Playing Repeated Security Games with No Prior Knowledge*

Haifeng Xu
University of Southern California
haifengx@usc.edu

Long Tran-Thanh
University of Southampton
l.tran-thanh@soton.ac.uk

Nicholas R. Jennings
University of Southampton
nrj@ecs.soton.ac.uk

ABSTRACT

This paper investigates repeated security games with *unknown* (to the defender) game payoffs and attacker behaviors. As existing work assumes prior knowledge about either the game payoffs or the attacker's behaviors, they are not suitable for tackling our problem. Given this, we propose the first efficient defender strategy, based on an adversarial online learning framework, that can provably achieve good performance guarantees without any prior knowledge. In particular, we prove that our algorithm can achieve low performance loss against the best *fixed* strategy on hindsight (i.e., having full knowledge of the attacker's moves). In addition, we prove that our algorithm can achieve an efficient competitive ratio against the optimal adaptive defender strategy. We also show that for zero-sum security games, our algorithm achieves efficient results in approximating a number of solution concepts, such as algorithmic equilibria and the minimax value. Finally, our extensive numerical results demonstrate that, without having any prior information, our algorithm still achieves good performance, compared to state-of-the-art algorithms from the literature on security games, such as SUQR [19], which require significant amount of prior knowledge.

General Terms

Game theory; Security; Theory

Keywords

Repeated Security Games; No-regret Learning; Adaptive Strategy

1. INTRODUCTION

In the recent years, security games have been widely used in many areas of artificial intelligence [24]. These games typically consist of a Stackelberg model in which the defender allocates a limited number of resources to protect a set of targets based on a randomized strategy, while the attacker, upon learning the strategy, chooses an optimal subset of targets to attack. Motivated by anti-terrorist patrolling, earlier work on security games typically focuses on one-shot game models, e.g., [22, 12]. However, recently, there has been a surge of interests in addressing various security domains involving *repeated* interactions between the defender and a *bounded-rational* attacker. These repeated security game models are motivated by many important real-world problems such as wildlife patrolling [26] and illegal fishing monitoring [11]. Due to the repeated manner of the games, one-shot models are not suitable to tackle these problems, since they do not take into account the learning and adaptive behaviour of the attackers. As such, new solutions are required to address this challenge within the repeated security games. In the literature on security games, such solutions typically assume a specific bounded-rationality attacker behavior model (e.g., the Quantal Response (QR) model) in order to predict the future behaviour of the attacker. However, as pointed out by Kar et al. [14], these bounded-rationality models suffer from a number of limitations. Particularly, they fall short in capturing adaptive attackers, who can adversarially change their attacking strategy over time based on the defender's past actions. Unfortunately, attackers are typically adaptive in real scenarios, making such approaches unsuitable to tackle real problems.

Although Kar et al. [14] refined the attacker behavior model and took into account the adaptive behaviour of attackers, such type of approaches still has a number of shortcomings, namely: (i) it assumes that the attacker's behaviour model, along with *all* the features/patterns that affect the model, is known a priori; (ii) it assumes that the attacker payoffs are known by the defender in advance; and (iii) it is computationally intractable and only local optimal strategy can be computed, therefore no theoretical performance guarantee can be provided. However in real-scenarios, attackers are not necessarily following a particular model and it is very hard to predict their behaviors. Moreover, it is widely recognized that the defender usually does not know the attacker's payoffs (Kiekintveld et al. [15], Blum et al. [3]). In fact, the defender may *even* not precisely know her own payoffs due to uncertainties in some domains. For example, in wildlife poaching or illegal fishing domains, the payoff of a target (i.e., a subarea) at each round depends on the *amount* and *types* of species showing up, which is random and difficult to estimate due to too much uncertainty in nature.

To overcome these issues, we propose a novel defender strategy for repeated security games, namely Follow the Perturbed Leader with Uniform Exploration (or **FPL-UE** for short), which is a variant of the celebrated Follow the Perturbed Leader (FPL) algorithm, a state-of-the-art method from the online learning theory literature [13]. In particular, we show that the defender's patrolling problem in repeated security games can be formulated as a combinatorial adversarial online learning problem, where at each round, an opponent (i.e., the attacker) adversarially sets a multidimensional vector of positive rewards, and the learner (i.e., the defender) can only choose to see a *subset* of entries of this vector (i.e., targets to protect), while the rest remains *unrevealed*. The learner's reward is the sum of the revealed entries, and her goal is to efficiently

*First two authors contribute equally.

Appears in: *Proceedings of the 15th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2016), J. Thangarajah, K. Tuyls, C. Jonker, S. Marsella (eds.), May 9–13, 2016, Singapore.*

maximise the total rewards against this adaptive and adversarial opponent (for more details, see, e.g., [6]). The current state of the art of the literature is the method proposed by Neu and Bartok [19].[1] However, their algorithm works only when the learner suffers *loss* (i.e., the goal is loss minimisation), while in our case, as we will show in Section 2, the learner collects *rewards* (i.e., the goal is reward maximisation). As Neu and Bartok noted *explicitly* in the paper, their algorithm, in particular, a key lemma in proving the regret guarantee, *cannot* be directly adapted for the reward scenario to guarantee fast regret convergence. As such, the FPL algorithm of Neu and Bartok cannot be directly applied to our setting. Against this background, we propose a new analysis for the reward maximisation scenario. In particular, we show that FPL-UE, which augments the algorithm in [19] with more ingredient of exploration and exploits the structures of security games, can provide efficient and provable performance guarantees for the defender in a repeated security game. Our numerical evaluation based on simulations also show the advantage of our algorithm and the failure of convergence of the algorithm in [19] when dealing with reward maximization cases.

