, we mean that, if there were $|\chi|$ attacks in the original dataset, we randomly picked a target cell and removed one attack data point and repeated this until 50% of the attack data (i.e. $round(|\chi|/2)$ attack data points) remained. Therefore, by 0% deviation, we mean that we removed 50% attacks from each target cell to make the new dataset. Similarly, by 5% deviation, we mean that the 1-norm distance between the new dataset obtained by removing 50% of the attack data and the original dataset is 0.05, and so on.

For each payoff structure we show (Figs. 23(a) and 24(b)) the average 1-norm distances between the round 2 defender strategies generated when datasets with various deviations (0%, 5%, 10% and 15%) from the original dataset were used to learn the model parameters, as opposed to learning from the complete dataset. We can observe from Figs. 23(a)–24(b) that the average 1-norm distance between the coverage probability x_i; $0 \leq x_i \leq 1$ for any target i between the original and 5% deviation datasets is no more than 0.044 for any of the payoffs. However, when the deviation from the original dataset increases to 15%, the average 1-norm distance also increases. Note that if the proportion of attacks over the targets were same as that of the original dataset, then the defender strategy generated would also be exactly the same modulo rounding errors.

14. Conclusions and future work

Although several competing human behavior models have been proposed to model and protect against boundedly rational adversaries in repeated Stackelberg security games, no study has yet been conducted against actual human subjects to show which is the best model in such repeated settings. This article provides three major contributions towards answering that question and therefore, provides an advancement to the field of "security games", a key area of research in Artificial Intelligence. Given the important applications of security games in areas such as protecting wildlife and fisheries, where there are repeated interactions between the defenders and adversaries, our contributions are critical for such domains.

First, we introduce a novel human behavior model called SHARP for repeated SSG settings. SHARP has three major novelties: (i) It models the adversary's adaptive decision making process by reasoning based on success or failure of the adversary's past actions on exposed portions of the attack surface. (ii) It accounts for lack of information about the adversary's preferences due to insufficient exposure to attack surface by reasoning about similarity between exposed and unexposed areas of the attack surface, and also incorporating a confidence based discounting parameter to model the learner's trust



(a) Average 1-norm distance for ADS3 (b) Average 1-norm distance for ADS4

Fig. 24. (a) and (b): Average 1-norm distances between defender strategies generated by SHARP when the model is learned based on randomly sampled 50% data (0%, 5%, 10% and 15% deviation from actual data) and when the model is learned from the complete data set. Results are shown for ADS_3 and ADS_4 respectively.

in the available data. (iii) It integrates a non-linear probability weighting function to model the adversary's perception of probabilities.

Second, we provide empirically supported methodological contributions towards conducting human subjects experiments in repeated measures settings on AMT. We provide analysis of the contribution of our approaches that help maintain high participant retention rates throughout the course of 2.3 week-long studies (46 weeks in total) and thus providing the grounds for fair comparison of various human behavior models in repeated Stackelberg Security Games.

Third, we provided results from the first repeated measures study of competing models in repeated SSGs to test the performance of SHARP along with existing approaches. We conducted experiments on four different payoff structures on the Amazon Mechanical Turk platform and also on one payoff structure in the real world, at the Bukit Barisan Seletan National Park in Indonesia, with security experts from the local government as well as various NGOs (WWF, WCS, YABI, etc.) who are in charge of protecting wildlife in the national park. Our results show that: (i) Human perceptions of probability are S-shaped, contradicting the inverse S-shaped observation from prospect theory. (ii) Existing human behavior models and algorithms perform poorly in initial rounds of repeated SSGs, with P-BSUQR which was earlier proposed as the basis for wildlife security application PAWS, performing poorly throughout all the rounds; (iii) a simpler model, P-SUQR, which was originally proposed for single-shot games recovers significantly after initial round losses; and (iv) SHARP performs significantly better than existing approaches consistently over all the rounds, most notably in the initial rounds.

Whereas in our work we have found support for S-shaped probability weighting curves, there are existing works that demonstrate inverse S-shaped curves in their domains [48,77]. As mentioned earlier in Section 7.1, one hypothesis is that human probability weighting could be context dependent, for example, results may vary depending on the number of alternatives available to the decision maker in the domain under consideration. Therefore, in the future, we would like to conduct further studies to identify settings when people may exhibit an S-shaped probability weighting curve as opposed to an inverse S-shaped one. We would also want to explore existing models of human discounting as we mentioned earlier in Section 8.1. Furthermore, while we have treated the reward to the poacher as a fixed reward (see Section 4.3, footnote 3) if the ranger does not capture the poacher, in reality, the poacher has to further take into account the uncertainty over capturing a hippo. This is a further layer of uncertainty that may need to be investigated in the future.

Acknowledgements

This research was supported by MURI Grant W911NF-11-1-03. We would also like to thank World Wildlife Fund (WWF) for their assistance in conducting real-world human subjects experiments with security experts in Indonesia.

Appendix A. Algorithm to learn PSU model parameters

The algorithm first randomly divides all data from past rounds into training and testing data (Line 2), and then splits the training data into K training and validation data sets (Line 3). Considering a set of $\{\delta, \gamma\}$ combinations, we learn the other parameters of the model and compute the average validation error over all the validation datasets for each $\{\delta, \gamma\}$ combination (Line 14). We then choose the $\{\delta, \gamma\}$ combination with the minimum average validation error (Line 16) and using that combination we re-learn the other model parameters and that forms our final weight vector (Line 17).

Appendix B. Proof of Theorem 7.1

Proof. Assume $R_i^d(>0)$, $P_i^d(<0)$, $R_i^a(>0)$ and $P_i^a(<0)$ Also assume that the defender has $M \in \mathbb{N}^+$ defender resources. Let q_i be the attacking probability for target i. According to SUQR model,

Algorithm 2 Algorithm to learn the weights of P-SUQR and its variations in repeated Stackelberg games.

INPUT: Data from R rounds: $D^1, D^2, ..., D^R$.
OUTPUT: Learned weights (δ_p, γ_p, ω_1, ω_2, ω_3, ω_4).