Furthermore, our approach also enjoys the following advantages: (i) it does not require any prior knowledge about attacker's behaviour, and thus, is suitable for handling any types of attackers; (ii) it does not require any prior knowledge about the game payoffs either, and thus, can deal with payoff uncertainty; and (iii) it has efficient theoretical performance guarantees. In particular, the algorithm assumes an arbitrary attacker, and puts no assumptions on their behaviour. It then efficiently balances between exploration (i.e., learn which strategy is the best against the particular attacker) and exploitation (maximises the total utility over time). To do so, at each round, our algorithm calculates a mixed strategy that is a careful combination of uniform random distribution (for exploration) and a distribution derived from solving an optimisation problem (for exploitation) with perturbed values (i.e., with some artificially added noise). By doing so, we show that FPL-UE can provably achieve low-regret (i.e., performance loss) bounds, compared to that of the best fixed strategy on hindsight.

In particular, we show that the regret bound of FPL-UE is at most $O(\sqrt{T})$ in total within T time steps. This sub-linear regret implies that the average regret per time step is converging to 0 as T tends to infinity. Thus, the behaviour of FPL-UE converges to the best fixed strategy (i.e., the best response to the attacker's strategy) on hindsight, with a convergence rate of $O(\frac{1}{\sqrt{T}})$. We further show that within zero-sum security games (which are well-motivated by many real-world applications [11]), if both the defender and attacker use FPL-UE to make their actions, they can achieve an approximate (algorithmic) equilibrium. We also show that the minimax value, which is a powerful solution concept of the zero-sum games, can also be approximated with provable approximation guarantees, by using FPL-UE against a simulated user, who also uses FPL-UE to make decisions. As such, our work contributes to the state of the art in the following aspects:

- We provide a novel approach, based on online combinatorial optimisation, for designing efficient defender strategies in repeated security games. Our approach does not require additional prior knowledge about the attacker and the environment, and can be applied against adaptive attackers. Our approach is the first defender strategy for repeated security games that enjoys provable theoretical performance guarantees.

[1]We refer the reader to [6, 20] for a more detailed survey of the combinatorial online learning literature.

- We also show that for an important sub-class of repeated security games, namely the zero-sum games, we can approximate the algorithmic equilibrium and the minimax value of the game by applying our strategy for both sides.

- Finally, by using extensive numerical evaluations, we demonstrate that our algorithm, while it does not require any prior knowledge, can still achieve competitive performance, compared to the defender strategy generated using the Subjective Utility Quantal Response (**SUQR**) model [21], a state-of-the-art attacker behavior model that has been tested in real-world repeated security games for protecting wildlife and fishery [4, 7], which heavily relies on the existence of prior knowledge. We also show that the additional uniform exploration step in our algorithm is essential, as it significantly outperform the existing version of Neu and Bartok [19] in practice.

1.1 Additional Related Work

Our work is potentially related to two lines of research in security games. One line of work deals with uncertainties in security games, including payoff uncertainties, attacker surveillance and behavior uncertainties [23, 15, 1, 21]. These works either require strong model assumptions (e.g., the Quantal Response assumption) or are too conservative (e.g., the robust optimization approach). Moreover, the models are typically computationally hard and few theoretical guarantees can be given. In contrast, our approach requires no prior knowledge and is efficient. Another line of work is the recent research on learning in security games, but they are all different from ours. [3] considers the setting with unknown attacker payoffs and studies the defender's problem of learning the Stackelberg mixed strategy by attacker-best-response queries. Their setting, goal and approach are different from ours. [2] considers repeated security games with varying attacker types captured by different payoff matrices and uses the online learning approach, but they assume full knowledge of the game payoffs and perfect rationality of the attackers. Also, their algorithm is not computationally efficient. (Klima et al [16, 17]) consider repeated border patrolling with an online learning approach. They experimentally applied several known learning algorithms, but with no theoretical analysis.

2. PROBLEM FORMULATION

In this section, we first describe the repeated security game setting and discuss the assumption of information available to the defender. We then show how to formulate the repeated security game as an adversarial online combinatorial optimisation.

The Game: We consider a repeated security game played between a defender and an attacker. The defender has k security resources and needs to protect n (with $n >> k$) targets, while the attacker also has *multiple* attack resources and can attack at most m targets at the same time. We use $[n]$ to denote the set of all targets. A defender pure strategy is a subset of $[n]$, with cardinality *at most* k, indicating the set of protected targets in the pure strategy. Alternatively, we may use a *binary* vector $v \in \{0,1\}^n$ to denote a generic defender pure strategy, where entry i is 1 if and only if target i is protected in this pure strategy. Throughout this paper, we will use an n-dimensional binary vector to denote a pure strategy, and $\mathcal{V} \subseteq \{0,1\}^n$ to denote the set of all defender pure strategies. Therefore, $||v||_1 \leq k$ for any $v \in \mathcal{V}$. However, set $\mathcal{V}$ needs *not* to be the set of all v's satisfying $||v||_1 \leq k$ due to possible scheduling constraints in practice (see [12] for examples). Naturally, we assume that any target can be protected by at least some $v \in \mathcal{V}$. A defender mixed strategy is simply a distribution over $\mathcal{V}$. Similarly, we use $a \in \{0,1\}^n$ to denote a generic attacker pure strategy and

For all t = 1,2,...,T, repeat

1. The defender computes a mixed strategy from which she samples a pure strategy $v_t \in \mathcal{V}$ to play.
2. The attacker plays a pure strategy $a_t \in \mathcal{A}$;
3. The defender gets a utility depending on both v_t, a_t and potential uncertainties of U_i^u, U_i^c for all $i \in [n]$.
4. The defender observes feedbacks from the targets she visited in the pure strategy v_t;

Figure 1: The Repeated Security Game Procedure

set $\mathcal{A}$ to denote the set of all attacker pure strategies. Naturally, $||a||_1 \leq m$ for any $a \in \mathcal{A}$. Given that target i is attacked, the defender gets utility $U_i^c \in [-0.5, 0.5]$ if target i is *covered* and gets $U_i^u \in [-0.5, 0.5]$ if i is *uncovered*.[2] As a standard assumption, we assume $U_i^c > U_i^u$, i.e., covering a target is strictly better for the defender than uncovering it. We allow *uncertainties* of U_i^c and U_i^u, since they can be random, potentially depending on environmental factors. The game is played for T rounds (see Figure 1).