1: **for** $r = 1$ to R **do**
2: Randomly divide the collected data D^r into 1 training (Tr^r) and 1 test (Te^r) set.
3: Take the training samples (Tr^r) and randomly divide it into K training (${}^kTrv^r$) and validation (${}^kValr$) splits ($1 \le k \le K$).
4: **end for**
5: Consider a range of values for both δ and γ (Eqn. (9) in the article).
6: Discretize each range and consider all possible $\{\delta, \gamma\}$ pairs in that range. Let there be M such pairs.
7: **for** $i = 1$ to M **do**
8: **for** $k = 1$ to K **do**
9: Given training splits ${}^kTrv^1, {}^kTrv^2, ..., {}^kTrv^R$, learn the weights ${}^k\omega = ({}^k\omega_1, {}^k\omega_2, {}^k\omega_3, {}^k\omega_4)$ of Eqn. (10) in the article by using MLE to maximize the sum of log-likelihoods over all such training splits [62].
10: Predict using the learned weights ${}^k\omega$ on the corresponding validation splits ${}^kVal^1, {}^kVal^2, ..., {}^kVal^R$.
11: Calculate the prediction errors ${}^kErr^1, {}^kErr^2, ..., {}^kErr^R$ on the validation sets ${}^kVal^1, {}^kVal^2, ..., {}^kVal^R$ respectively.
12: Calculate the sum of all prediction errors ${}^kErr^1, {}^kErr^2, ..., {}^kErr^R$ and let it be ${}^kErr$.
13: **end for**
14: Calculate the average of all K prediction errors ${}^kErr$ ($1 \le k \le K$) and let that be denoted by $AvgErr_i$.
15: **end for**
16: Let p be the index of the $\{\delta, \gamma\}$ pair with the minimum $AvgErr_i$ ($i = 1$ to M). Choose $\{\delta_p, \gamma_p\}$ as the parameter values of the probability weighting function that best describes the probability weights of the adversary population.
17: Given training sets $Tr^1, Tr^2, ..., Tr^R$ and $\{\delta_p, \gamma_p\}$, learn the weights $\omega = (\omega_1, \omega_2, \omega_3, \omega_4)$ of Eqn. (10) in the article by using MLE to maximize the sum of log-likelihoods over all such training splits [62]. The final learned weight set is then (δ_p, γ_p, ω_1, ω_2, ω_3, ω_4).

$$q_i = \frac{e^{\omega_1 x_i + \omega_2 R_i^a + \omega_3 P_i^a}}{\sum_j e^{\omega_1 x_j + \omega_2 R_j^a + \omega_3 P_j^a}}$$

We rewrite the defender's expected utility $U_i^d(x)$ as: $U_i^d(x) = (R_i^d - P_i^d)x_i + P_i^d$. Then defender's overall expected utility can be represented as

$$f(x) = \sum_i q_i U_i^d(x)$$

and the optimization problem is to maximize $f(x)$ under the constraints $\sum_i x_i \le K$ and $0 \le x_i \le 1$.

Assume $\bar{x}$ is the optimal defender strategy and *opt* is the optimal value of defender's overall expected utility. Let $\bar{S}$ be the set of targets with positive coverage probability, i.e., $\bar{S} = \{i|\bar{x}_i > 0\}$. Then $\forall i \in \bar{S}$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} \ge 0$. Otherwise, a defender strategy with a lower coverage probability on target i will achieve a higher defender expected utility than $\bar{x}$, contradict with the optimality. Formally, let $\Delta_i = (0, 0, ..., \delta, 0, 0)$ be a vector with an infinitesimal positive value in ith row. If $\frac{\partial f}{\partial x_i} < 0$, then $f(\bar{x} - \Delta_i) = f(\bar{x}) - \delta \frac{\partial f}{\partial x_i}|_{\bar{x}} > f(\bar{x})$.

Further, the targets in $\bar{S}$ can be divided into two subsets $\bar{S}_1$ and $\bar{S}_2$ where $\bar{S}_1 = \{i|\bar{x}_i = 1\}$ and $\bar{S}_2 = \{i|0 < \bar{x}_i < 1\}$. Then $\forall i, j \in \bar{S}_2$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} \ge 0$. Otherwise, a defender strategy that moves a little bit coverage probability from a target with higher partial derivative to a target with a lower partial derivative will achieve a higher defender expected utility than $\bar{x}$, contradict with the optimality. Formally, if $\frac{\partial f}{\partial x_i}|_{\bar{x}} > \frac{\partial f}{\partial x_j}|_{\bar{x}}$,

$$f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) = f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x} + \Delta_i) + f(\bar{x} + \Delta_i) - f(\bar{x}) \tag{27}$$

$$= -\delta \frac{\partial f}{\partial x_j}|_{\bar{x}+\Delta_i} + \delta \frac{\partial f}{\partial x_i}|_{\bar{x}} \tag{28}$$

$$= -\delta(\frac{\partial f}{\partial x_j}|_{\bar{x}} + \delta \frac{\partial f^2}{\partial x_i \partial x_j}|_{\bar{x}}) + \delta \frac{\partial f}{\partial x_i}|_{\bar{x}} \tag{29}$$

$$= \delta(\frac{\partial f}{\partial x_i}|_{\bar{x}} - \frac{\partial f}{\partial x_j}|_{\bar{x}}) - \delta^2 \frac{\partial f^2}{\partial x_i \partial x_j}|_{\bar{x}} \tag{30}$$

$$> 0 \tag{31}$$

The last inequality is achieved by neglecting the second order term. So $f(\bar{x} + \Delta_i - \Delta_j) > f(\bar{x})$.

Moreover, when $\omega_1 > 0$, $\forall i, j \in \bar{S}_2$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} = 0$. Select two targets $i, j \in \bar{S}_2$. As $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}}$, $f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) = -\delta^2 \frac{\partial f^2}{\partial x_i \partial x_j}|_{\bar{x}}$ according to line (4). The partial derivative of function f is

$$\frac{\partial f}{\partial x_i}|_{\bar{x}} = \omega_1 q_i (U_i^d - f) + q_i (R_i^d - P_i^d)$$

If $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} > 0$, then we have

$$\omega_1(f - U_i^d) < R_i^d - P_i^d \tag{32}$$

and

$$\omega_1(f - U_j^d) < R_j^d - P_j^d \tag{33}$$

Thus

$$\frac{\partial^2 f}{\partial x_i \partial x_j} = \omega_1^2 q_i q_j (2f - U_i^d - U_j^d) - \omega_1 q_i q_j (R_i^d - P_i^d + R_j^d - P_j^d) \tag{34}$$

$$= \omega_1 q_i q_j (\omega_1 (f - U_i^d) + \omega_1 (f - U_j^d) - (R_i^d - P_i^d + R_j^d - P_j^d)) \tag{35}$$

$$< \omega_1 q_i q_j (R_i^d - P_i^d + R_j^d - P_j^d - (R_i^d - P_i^d + R_j^d - P_j^d)) \tag{36}$$

$$= 0 \tag{37}$$

The inequality in line (10) comes from (6) and (7) and the fact the $\omega_1 > 0$. So we have $f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) > 0$, which means moving some coverage probability from target i to target j in $\bar{S}_2$ leads to a defender strategy with higher expected utility. It contradicts with the optimality.

We now prove that $\|\bar{S}_2\| < 2$. As we know $\forall i, j \in \bar{S}_2$, $\frac{\partial f}{\partial x_i}|_{\bar{x}} = \frac{\partial f}{\partial x_j}|_{\bar{x}} = 0$ and $\frac{\partial f^2}{\partial x_i \partial x_j} = 0$, so $f(\bar{x} + \Delta_i - \Delta_j) - f(\bar{x}) = 0$ and we can move some coverage probability from target i to target j to get another optimal strategy $\hat{x}$ with the same expected defender utility opt while $i, j \in \hat{S}_2$. As the $\hat{x}$ is also an optimal strategy, it also satisfies $\frac{\partial f}{\partial x_i}|_{\hat{x}} = \frac{\partial f}{\partial x_j}|_{\hat{x}} = 0$. So we have

$$\omega_1(opt - U_i^d(\hat{x}_i)) = R_i^d - P_i^d \tag{38}$$

$$\omega_1(opt - U_i^d(\bar{x}_i)) = R_i^d - P_i^d \tag{39}$$

From (12)–(13), we get

$$(R_i^d - P_i^d)(\hat{x}_i - \bar{x}_i) = 0 \tag{40}$$

As $\hat{x}_i \neq \bar{x}_i$, we get $R_i^d = P_i^d$, which contradicts with the payoff structure of the game.

Next, we prove $\|\bar{S}_2\| \neq 1$. Assume target i is the only element in $\bar{S}_2$. If $\frac{\partial f}{\partial x_i}|_{\bar{x}} > 0$, we can increase x_i to get a better defender strategy without violating the constraint of total number of resources as $K \in \mathbb{N}$ and all targets other than i are covered with probability 0 or 1. This contradicts with optimality. So $\frac{\partial f}{\partial x_i}|_{\bar{x}} = 0$, i.e., $\omega_1(f - U_i^d(\bar{x}_i)) = R_i^d - P_i^d$. Then

$$\frac{\partial^2 f}{\partial x_i^2} = \omega_1 q_i (\omega_1 (1 - 2q_i)(U_i^d - f) + 2(R_i^d - P_i^d)(1 - q_i))$$

$$= 3\omega_1 q_i (R_i^d - P_i^d)$$

$$> 0$$

So x_i is a minimum point and increasing x_i can get a better defender strategy. Again, this contradicts with optimality.

So $\bar{S}_2 = \emptyset$ and the coverage probabilities of the optimal strategy are chosen only from $0, 1$, i.e., the optimal defender strategy is a pure strategy. □

Appendix C. Sample email

Hi,

Thank you for participating in our experiment. Your base compensation for round 3 has been paid to you via AMT. Thank you also for your valuable comments and suggestions about the game and its strategies. We will definitely take those into account later on. Now, we would want you to participate in the 4th round of our experiment. Please follow the link below to participate: http://cs-server.usc.edu:16568/gamelink/index.jsp

In the first page, please read carefully the compensation details. You will be starting with the performance bonus that you earned in the last round. **The last date to participate in this round of our experiment is Wednesday (November 6 2014) 4 pm PST**. Please try to complete the experiment by the deadline because otherwise deployment of the next round gets delayed.

You are very important to the study and your continued participation is critical. Don't be discouraged if you got caught by a ranger in this round. The chance to play again and earn performance and completion bonuses are coming in a few days. We look forward to your continued participation.

Thank you.

Table 5
Average time (in seconds) taken to play the actual game per round.

Round 1	Round 2	Round 3	Round 4	Round 5
61	52	47	43	39

Appendix D. Challenges and remedies of online repeated measures experiments

In this section we discuss a set of challenges that we faced during our repeated measures experiments on AMT and our methodological contributions towards mitigating those challenges.

For our repeated measures experiments, due to unavailability of data, the strategy shown for each first round of the real game was Maximin. We then learned the model parameters based on previous rounds' data, recomputed and redeployed strategies, and asked the *same* players to play again in the subsequent rounds. For each model, all five rounds were deployed over a span of weeks. Such repeated measures studies on AMT are rare in game-theoretic studies; and certainly none have been conducted in the context of SSGs. Indeed, while the total time of engagement over our 20 experimental settings was 46 weeks, each setting required on average 2.3 weeks (see Table 1). One interesting statistic to note is that the average amount of time taken by the participants to play the actual game based on which the results in our experiments are generated, is 45 seconds, as obtained by computing from the data over all four payoff structures. The average time spent on the actual game per round is shown in Table 5. This, in addition to comments and feedback from the participants (Appendix E), indicates that the participants were spending time considering the trade-offs between the risk of getting captured and obtaining high rewards.

When we started conducting the experiments, we observed that there were very high attrition rates (i.e. people dropped out) for the number of participants between rounds of the game. The varying number of participants from one round to another made it difficult to not only compare between the performance of the model between rounds but also at the end of the five rounds. We hypothesized that the low participant retention rates were due to the following reasons: (i) our initial payment scheme for the participants did not have a large payout at the end of all the rounds of the experiment and therefore participants could potentially leave the experiment at any time depending on how much money they were satisfied with; (ii) initially, each round on average lasted 3.5 weeks as some participants would complete the experiments quickly while others would take a long time to respond: hence several participants may have been dropping out due to such lengthy rounds; and (iii) the lack of commitment to complete a few weeks long repeated measures experiment could also be an issue, as has previously been found in similar repeated measures studies [29].

To mitigate the above challenges, we took the following steps: (i) We set up the payment scheme to consistently reward participation in each round plus offering a relatively high completion incentive at the end of the experiment; (ii) Although we allowed respondents sufficient time (3–4 days on average) to respond [59], as giving them an immediate deadline to finish a particular task can result in high attrition rates, we also maintained persistent contact by sending repeated reminders [21] to the participants, especially to participants who did not respond immediately; (iii) Prior to beginning the first round of our experiment, we asked participants to commit to completing all five rounds, i.e., remain in the study through completion, to be eligible for study enrollment.

The problems addressed by the development of each of our strategy (for example, the choice of concrete metrics, payoffs and incentive mechanisms) does indeed lead to the development of some methodological contributions towards conducting such repeated measures experiments on crowdsourcing platforms like AMT. Since we did not find any related research in this area which specifies a minimum accepted participant retention rate for a repeated measures study, in our work we attempted to achieve a retention rate of at least 80%. This also ensured sufficient number of participants for statistical significance tests. We therefore implemented the steps mentioned above, in order, and measured the effect of the implementation of the corresponding strategy until we achieved the desired participant retention rate. Next, we discuss the implementation details of each of the steps taken (in the order they were implemented) to reduce attrition rates and also provide results showing the improvements due to our approaches.

D.1. Step 1: payment scheme

In our initial payment scheme, shown in Table 6, column 2, participants were paid a fixed 'base compensation' (=\$0.50) for participation in each round of the experiment and a 'performance bonus' based on the points earned (or lost) in each round by attacking a particular target region in the game. The participants started with an initial amount of \$0.50 as the 'performance bonus' in each round. For each reward point earned in a particular round (i.e., if they successfully poached), \$0.10 was added to the initial 'performance bonus'. For each point lost (i.e., if they were captured by the ranger), \$0.10 was deducted from their current 'performance bonus'. The bonus at the end of a particular round was *not* carried forward to the next round and was paid along with the fixed 'base compensation' for that round. For example, for an experiment with two rounds and \$0.50 as the 'base compensation' for each round, if a participant earned a reward point of 9 in the first round and got a penalty of 1 in the second round, (s)he was paid $\$(0.50 + (0.50 + 9 * 0.10)) = \$1.90$ at the end of round 1 and $\$(0.50 + (0.50 - 0.10 * 1)) = \$0.90$ at the end of round 2. With this payment scheme in place, we observed that there

Table 6
Comparison between payment schemes.