One might wonder why repeated security games can be viewed as an online *reward* maximization problem since at the first glance, security games seem a loss minimization problem, i.e., the defender wants to minimize the loss from attack. Some thoughts reveal that this is actually *not* true, because the defender actively seeks to catch attackers in security games. In particular, at each round, the attacker attacks several targets. Then the defender's task is precisely to find these attacked targets and convert their states from "unprotected" to "protected", by which her utility converts from U_i^u to U_i^c, or equivalently, gains a reward $U_i^c - U_i^u (> 0)$. As shown later, our mathematical formulation formalizes this intuition.

Information and Behaviour Assumptions: The amount of information (or knowledge) players possess in a game has a profound influence on their equilibrium behaviour. Previous work in security games mostly assumes plenty knowledge for the defender and attacker. They know the payoff structures of the game, or at least the *range* of payoffs in some uncertain settings; And they know each other's actions or behavior models. However, as we mentioned above, in some important domains like wildlife poaching or illegal fishing, the value of a target at each round is unknown a priori and depends on random environmental factors. Moreover, even given the payoffs, it is still hard to predict the attackers' behaviours. This is due to at least two reasons: (i) the attacker may have different knowledge and constraints from what the defender thought; (ii) the attacker may be irrational to any extent.

Instead, our model adopts a completely different perspective – we do not require the defender to have any knowledge regarding the payoffs in advance. More practically, we assume the defender can only observe the real-time utilities at those targets where a patroller is sent. On the other hand, the only requirement for the attacker is that, he *cannot* observe the defender's move at current round, i.e., players move simultaneously at each round. This is reasonable because each round models a single-shot game. Furthermore, we assume that the defender is an expected utility maximiser. Finally, our algorithm requires no behaviour model for the attacker. Put differently, the attacker could be a utility maximiser or be irrational to any extent; could know the payoff structure or may be uncertain about the game to any level; could be totally adversarial, or could be a random player.

Utility Model and Problem Formulation: Given any defender and attacker pure strategy v_t and a_t at time t, the defender's utility is:

$$u(v_t, a_t) = \sum_{i\in[n]} v_{t,i} a_{t,i} U_i^c + \sum_{i\in[n]} (1 - v_{t,i}) a_{t,i} U_i^u$$

[2]Here, "0.5" is for normalization reason.

where the first [second] term is the utility from protected [unprotected] targets. We rewrite the utility as follows:

$$\begin{aligned} u(v_t, a_t) &= \sum_{i\in[n]} v_{t,i} a_{t,i} [U_i^c - U_i^u] + \sum_{i\in[n]} a_{t,i} U_i^u \\ &= v_t \cdot r_t(a_t) + C(a_t) \end{aligned} \tag{1}$$

where $r_t(a_t) \in R^n$ satisfying $r_{t,i} = a_{t,i}[U_i^c - U_i^u] \in [0, 1]$ (since $0.5 \geq U_i^c > U_i^u \geq -0.5$) and $C(a_t) = \sum_{i\in[n]} a_{t,i} U_i^u$ both depend *only* on the attacker strategy a_t. Here " $\cdot$ " denotes vector inner product, as will be used throughout the paper.

Eq. (1) provides another view of a repeated security game. That is, given the attacker's strategy at any round, the defender's task is to "collect" positive reward using k resources. This naturally connects to combinatorial adversarial online learning settings [6]. Note that if r_t is drawn independently from the same distribution for any t, then r_t is stochastic and this case admits efficient and regret-tight algorithms [10, 18].

In this paper, however, we consider that $r_t(a_t)$ is chosen adversarially. This is due to the following reasons. First, it is consistent with the nature of defender-attacker interactions especially when the attacker is adaptive. Second, due to irrationality and defender's incomplete knowledge of the attacker as well as uncontrollable environmental factors affecting payoffs, it is very difficult to estimate a distribution for r_t. Therefore, we take the worse-case analysis and assume that the reward is chosen adversarially. Let $\mathcal{F}_t$ denote the history information of the game by time t (inclusive), and $\mathcal{F}_0$ denote no history information at all. We allow r_t to depend on $\mathcal{F}_{t-1}$ but *not* v_t, i.e., the defender's play at round t. Given $a_1, ..., a_T$, we are interested in finding an *online* policy $v_1(\mathcal{F}_0), ..., v_T(\mathcal{F}_{T-1})$ (possibly randomized) that maximizes the defender's expected utility $\mathbb{E}\left[\sum_{t=1}^T u(v_t, a_t)\right]$ where the expectation is taken over the randomness of the policy and environment. Alternatively, we aim at minimizing the defender's *regret*, defined as:

$$\begin{aligned} R_T &= \max_{v\in\mathcal{V}} \sum_{t=1}^T u(v, a_t) - \mathbb{E}\left[\sum_{t=1}^T u(v_t, a_t)\right] \\ &= \max_{v\in\mathcal{V}} \sum_{t=1}^T r_t \cdot v - \mathbb{E}\left[\sum_{t=1}^T r_t \cdot v_t\right], \end{aligned} \tag{2}$$

where the first term $\max_{v\in\mathcal{V}} \sum_{t=1}^T r_t \cdot v$ is the utility of the optimal hindsight pure strategy,[3] and serves as a benchmark. Therefore, this gives a regret minimization formulation with linear *reward* function $r_t \cdot v_t$ where $r_t \in [0, 1]^n$ may be adversarially chosen.

This type of regret notion with optimal fixed strategy is common within the online learning theory literature [5, 6]. The underlying reason is that it is typically impossible to learn the optimal (adaptive) strategy (see [5, 6, 13]). In fact, as the attacker can arbitrarily (and adversarially) change his choice of a_t at each t, the optimal strategy at round t against that a_t can be independent from the history. As such, there is simply no way to predict a_t from the previous observations, and thus, the optimal adaptive strategy cannot be learned.

On the other hand, the best fixed strategy on hindsight can be efficiently learned with access to previous observations. A key intuition behind this is that as we play more and more rounds, no matter how adversarial the attacker will be in the next round, his choice of a_t for that particular round will have less effect on the performance of the best fixed strategy on hindsight, compared to the many previously played rounds.

[3]Notice that there always exists an optimal hindsight pure strategy even we optimize over the set of mixed strategies.

It is worthwhile to note that while the best fixed strategy (on hindsight) is more efficiently learnable, its performance can be arbitrarily bad, compared to that of the optimal adaptive strategy. However, we will show that it is not in our case. Particularly, in the next section we will propose a defender strategy that can achieve both low regret against the best fixed strategy, and provable convergence to a near-optimal adaptive strategy.

3. A LOW-REGRET DEFENCE STRATEGY

In this section, we propose FPL-UE, an FPL-based online learning algorithm for efficiently determining a low-regret defence strategy. To do so, we first brief the main concept of FPL. We then detail the modifications Neu and Bartok introduced to make FPL suitable for combinatorial online learning problems. Finally, we describe FPL-UE. Note that while FPL-UE inherits the main design spirit from FPL, our main contribution is to provide the theoretical guarantee for our setting. In fact, FPL has become an algorithm design concept in online learning literature and there is a family of FPL-based algorithms, which all share similar concept. The key challenges for designing these algorithms lie at the convergence analysis of the algorithm for different settings (which is also the case here).

The FPL algorithm: This type of online learning approach maintains a reward estimate $\widehat{r}_{t,i}$ for each target i and round t, with $\widehat{r}_{1,i} = 0$. Let $\widehat{r}_t$ be the vector of these estimates at round t, and let $z = (z_1, ..., z_n)$ be a random vector such that each $z_i \sim \exp(\eta)$ is independently drawn from the exponential distribution $\exp(\eta)$ with parameter η to be specified. At each round, the algorithm chooses a defender pure strategy v_t

$$v_t = \arg\max_{v \in \mathcal{V}} \{v \cdot (\widehat{r}_t + z)\}, \tag{3}$$

which collects the maximum estimated reward perturbed by the noise vector z. Since z is random, we will view v_t as a random vector as well. After observing the reward $r_{t,i}$ at any chosen target i, the corresponding reward estimates of the chosen target at round $t+1$ can be updated as follows:

$$\widehat{r}_{t+1,i} = \widehat{r}_{t,i} + \frac{r_{t,i}}{p_{t,i}} \mathbb{I}(t,i)$$

where $\mathbb{I}(t,i)$ is an indicator function indicating whether target i was chosen at round t, and $p_{t,i}$ is the probability that target i was chosen within that round. Note that the term $\frac{r_{t,i}}{p_{t,i}}\mathbb{I}(t,i)$ is an unbiased estimator of $r_{t,i}$ (since $\mathbb{E}[\frac{r_{t,i}}{p_{t,i}}\mathbb{I}(t,i)] = r_{t,i}$), and it is more preferred in the online learning literature, compared to the directly observed reward value $r_{t,i}$. This is due to convenience of theoretical analysis. However, while $\mathbb{I}(t,i)$ is fully observable (i.e., we either choose target i or not), $p_{t,i}$ cannot be computed efficiently, as it cannot be expressed in a closed form. To overcome this issue, Neu and Bartok proposed a method, called Geometric Re-sampling (GR), to estimate the value of $1/p_{t,i}$, and can be described as follows (see Algorithm 1):

The GR algorithm: The algorithm is based on the following observation: at round t, $v_{t,i}$ takes value 1 with probability $p_{t,i}$, therefore if we *simulate* strategy v_t, denoted as $\widetilde{v}$ in Algorithm 1, for enough trails, the number of trails needed for $\widetilde{v}_i$ to hit value 1 for the first time is a geometric distribution with mean $1/p_{t,i}$. The GR algorithm precisely follows this observation and estimates $1/p_{t,i}$ by simulating enough trails of $\widetilde{v}$ until $\widetilde{v}_i$ hits 1. However, since there is a positive probability that $\widetilde{v}_i = 1$ will never happen, GR might not ever stop in the worse case, making the algorithm computationally inefficient. To overcome this issue, GR truncates the number of trials with a finite value M, and all the $K(t,i)$ in Algorithm 1, that have not been set yet, will be set to be M (steps 7, 8). This truncation introduces a bias for the estimation of $1/p_{t,i}$. However, the bias can be properly handled (see Lemma 5 in Section 7).

Algorithm 1 The GR Algorithm

Input: $\eta \in \mathbb{R}^+, M \in \mathbb{Z}^+, \widehat{r} \in \mathbb{R}^n, t \in \mathbb{N}$;
Output: $K(t) := \{K(t,1), \ldots, K(t,n)\} \in \mathbb{Z}^n$
1: Initialize $\forall i \in [n] : K(t,i) = 0$, $k = 1$;
2: **for** k=1,2,...,M **do**
3: Repeat step $4 \sim 10$ in Algorithm 2 once just to produce $\widetilde{v}$ as a simulation of v_t.
4: **for all** $i \in [n]$ **do**
5: **if** $k < M$ **and** $\widetilde{v}_i = 1$ **and** $K(t,i) = 0$ **then**
6: Set $K(t,i) = k$;
7: **else if** $k = M$ **and** $K(t,i) = 0$ **then**
8: Set $K(t,i) = M$;
9: **end if**
10: **end for**
11: **if** $K(t,i) > 0$ for all $i \in [n]$, **then** break;
12: **end for**