Types of compensation	Initial payment scheme	Modified payment scheme
Base compensation per round	0.50$ – paid after each round	0.50$ – paid after each round
Performance bonus per round	0.50$ + 0.10$ per reward point (or −0.10$ per penalty point) – paid after each round	Start with 1.50$ in round 1, then 0.10$ per reward point (or −0.10$ per penalty point) in every round – added to previous rounds' performance bonus and gets carried forward; total accumulated amount is paid after 5 rounds
Completion bonus	0$	2.50$ – paid after 5 rounds

were very high attrition rates, i.e., very few people returned to play in each round, thus making it difficult to compare the performances on various models on a varying number of participants for each model. This is shown in Fig. 25(a), where the x-axis shows rounds of the game and the y-axis shows retention rates. Note that we had to abandon the experiments due to high attrition rates (low retention rates) in round 5 for one of the models (PSUQR) in the first trial and rounds 4 and 5 for PSUQR in the second trial. The failure of this method led us to implement a new payment scheme which is discussed below.

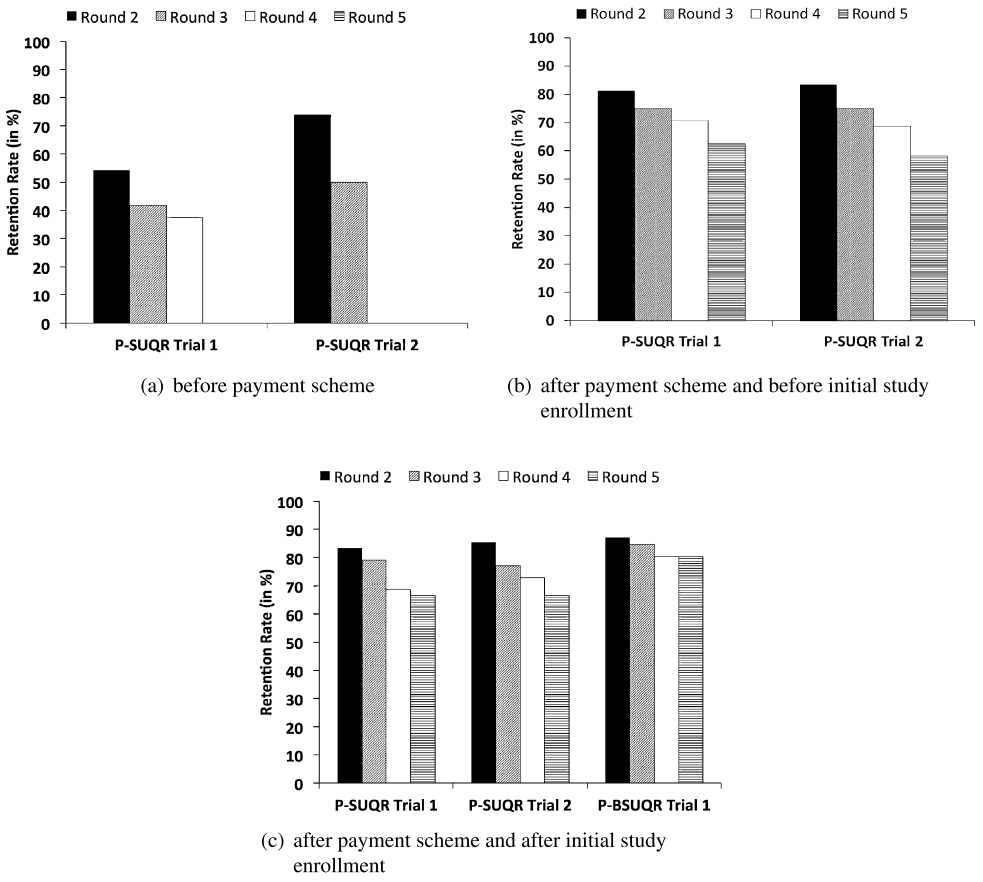
We made three changes to our first method of compensation. First, we introduced a 'completion bonus' (=$2.50) for completing all the rounds of the experiment. Second, like before, to motivate the subjects, the participants were incentivized based on the reward/penalty of the region they chose to attack, i.e., 'performance bonus'. However now, while the base compensation was paid after each round was completed, the 'performance bonus' was carried forward from one round to the next and paid along with the 'completion bonus' at the end of all the rounds of the experiment. Third, the players now started with an initial 'performance bonus' of $1.50 in round 1 and they could win up to a maximum and a minimum amount in each round and hence a very high 'performance bonus' at the end of all the rounds, based on how successful they were. We still had the same 'base compensation' for each round as $0.50, thus resulting in a total base compensation of $2.50 over 5 rounds. However, the maximum amount they could potentially earn at the end of all the rounds from only the performance and completion bonus was as high as $7.60. The performance and completion bonus together at the end of all the rounds was much higher as compared to the total base compensation earned for playing all the 5 rounds. This ensured that majority of the participants remained motivated and returned to play all the rounds. A detailed comparison of the initial and modified payment schemes are shown in Table 6.

To better understand the impact of our new payment scheme, let us take the previous example of a two-round experiment where a participant earned a reward point of 9 in the first round and a penalty of 1 in the second round. According to our new payment scheme, (s)he was paid $0.50 at the end of round 1 (the bonus compensation for round 1). (S)he also earned a performance bonus of $\$(1.50 + 9 * 0.1) = \$2.40$ in round 1 which was carried forward to round 2 and *not* paid at the end of round 1. Then at the end of round 2 she was paid $\$(0.50 + (2.40 - 1 * 0.1) + 2.50) = \$5.30$ (base compensation for round 2 (=$0.50) + performance bonus at the end of round 2 (=$2.30) + completion bonus (=$2.50)). As mentioned before, as compared to our initial payment scheme, this high amount at the end of all the five rounds of our experiments ensured that a relatively high number of participants were retained till the end of the study. On an average, including all the compensations, each participant was paid $7.60 upon completion of our five-round experiments. There were also participants who earned as high as $9 at the end of the five rounds including all the compensations. The effect of this payment scheme on participant retention rate can be seen in Fig. 25(b).

Although the new payment scheme proved effective in retaining more participants, one possibility to be considered is that the performance bonus should not have caused any bias such that the subjects who performed well are more likely to participate in future rounds but who performed poorly are more likely to drop off. We observe from our data that the average retention rates over all games for people who succeeded in the previous round and those who failed in the previous round are 90% and 92% respectively. Therefore, we conclude based on our data that no bias was introduced due to the design of our payment schemes.

D.2. Initial study enrollment

Even though the implementation of the new payment scheme saw an increase in retention rate as shown in Fig. 25(b), there was still a decrease in retention rates over rounds. Therefore, we implemented an approach where the participants had to commit to completing all five rounds before starting the first round of the game. Commitment has been shown to be effective in the past in various scenarios [4,7]. In our game, the participants were asked to either 'agree' or 'disagree' to this commitment. On an average, 96% of the participants who enrolled in AMT for our study agreed to this commitment. These participants were then allowed to proceed towards playing the first round of the game. On the other hand, if they did not agree, they were thanked for their interest in our study, but not allowed to participate any further. The effect of this on the retention rate can be seen in Fig. 25(c). This clearly shows that a significant number of participants with prior commitment towards completing all the rounds of the experiment, returned and completed all the rounds.

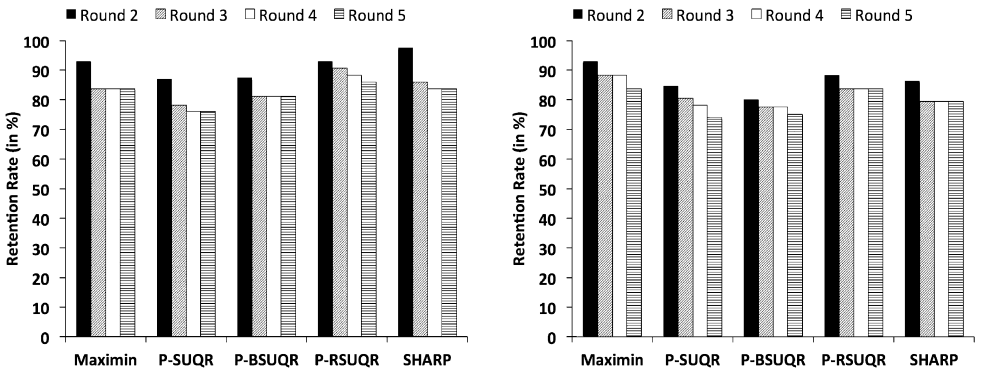


(a) before payment scheme

(b) after payment scheme and before initial study enrollment

(c) after payment scheme and after initial study enrollment

Fig. 25. Retention rates for various models (a) before implementation of our payment scheme, and (b) after implementation of our payment scheme and before implementation of initial study enrollment procedure, and (c) after implementation of our payment scheme and initial study enrollment.



(a) ADS_1

(b) ADS_2

Fig. 26. Retention rates for various models over 4 rounds, starting from round 2 to round 5, on (a) ADS_1 and (b) ADS_2 respectively.

D.3. Reminder emails

Even though the implementation of the payment scheme and initial study enrollment procedures increased the retention rate as shown in Fig. 25(c), the retention rate still decreased over rounds for some of the experiments, even though at a slower rate. Therefore, we sent repeated reminders to the participants with clearly stated deadlines to ensure that they (i) do not forget to participate in the current round, and (ii) also remain motivated throughout the study. The emails were worded carefully and a sample email is shown in Appendix C. Results are shown in Figs. 26(a) and 26(b).

In this section, we gave an overview of our trial and validation games, tested a set of hypothesis to improve participant retention rates for our AMT repeated measures experiments and showed the results of the deployment of each of our strategies to mitigate the challenges in retaining participants. We observed that a delayed compensation scheme along with prior participant commitment and repeated reminders throughout the course of the experiment helped in achieving an average retention rate of 83.69%, which is above the 80% retention rate. In Section 12, we will show results from the comparison of our models based on the data obtained from the corresponding number of participants retained per round.

Appendix E. Participant feedback

After the actual game was over, we asked the participants for feedback regarding the games they played. We asked them two specific questions regarding: (i) their experiences playing the game; and (ii) any strategy they employed while playing the actual game. For point (i), participants primarily mentioned that they enjoyed playing the game and that the instructions were easy to understand while some even mentioned that it was interesting to play a game that involved taking decisions while balancing risk and reward. For point (ii), most participants mentioned that they tried to balance risk and rewards by looking for target areas close to their starting point that had relatively high animal density but still a reasonable probability of success. This risk-reward balance is consistent with the model we learned which put different weights on defender coverage and adversary reward and penalty. This feedback in essence supports formulations such as SHARP studied in this article. Few participants mentioned that they took risks by attacking target areas with high animal density even if the coverage probabilities in that target area was relatively high. Below we share some key feedback by the participants regarding their game playing experiences. Note that these are actual comments from the participants and have not been modified in any way.

E.1. Feedback for "Please tell us about your experience playing the game"

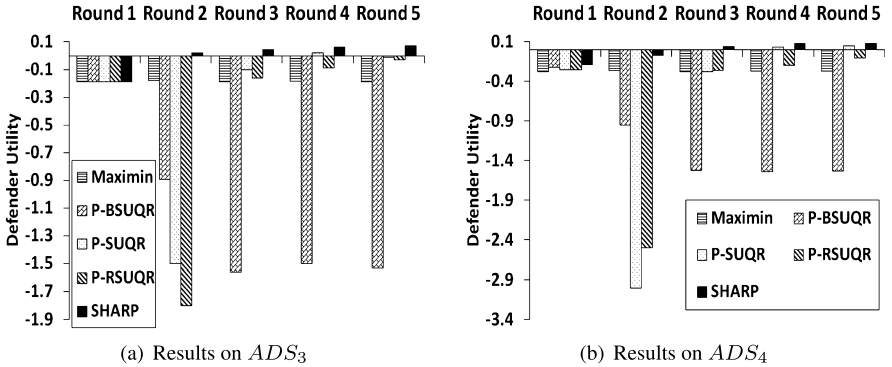
(a) Easy to understand and the visual indications make it even easier, (b) The game was enjoyable and easy to understand, (c) The game was interesting, no bugs encountered, (d) I thought it was fun, very clearly laid out and enjoyable to play, (e) It was fun and kind of exciting. I liked the opportunity and it was interesting to balance risk and reward.

E.2. Feedback for "Did you use a particular strategy in playing the game? If yes, please specify"

(a) Find a greenish square with many hippos, as close as possible to the starting location, (b) I would only target areas with greater than 50% success rate, (c) My basic strategy was to find the most populated, greenest and closest square, (d) I stayed away from the darker red areas, (e) I tried to balance the risk and reward factors. That is, what would be acceptable as a loss versus what I could possibly gain, (f) I tried to get the maximum payoff while minimize the risk of getting caught to an acceptable level, (g) I decided to risk it and set traps in areas that payed well, even though there is high chance that I will get caught.

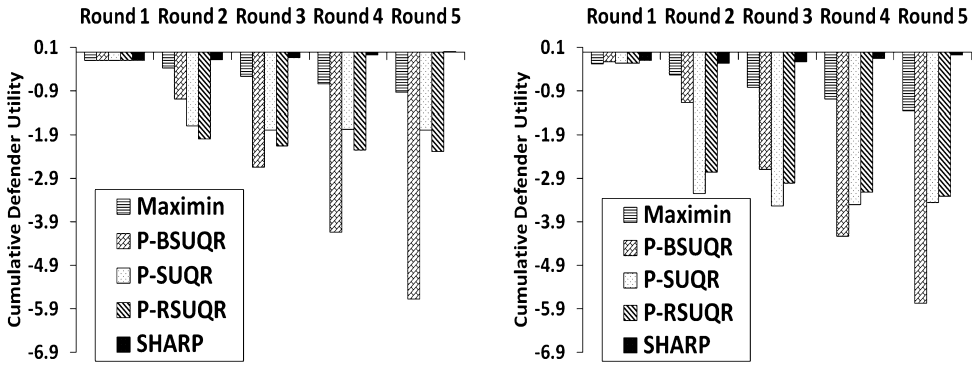
Appendix F. Experimental results on ADS_3 and ADS_4