Algorithm 2 The FPL-UE Algorithm

Parameter: $\eta \in \mathbb{R}^+, M \in \mathbb{Z}^+, \gamma \in [0,1]$;
1: Initialize the estimated reward $\widehat{r} = \mathbf{0} \in R^n$;
2: Pick the set of exploration strategies $E = \{v_1, ..., v_n\}$ such that target i is protected in pure strategy v_i.
3: **for** t=1,...,T **do**
4: Sample $flag \in \{0,1\}$ such that $flag = 0$ with prob. γ;
5: **if** $flag = 0$ **then**
6: Let v_t be a uniform randomly sampled strategy from E;
7: **else**
8: Draw $z_i \sim \exp(\eta)$ independently for $i \in [n]$ and let $z = (z_1, ..., z_n)$;
9: Let $v_t = \arg\max_{v \in \mathcal{V}} \{v \cdot (\widehat{r} + z)\}$;
10: **end if**
11: Adversary picks $r_t \in [0,1]^n$ and defender plays v_t.
12: Run $\mathrm{GR}(\eta, M, \widehat{r}, t)$: estimate $\frac{1}{p_{t,i}}$ as $K(t,i)$;
13: Update $\widehat{r}(i) \leftarrow \widehat{r}(i) + K(t,i) r_{t,i} \mathbb{I}(t,i)$; where $\mathbb{I}(t,i) = 1$ for i satisfying $v_{t,i} = 1$; $\mathbb{I}(t,i) = 0$ otherwise;
14: **end for**

The FPL-UE algorithm: As mentioned earlier, it is not possible to directly apply FPL and GR to our settings. A key reason behind this is that the value of $p_{t,i}$ can be arbitrarily small. While this is not a problem for loss minimisation, it turns out to be a major challenge within our setting. As such, we overcome this issue by introducing additional fraction of uniform exploration to the algorithm. In particular, instead of solely relying on Eq. (3) to determine the defender pure strategy, we uniformly randomly choose a vector v_t from some pre-specified set E with carefully chosen probability $\gamma > 0$, while sets v_t to be the solution of Eq. (3) only with probability $(1-\gamma)$. This seemingly "arbitrary" modification actually allows us to provide effective theoretical analysis of convergence, and interestingly, our extensive simulations also show the necessity of the uniform exploration component – our algorithm outperforms, and in some cases significantly outperforms, the algorithm of Neu and Bartok in the repeated security game setting (see Section 4 and 6 for more details).

Given all these, the FPL-UE algorithm (Algorithm 2) can be described as follows. At each round t, our algorithm either does a uniform random exploration with probability γ (step 6) or plays an FPL strategy with probability $1-\gamma$ (steps 8, 9). Then the algorithm estimates the reward of round t by GR *after* playing the strategy v_t and observing reward (step $11-13$). Notice that, when updating $\widehat{r}$

(step 13), the i'th entry is updated only when $v_{t,i} = 1$, i.e., target i is visited. Otherwise, it keeps unchanged.

4. PERFORMANCE ANALYSIS

Given the description of FPL-UE, we now investigate the theoretical properties of the algorithm. In particular, our main theoretical result is the following guarantee of both computational efficiency and regret bound for FPL-UE.

THEOREM 1. *FPL-UE runs in $poly(n, k, T)$ time if the defender can best respond to any reward vector in $poly(n, k)$ time*[4]*. The regret R_T of FPL-UE is upper bounded as:*

$$R_T \leq \gamma mT + 2Tke^{-M\frac{\gamma}{n}} + \frac{k(\log n + 1)}{\eta} + \eta mT \min(m, k).$$

In particular, with $\eta = \sqrt{\frac{k(\log n+1)}{mT\min\{m,k\}}}$, $\gamma = \frac{\sqrt{k}}{\sqrt{mT}}$ and $M = n\sqrt{\frac{mT}{k}}\log(Tk)$, R_T is at most $\mathcal{O}\left(\sqrt{kmT\min\{m,k\}\log n}\right)$.

The constant of the polynomial is approximately 2. We note that the convergence ratio depends on parameters that are *specific* to security games, for example, the number of attacker resources m. This translates to the upper bound of the sum of the reward vector. Our analysis explores such structure and provides better convergence ratio for security game settings than state-of-the-art algorithms (see Section 6 for more details). As a special case, when $m = n$, this is the general combinatorial adversarial online learning problem for *reward* maximisation with semi-bandit feedback, and for such general settings, FPL-UE achieves regret upper bound $\mathcal{O}\left(k\sqrt{nT\log n}\right)$, which matches the bound for the *loss* case in the state-of-the-art work [19]. We defer the detailed proof to Section 7.

We now investigate the performance of FPL-UE, compared to that of the optimal (adaptive) strategy on hindsight. This is the *best possible* defending strategy one can hope – at each round, the defender can first observe the attacker's move and then play a best response. Let $A = \{a_1, \ldots, a_T\}$ denote any attacker strategy over T rounds. Recall that $a_t \in \{0, 1\}^n$ is the attacker's strategy at round t with $||a_t||_1 \leq m$. Let $OPT(A)$ denote the total *rewards* of the optimal (adaptive) defender strategy on hindsight against A, and let $FPL_(A)$ denote the expected total *rewards* of the defender by applying FPL-UE.[5] As we explained at the end of Section 2, it is generally not possible to provide any theoretical guarantee for $FPL(A)$, when compared with $OPT(A)$. Interestingly, we show that FPL-UE can gain an "almost" $\frac{k}{n}$ fraction of $OPT(A)$ in repeated security game settings.

PROPOSITION 2. *Assuming no schedule constraints, we have*

$$FPL(A) \geq \frac{k}{n}OPT(A) - \mathcal{O}\left(\sqrt{Tmk\min\{k,m\}\log n}\right).$$

Generally, $FPL(A)$ and $OPT(A)$ are of order T, so the term $\mathcal{O}\left(\sqrt{Tmk\min\{k,m\}\log n}\right)$ is relatively negligible. Due to space limitations, all the proofs, *except* for the proof of Theorem 1, are deferred to the online appendix of the paper. In what follows, we will detail a number of implications of these theoretical results, from a game theoretic perspective.

[4]This holds widely in security games.