F.1. Defender utilities



(a) Results on ADS_3 (b) Results on ADS_4

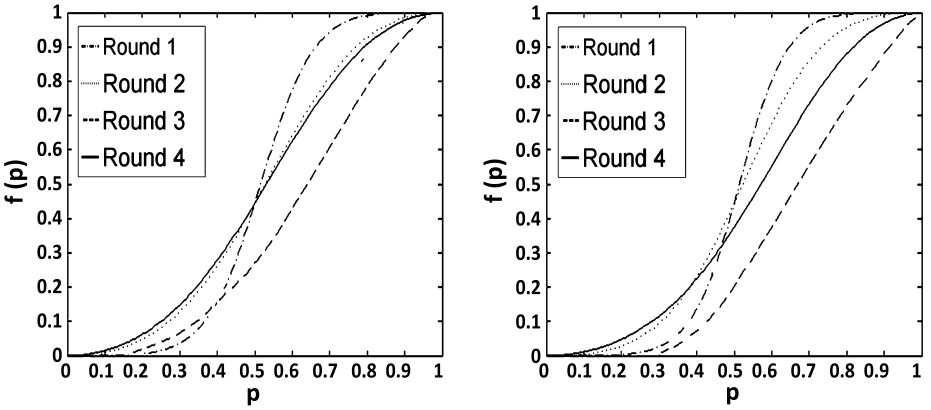
Fig. 27. Defender utilities for various models on ADS_3 and ADS_4 respectively.



(a) Results on ADS_3 (b) Results on ADS_4

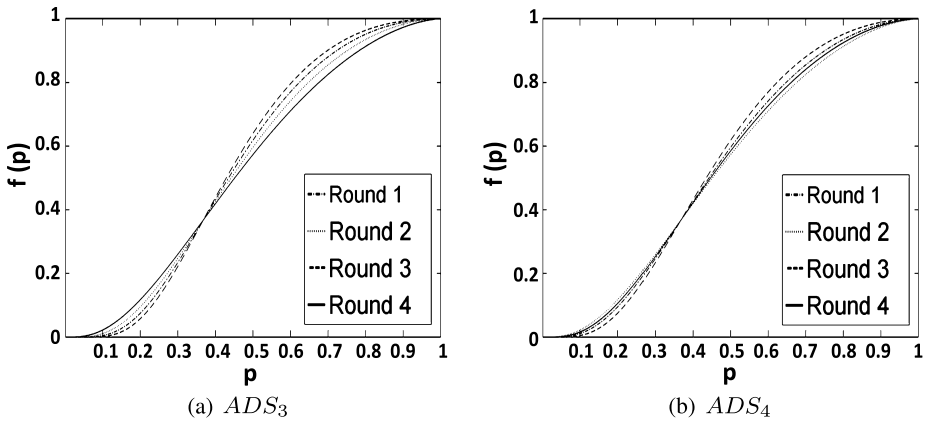
Fig. 28. Cumulative defender utilities for various models on ADS_3 and ADS_4 respectively.

F.2. Learned probability curves



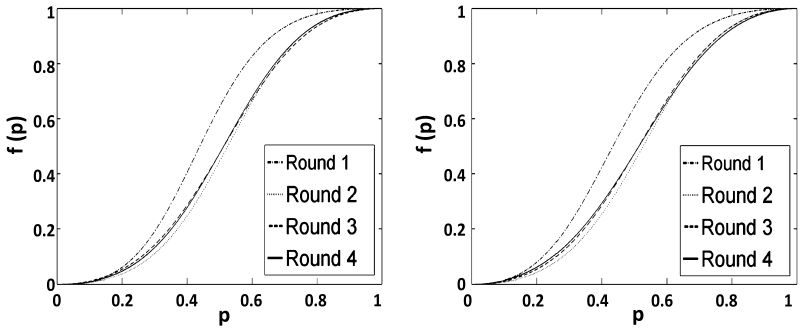
(a) ADS_3 (b) ADS_4

Fig. 29. Learned probability curves for P-SUQR on ADS_3 and ADS_4 respectively.



(a) ADS_3 (b) ADS_4

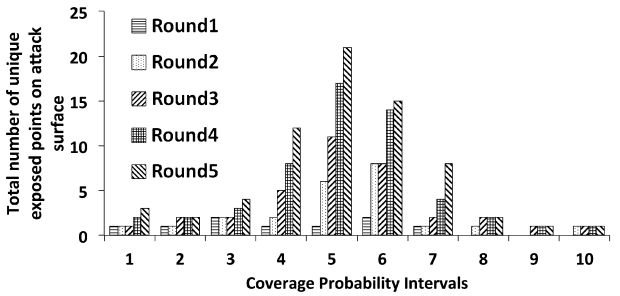
Fig. 30. Learned probability curves with Prelec's probability weighting function for P-SUQR on ADS_3 and ADS_4 respectively.



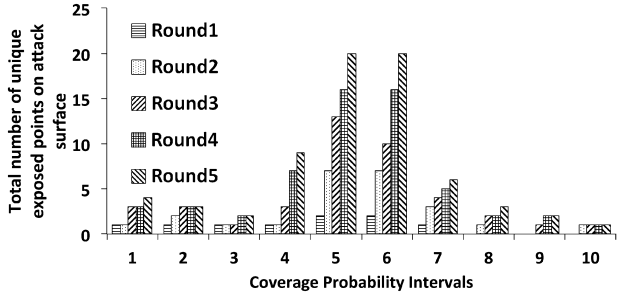
(a) ADS_3 (b) ADS_4

Fig. 31. Learned probability curves for PWV-SUQR on ADS_3 and ADS_4 respectively.

F.3. Evidence of attack surface exposure



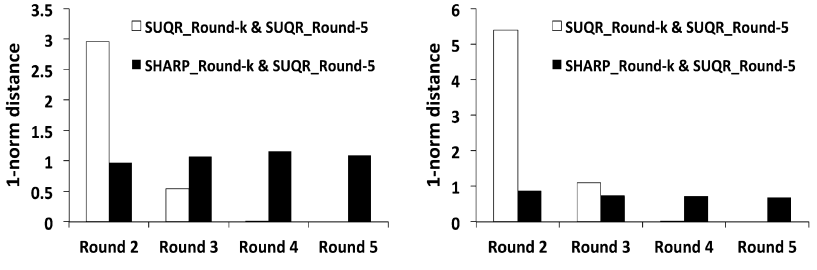
(a) ADS_3



(b) ADS_4

Fig. 32. Total number of unique exposed target profiles till the end of each round for each coverage probability interval for ADS_3 and ADS_4.

F.4. Adaptiveness of SHARP



(a) ADS_3 (b) ADS_4

Fig. 33. Adaptivity of SHARP and convergence of P-SUQR on payoff structures ADS_3 and ADS_4 respectively.

References

[1] Ugandans fear curse of oil wealth as it threatens to blight 'pearl of Africa', http://www.theguardian.com/world/2013/dec/29/ugandans-oil-blight-pearl-africa, Accessed: November 8, 2014.
[2] Y.D. Abbasi, M. Short, A. Sinha, N. Sintov, C. Zhang, M. Tambe, Human adversaries in opportunistic crime security games: evaluating competing bounded rationality models, in: Conference on Advances in Cognitive Systems, 2015.
[3] M. Abdellaoui, O. L'Haridon, H. Zank, Separating curvature and elevation: a parametric probability weighting function, J. Risk Uncertain. 41 (1) (2010) 39–65.
[4] E. Aharonovich, P.C. Amrhein, A. Bisaga, E.V. Nunes, D.S. Hasin, Cognition commitment language, and behavioral change among cocaine-dependent patients, Psychol. Addict. Behav. 22 (4) (2008) 557–567.
[5] Y. Alarie, G. Dionne, Lottery decisions and probability weighting function, J. Risk Uncertain. 22 (1) (2001) 21–33.
[6] A. Azaria, Y. Gal, S. Kraus, C. Goldman, Strategic advice provision in repeated human–agent interactions, Auton. Agents Multi-Agent Syst. (2015) 1–26.
[7] K. Baca-Motes, A. Brown, A. Gneezy, E.A. Keenan, L.D. Nelson, Commitment and behavior change: evidence from the field, J. Consum. Res. 39 (5) (2013) 1070–1084.
[8] K. Bagwell, Commitment and observability in games, Technical report, University, Center for Mathematical Studies in Economics and Management Science, 1992.
[9] M.-F. Balcan, A. Blum, N. Haghtalab, A.D. Procaccia, Commitment without regrets: online learning in Stackelberg security games, in: Proceedings of the Sixteenth ACM Conference on Economics and Computation, EC '15, 2015.
[10] J. Beck, W. Forstmeier, Superstition and belief as inevitable by-products of an adaptive learning strategy, Hum. Nat. 18 (1) (2007) 35–46.
[11] A.W. Beggs, On the convergence of reinforcement learning, Int. J. Econ. Theory 122 (1) (2005) 1–36.
[12] A.J. Berinsky, G.A. Huber, G.S. Lenz, Evaluating online labor markets for experimental research: amazon.com's mechanical turk, Polit. Anal. 20 (3) (2012) 351–368.
[13] C. Bishop, Pattern Recognition and Machine Learning, Springer, 2007.
[14] A. Blum, N. Haghtalab, A. Procaccia, Learning optimal commitment to overcome insecurity, in: Proceedings of the 28th Annual Conference on Neural Information Processing Systems, NIPS, 2014.
[15] E. Brunswik, The conceptual framework of psychology, in: International Encyclopedia of Unified Science, vol. 1, University of Chicago Press, 1952, Number 10.
[16] M. Caravolas, C. Hulme, M.J. Snowling, The foundations of spelling ability: evidence from a 3-year longitudinal study, J. Mem. Lang. 45 (4) (2001) 751–774.
[17] R. Ceren, P. Doshi, M. Meisel, A. Goodie, D. Hall, On modeling human learning in sequential games with delayed reinforcements, in: Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 3108–3113.
[18] C. Chabris, D. Laibson, J. Schuldt, Intertemporal choice, in: The New Palgrave Dictionary of Economics, vol. 2, 2006.
[19] R. Cominetti, E. Melo, S. Sorin, A payoff-based learning procedure and its application to traffic games, Games Econ. Behav. 70 (1) (2010) 71–83.
[20] V. Conitzer, T. Sandholm, Computing the optimal strategy to commit to, in: Proceedings of the 7th ACM Conference on Electronic Commerce, EC '06, 2006, pp. 82–90.
[21] R.B. Cotter, J.D. Burke, M. Stouthamer-Loeber, R. Loeber, Contacting participants for follow-up: how much effort is required to retain participants in longitudinal studies?, Eval. Program Plann 28 (1) (2005) 15–21.
[22] J. Cui, R. John, Empirical comparisons of descriptive multi-objective adversary models in Stackelberg security games, in: Conference on Decision and Game Theory for Security, GameSec, 2014.
[23] Y. Deng, D.S. Hillygus, J.P. Reiter, Y. Si, S. Zheng, Handling attrition in longitudinal studies: the case for refreshment samples, Stat. Sci. 28 (2) (2013) 238–256.
[24] L. Devenport, Superstitious bar pressing in hippocampal and septal rats, Science 18 (1) (1979) 35–46.
[25] T.G. Dietterich, Approximate statistical tests for comparing supervised classification learning algorithms, Neural Comput. 10 (7) (1998) 1895–1923.
[26] S.A. Dudani, The distance-weighted k-nearest-neighbor rule, IEEE Trans. Syst. Man Cybern. Syst. SMC-6 (4) (April 1976) 325–327.
[27] J. Elster, A plea for mechanisms, in: Social Mechanisms: An Analytical Approach to Social Theory, 2005.
[28] I. Erev, A. Roth, Predicting how people play games: reinforcement learning in experimental games with unique, mixed strategy equilibria, Am. Econ. Rev. 88 (4) (9, 1998) 848–881.
[29] M. Estrada, A. Woodcock, P.W. Schultz, Tailored panel management: a theory-based approach to building and maintaining participant commitment to a longitudinal study, in: Evaluation Review, 2014.
[30] N. Etchart-Vincent, Probability weighting and the level and spacing of outcomes: an experimental study over losses, J. Risk Uncertain. 39 (1) (2009) 45–63.
[31] F. Fang, P. Stone, M. Tambe, When security games go green: designing defender strategies to prevent poaching and illegal fishing, in: International Joint Conference on Artificial Intelligence, IJCAI, 2015.
[32] D. Farrington, R. Loeber, B. Welsh, Longitudinal-experimental studies, in: Handbook of Quantitative Criminology, Springer, New York, 2010.
[33] N. Feltovich, Reinforcement-based vs. belief-based learning models in experimental asymmetric-information games, Econometrica 68 (3) (2000) 605–641.
[34] B. Ford, T. Nguyen, M. Tambe, N. Sintov, F.D. Fave, Beware the soothsayer: from attack prediction accuracy to predictive reliability in security games, in: Conference on Decision and Game Theory for Security, GameSec, 2015.
[35] N. Gans, G. Knox, R. Croson, Simple models of discrete choice and their performance in bandit experiments, Manuf. Serv. Oper. Manag. 9 (4) (9, 2007) 383–408.
[36] H. Goldstein, Handling attrition and non-response in longitudinal data, Longitud. Life Course Stud. 1 (1) (2009) 63–72.
[37] R. Gonzalez, G. Wu, On the shape of the probability weighting function, Cogn. Psychol. 38 (1999) 129–166.
[38] M. Hamisi, Identification and mapping risk areas for zebra poaching: a case of Tarangire National Park, Tanzania, Thesis, ITC, 2008.
[39] G. Hammond, A correlation of reaction rates, J. Am. Chem. Soc. 77 (2) (1955).
[40] W. Haskell, D. Kar, F. Fang, M. Tambe, S. Cheung, E. Denicola, Robust protection of fisheries with compass, in: Innovative Applications of Artificial Intelligence, IAAI, 2014.
[41] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer-Verlag, 2009.
[42] G.W. Heiman, Research Methods in Psychology, 3rd edition, Houghton Mifflin Company, Boston and New York, 2002.
[43] E. Hopkins, Two competing models of how people learn in games, Technical report, David K. Levine, 2001.
[44] S.J. Humphrey, A. Verschoor, The probability weighting function: experimental evidence from Uganda, India and Ethiopia, Econ. Lett. 84 (3) (September 2004) 419–425.
[45] S. Jajodia, A.K. Ghosh, V. Swarup, C. Wang, X.S. Wang, Moving Target Defense: Creating Asymmetric Uncertainty for Cyber Threats, 1st edition, Springer Publishing Company, Incorporated, 2011.
[46] M. Johanson, M. Bowling, Data biased robust counter strategies, in: Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics, AISTATS, 2009.