[5]Note that reward $\neq$ defender utility, due to the extra, uncontrollable and generally negative, term $C(a_t)$ (see Equation 1). We compare the algorithm performance using *rewards*, due to two reasons: 1. $C(a_t)$ is uncontrollable by any algorithm; 2. reward is positive, thus the ratio in Proposition 2 is meaningful.

5. ZERO-SUM SECURITY GAMES

In this section, we consider *zero-sum* security games. Such games can be found in many real-world scenarios, e.g., the illegal fishing monitoring [11]. We first start with the estimation of an approximate algorithmic equilibrium in repeated zero-sum security games. We then investigate how to estimate the minimax value of the game.

Let $V = \{v_1, \ldots, v_T\}$ denote a strategy of the defender over T rounds. Suppose for now that the attacker is also a utility maximiser. Now, consider a pair of defender-attacker strategies (V, A). Let $U_D(V, A)$ denote the expected performance (i.e., the total utility) of the defender strategy V against the attacker strategy A. Note that the expected performance of attacker strategy A against V is $-U_D(V, A)$.

Approximate Algorithmic Equilibrium: A direct implication of Proposition 2 is the estimation of an approximate algorithmic (or program) equilibrium, which is the approximate version of the program equilibrium introduced by Tennenholtz [25]. It can be defined as follows:

DEFINITION 1. *For any $\varepsilon > 0$, the strategy profile (V, A) is a* ε-approximate algorithmic equilibrium *of the game if and only if for any V' and A', we have $U_D(V', A) \leq U_D(V, A) + \varepsilon$ and $U_D(V, A) \leq U_D(V, A') + \varepsilon$.*

Consider the case when both the defender and the attacker applies FPL-UE to choose their actions. Let

$$\varepsilon = \frac{n-k}{n}mT + 3\sqrt{kmT\min\{m,k\}\log n} \tag{4}$$

COROLLARY 3. *Assume $\min\{k, m\}\log n \geq 3$ and no schedule constraints. The strategy profile* (FPL-UE, FPL-UE) *(i.e., both players use FPL-UE) is an ε-approximate algorithmic equilibrium of the repeated zero-sum security game, where ε is defined as in Eq.* (4)*.*

Minimax Value of the Game: In zero-sum games, the minimax (or maxmin) value of the game is a powerful solution concept, which provides a guarantee we can achieve even in the worst case scenario, and it is well known that the minimax value exists and can be efficiently computed, e.g., by linear programming [9]. In particular, let $\mathbf{v}/\mathbf{a}$ denote a defender/attacker *mixed* strategy, and $\mathbf{u}^*$ denote the minimax value of the zero-sum security game. From von Neumann's Minimax Theorem, we have $\mathbf{u}^* = \max_{\mathbf{v}}\min_{\mathbf{a}} u(\mathbf{v}, \mathbf{a}) = \min_{\mathbf{a}}\max_{\mathbf{v}} u(\mathbf{v}, \mathbf{a})$. However, the minimax value *cannot* be calculated without having the full knowledge of the game, which unfortunately is indeed the case in our setting. Nevertheless, we can approximate this value by the FPL-UE strategy.

COROLLARY 4. *Assume $\min\{k, m\}\log n \geq 3$. Suppose that both the defender and attacker apply FPL-UE to make their actions. We have:*

$$\left|\mathbf{u}^* - \frac{U_D(\text{FPL},\text{FPL})}{T}\right| \leq 3\sqrt{\frac{km\min\{m,k\}(\log n+1)}{T}}$$

where $U_D(\text{FPL}, \text{FPL})$ denotes the utility of the defender applying FPL-UE against a FPL-UE attacker.

That is, to estimate the mimimax value of the game, we just need to simulate a game against an FPL-UE attacker and consider the average utility. Indeed, as T tends to infinity, the approximation gap converges to 0. Note that since the simulated attacker is not a real one, we still do not need to have any prior knowledge about real attackers. In fact, we only consider a simulated attacker, whose actions can be fully simulated. On the other hand, we require the knowledge of the payoff matrix (i.e., what is the payoff of each

action pairs of the players) to calculate the minimax value of the game. This assumption, however, is reasonable, as in the zero-sum games, the payoff of the attacker is negative version of the defender's payoff. Thus, this assumption does not require prior knowledge about the attacker either.

6. NUMERICAL EVALUATIONS

We run our algorithm against a number of commonly seen attacker models with simulations. In our simulations, the payoffs U_i^c's and U_i^u's are randomly generated. In particular, for any target $i \in [n]$, we first drawn two numbers $a, b \in [-0.5, 0.5]$ uniformly at random, and then set $U_i^c = \max(a, b)$ and $U_i^u = \min(a, b)$, thus the condition $U_i^c \geq U_i^u$ is satisfied. Note that the defender has no a-priori knowledge about these payoffs.

We test our algorithm against different types of attackers, which together represent the majority of typical attacking models. Purely for the purpose of modeling the attacker's reaction to our algorithm, we also generate the attacker's payoffs in a similar fashion as the defender's payoffs, except that the attacker's payoff is higher if a target is *uncovered*. We consider 5 different types of attackers:[6]

- **Uniform**: an attacker with a uniformly random *mixed* strategy;
- **Adversarial**: an attacker with the maximin *mixed* strategy. That is, the attacker is fully adversarial – he only cares about minimizing the defender's utility;
- **Stackelberg**: the attacker always plays the optimal follower *pure* strategy of the Strong Stackelberg Equilibrium;
- **BestResponse**: at round t, the attacker best responds to the mixed strategy $\sum_{i=1}^{t-1} v_i/(t-1)$, i.e., the empirical defender mixed strategy in history;
- **QuantalResponse (QR)**: the attacker also responds to the empirical defender mixed strategy, but by a QR model [21].

Note that some attacker types are informationally very powerful, e.g., the *BestResponse* type knows all the payoffs as well as all the defender's past actions, while some may have very little knowledge (e.g., the *Uniform* type); Some types play mixed strategies (e.g., *Adversarial* type) while some play pure strategies (e.g., *Stackelberg* type); Some types are rational while some are not. Also note that our algorithm does not know which type of attackers we are facing. The simulation aims to test the "robustness" of the algorithm against varied types of attackers.