[47] M. Johanson, M. Zinkevich, M. Bowling, Computing robust counter-strategies, in: Proceedings of the Annual Conference on Neural Information Processing Systems, NIPS, 2007.
[48] D. Kahneman, A. Tversky, Prospect theory: an analysis of decision under risk, Econometrica 47 (2) (1979) 263–291.
[49] D. Kar, F. Fang, F.D. Fave, N. Sintov, M. Tambe, "A game of thrones": when human behavior models compete in repeated Stackelberg security games, in: International Conference on Autonomous Agents and Multiagent Systems, AAMAS, 2015.
[50] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection, in: IJCAI, Morgan Kaufmann, 1995, pp. 1137–1143.
[51] D. Korzhyk, V. Conitzer, R. Parr, Complexity of computing optimal Stackelberg strategies in security resource allocation games, in: Proceedings of the National Conference on Artificial Intelligence, AAAI, 2010, pp. 805–810.
[52] P. Leclerc, Prospect theory preferences in noncooperative game theory, PhD thesis, Virginia Commonwealth University, 2014.
[53] A.M. Lemieux, Situational Crime Prevention of Poaching, Crime Science Series, Routledge, 2014.
[54] J. Letchford, V. Conitzer, K. Munagala, Learning and approximating the optimal strategy to commit to, in: Proceedings of the 2nd International Symposium on Algorithmic Game Theory, SAGT '09, Springer-Verlag, Berlin, Heidelberg, 2009, pp. 250–262.
[55] P.K. Manadhata, J.M. Wing, An attack surface metric, IEEE Trans. Softw. Eng. 37 (3) (2011) 371–386.
[56] J. Marecki, G. Tesauro, R. Segal, Playing repeated Stackelberg games with unknown opponents, in: AAMAS, 2012, pp. 821–828.
[57] P. McCracken, M. Bowling, Safe strategies for agent modelling in games, in: Proceedings of the National Conference on Artificial Intelligence, AAAI, 2004.
[58] D. McFadden, Quantal choice analysis: a survey, Ann. Econ. Soc. Meas. 5 (4) (1976) 363–390.
[59] S.W. Menard, Handbook of Longitudinal Research: Design, Measurement, and Analysis, Academic Press, 2008.
[60] M. Montesh, Rhino poaching: a new form of organised crime, Technical report, College of Law Research and Innovation Committee of the University of South Africa, 2013.
[61] W. Moreto, To conserve and protect: examining law enforcement ranger culture and operations in Queen Elizabeth National Park, Uganda, Thesis, Rutgers, 2013.
[62] T.H. Nguyen, R. Yang, A. Azaria, S. Kraus, M. Tambe, Analyzing the effectiveness of adversary modeling in security games, in: AAAI, 2013.
[63] M.J. Osborne, A. Rubinstein, A Course in Game Theory, MIT Press, Cambridge, 1994.
[64] D.C. Parkes, A. Mao, Y. Chen, K.Z. Gajos, A. Procaccia, H. Zhang, TurkServer: enabling synchronous and longitudinal online experiments, in: Proceedings of the Fourth Workshop on Human Computation, HCOMP'12, AAAI Press, 2012.
[65] P. Paruchuri, J.P. Pearce, J. Marecki, M. Tambe, F. Ordonez, S. Kraus, Playing games for security: an efficient exact algorithm for solving Bayesian Stackelberg games, in: Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems – vol. 2, AAMAS, 2008, pp. 895–902.
[66] J. Pita, M. Jain, M. Tambe, F. Ordóñez, S. Kraus, Robust solutions to Stackelberg games, Artif. Intell. 174 (15) (2010) 1142–1171.
[67] J. Pita, R. John, R. Maheswaran, M. Tambe, S. Kraus, A robust approach to addressing human adversaries in security games, in: ECAI, 2012.
[68] M. Ponsen, S.D. Jong, M. Lanctot, Computing approximate Nash equilibria and robust best-responses using sampling, J. Artif. Intell. Res. (2011).
[69] D. Prelec, The probability weighting function, Econometrica 66 (3) (1998) 497–527.
[70] G. Seni, J.F. Elder, Ensemble methods in data mining: improving accuracy through combining predictions, Synth. Lect. Data Min. Knowledge Discov. 2 (1) (2010) 1–126.
[71] R.C. Silver, E.A. Holman, D.N. McIntosh, M. Poulin, V. Gil-Rivas, Nationwide longitudinal study of psychological responses to September 11, JAMA 288 (10) (2002) 1235–1244.
[72] B.F. Skinner, The Behavior of Organisms: An Experimental Analysis, Appleton-Century, New York, 1938.
[73] B.F. Skinner, Superstition' in the pigeon, J. Exp. Psychol. 38 (1948) 168–172.
[74] B.F. Skinner, Science and Human Behavior, Simon and Schuster, 1953.
[75] M. Tambe, Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned, Cambridge University Press, New York, NY, 2011.
[76] J. Tsai, Z. Yin, J. young Kwak, D. Kempe, C. Kiekintveld, M. Tambe, Urban security: game-theoretic resource allocation in networked physical domains, in: National Conference on Artificial Intelligence, AAAI, 2010.
[77] A. Tversky, D. Kahneman, Advances in prospect theory: cumulative representation of uncertainty, J. Risk Uncertain. 5 (4) (1992) 297–323.
[78] J. Twisk, W. de Vente, Attrition in longitudinal studies, J. Clin. Epidemiol. 55 (4) (2002) 329–337.
[79] Y.A. Wato, G.M. Wahungu, M.M. Okello, Correlates of wildlife snaring patterns in Tsavo West National Park, Kenya, Biol. Conserv. 132 (4) (2006) 500–509.
[80] R. Yang, B. Ford, M. Tambe, A. Lemieux, Adaptive resource allocation for wildlife protection against illegal poachers, in: International Conference on Autonomous Agents and Multiagent Systems, AAMAS, 2014.
[81] R. Yang, C. Kiekintveld, F. Ordonez, M. Tambe, R. John, Improving resource allocation strategy against human adversaries in security games, in: IJCAI, 2011.
[82] R. Yang, C. Kiekintveld, F. Ordonez, M. Tambe, R. John, Improving resource allocation strategies against human adversaries in security games: an extended study, Artif. Intell. 195 (2013) 440–469.
[83] R. Yang, F. Ordonez, M. Tambe, Computing optimal strategy against quantal response in security games, in: Proceedings of the 11th International Conference on Autonomous Agents and Multiagent Systems – vol. 2, AAMAS '12, 2012, pp. 847–854.
[84] M. Zollo, Superstitious learning with rare strategic decisions: theory and evidence from corporate acquisitions, Organ. Sci. 20 (5) (2009) 894–908.