Baselines: We choose two algorithms as baselines. The first is the **FPL** algorithm of Neu and Bartok [19], a *state-of-the-art algorithm* for combinatorial adversarial online learning. As Neu and Bartok pointed out, the regret bound for their FPL algorithm can only be proved in the cost minimization scenario. Nevertheless, we use it as a baseline to see how it compares to FPL-UE and how it performs in repeated security games where the goal is to maximize reward. Another baseline is the Subjective Utility Quantal Response (**SUQR**) attacker behavior model [21], a *state-of-the-art human behavior model* in security games that captures the attacker's bounded rationality. SUQR model has been tested in several human behavior experiments of security games in Amazon Mechanical Turk (AMT), and is shown to outperform the strong Stackelberg equilibrium strategy and maximin strategy when playing against real-world humans [21, 8, 14]. In our simulation, at each round T, the SUQR model first looks at all the attack records from the past $T-1$ rounds and then learns the attacker's behavior model based on these *history records* and the *attacker's payoffs*. As a result, the SUQR model will be refined each round with more history records. After learning the attacker's SUQR behavior model, the defender then computes an optimal mixed strategy against the behavior model and samples a pure strategy to play.

We note that it is *not* completely fair to compare FPL-UE with SUQR because SUQR requires much more defender prior knowledge (e.g., past attack records and attacker payoffs) than FPL-UE. Nevertheless, we aim to examine how competitive FPL-UE is when compared with SUQR.

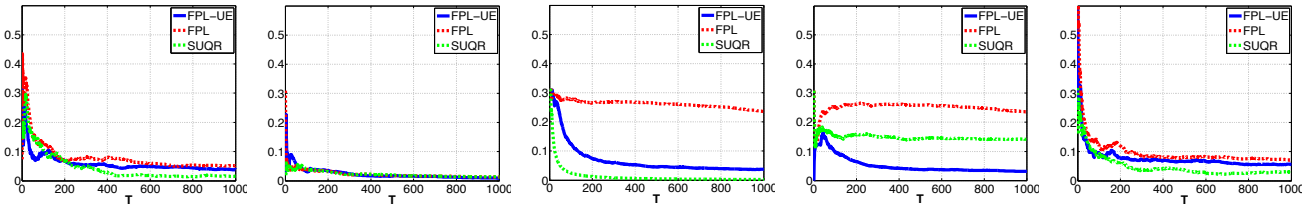
We set $n = 100$ and $k = 10$ in all our simulations, and test the convergence of the *average regret* (i.e., regret divided by T) for various m. We translate the defender utility of SUQR to regret (See Equation (2)). Figures 2, 3 and 4 show the case for $m = 1$, $m = 5$, and $m = 15$ respectively, within 1000 rounds. Note that average regret is upper-bounded by m. All these figures are the convergence plots for *one* randomly generated game instance, however we do emphasize that the general convergence trend is almost the same across the simulated instances except that the initial rounds in the figures may vary among different instances. Since the absolute value of regret at a fixed round T differs across different game instances, so averaging the regret over games destroys the convergence lines. So we only present one randomly chosen instance here. As an interesting side note, when $m = 15$, the defender has less resources than the attacker. To our knowledge, experiments for such cases have not been done before. We use it as a burden test for the robustness of our algorithm.

From the figures we know that FPL-UE converges in all these cases, while FPL fails to converge (at least within 1000 rounds) when played against *Stackelberg* type and *BestResponse* type. These figures clearly show that FPL-UE outperforms FPL. One interesting phenomenon is that FPL-UE always significantly outperforms FPL when playing against *Stackelberg* type and *BestResponse* type in all the instances we generated. Notice that these two cases are the "difficult" cases for online learning algorithms. The *Stackelberg* type always plays the same *pure* strategy over the whole game, therefore the best hindsight strategy is to protect the most valuable attacked targets, achieving a very high reward. Thus the algorithm takes longer time to converge, mainly due to the compensation for the big loss at the initial rounds where exploration happens mostly. While for the *BestResponse* type, the attacker is always adaptive to the algorithm. The comparison on these "difficult" types shows the advantage of FPL-UE over FPL.

One surprising observation is that, though requiring much more defender prior knowledge, SUQR does not obviously outperform FPL-UE. In fact, SUQR only weakly outperforms FPL-UE when playing against the *Uniform* type and *QuantalResponse* type. This is natural because these two types exactly lie at the realm of the SUQR model. For all the other three types, SUQR does not exhibit obvious advantage. In fact, when the attacker is totally adaptive, i.e., the *BestResponse* type, FPL-UE actually shows some weak advantage. We attribute this to the carefully designed adaptivity nature of the FPL-UE algorithm.

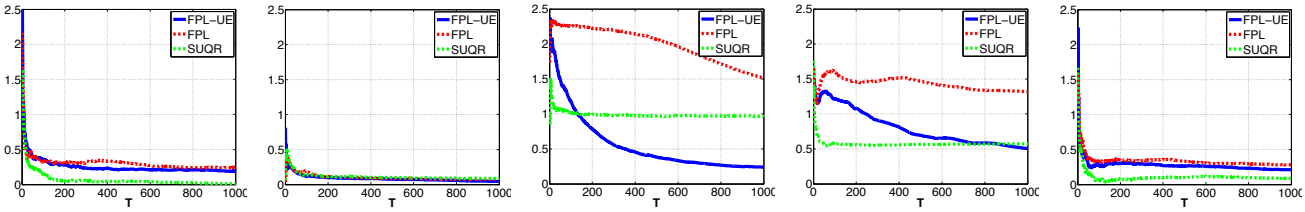
Finally, we observe that the regret against the *BestResponse* or *Stackelberg* type (the two difficult cases) with $m = 15$ is about 1.2 which approximates the regret upper bound ($\frac{m\sqrt{kT\log n}}{T} \approx 1.8$). Interestingly, this empirically shows that the algorithm approximates the upper bound regret when played in these hard cases. To summarize, depending on the rationality level, attacker strategy type and the amount of information the attacker has about the past games, the algorithm can converge at different rates, but will

[6] In the following description, if the attacker plays a *mixed* strategy, our simulation samples a *pure* strategy each round from the described mixed strategy to play.



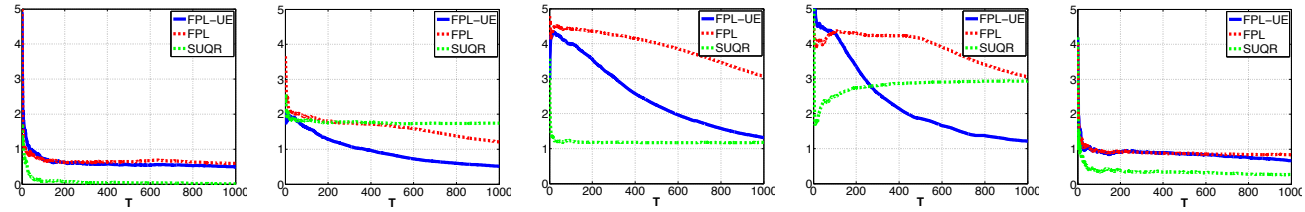
(a) Uniform (b) Adversarial (c) Stackelberg (d) BestResponse (e) QuantalResponse (QR)

Figure 2: Average regret (y-axis) against 5 attacker types after T rounds; Parameters: $n = 100, k = 10, \mathbf{m = 1}, \lambda = 2$ for *QR*.



(a) Uniform (b) Adversarial (c) Stackelberg (d) BestResponse (e) QuantalResponse (QR)

Figure 3: Average regret (y-axis) against 5 attacker types after T rounds; Parameters: $n = 100$, $k = 10$, $\mathbf{m = 5}$, $\lambda = 2$ for *QR*.



(a) Uniform (b) Adversarial (c) Stackelberg (d) BestResponse (e) QuantalResponse (QR)

Figure 4: Average regret (y-axis) against 5 attacker types after T rounds; Parameters: $n = 100$, $k = 10$, $\mathbf{m = 15}$, $\lambda = 2$ for *QR*.

always be upper bounded by the regret bound.

7. PROOF OF THEOREM 1

We now turn to prove Theorem 1. To do so, we first describe the following lemmas. Lemma 5, as proved in [19], captures the estimation bias of Geometric Re-sampling.

LEMMA 5. *[19]* $\mathbb{E}(\widehat{r}_{t,j}|\mathcal{F}_{t-1}) = (1-(1-p_{t,j})^M)r_{t,j}$.

In addition, the following lemma is a simple observation about the reward vector r_t of round t.

LEMMA 6. $||r_t||_1 \leq m$ *for any* t.

PROOF. Since $r_{t,i} \geq 0$, we have

$$||r_t||_1 = \sum_{i\in[n]} r_{t,i} = \sum_{i\in[n]} a_{t,i}[U_i^c - U_i^u] \leq \sum_{i\in[n]} a_{t,i} \leq m.$$

□

To analyse the algorithm, we first describe some notations. Let $\widehat{r}_t$, where $\widehat{r}_{t,i} = K(t,i)r_{t,i}\mathbb{I}(t,i) = K(t,i)r_{t,i}v_{t,i}$, denote the estimation of the reward at round t. Let

$$v_t^{FPL} = \arg\max_{v\in\mathcal{V}} v \cdot (\sum_{j=1}^{t-1} \widehat{r}_j + z), \tag{5}$$

denote the FPL strategy played *before* estimating $\widehat{r}_t$ (Step 9 in Algorithm 1). For the purpose of analysis, define the following *hindsight strategy* (imagine the defender can get the estimation $\widehat{r}_t$ *before* she plays a strategy at round t)

$$\widetilde{v}_t^{FPL} = \arg\max_{v\in\mathcal{V}} v \cdot (\sum_{j=1}^{t} \widehat{r}_j + z). \tag{6}$$

Therefore, we have $v_{t+1}^{FPL} \overset{st}{=} \widetilde{v}_t^{FPL}$ where "$\overset{st}{=}$" means stochastically equal due to the randomness of z.[7] Let $q_{t,i} = \mathbb{E}(v_{t,i}^{FPL})$ be the probability that target i is protected in strategy v_t^{FPL}, and $\widetilde{q}_{t,i} = \mathbb{E}(\widetilde{v}_{t,i}^{FPL})$ be the probability that target i is protected in strategy $\widetilde{v}_t^{FPL}$. Therefore $q_{t+1,i} = \widetilde{q}_{t,i}$. Notice that v_t^{FPL} is not the only possible strategy played at round t – with probability γ, the defender plays a uniformly randomly sampled pure strategy from set E. Let $p_{t,i} = \mathbb{E}(v_{t,i})$ denote the probability that target i is protected at time t in Algorithm 1. Due to the γ fraction of uniform exploration, we have $p_{t,i} \geq \frac{\gamma}{n}$ for any t, i since each target is protected by at least one pure strategy in E.

Now, we are ready to prove the regret upper bound of Algorithm 1. As observed, the algorithm does exploration with probability γ and exploitation with probability $1-\gamma$. We start from analyzing the exploitation part. Using the "be-the-leader" lemma [5] to sequence $(\widehat{r}_1 + z, \widehat{r}_2, ..., \widehat{r}_T)$, we obtain

$$\sum_{t=1}^{T} \widehat{r}_t \cdot \widetilde{v}_t^{FPL} + z \cdot \widetilde{v}_1^{FPL} \geq \sum_{t=1}^{T} \widehat{r}_t \cdot v + z \cdot v, \forall v \in \mathcal{V}, \tag{7}$$

where $\widetilde{v}_t^{FPL}$ is defined in Equation (6). By rearranging Inequality

[7]Recall that, two random variable A, B are stochastically equal if $\mathbb{P}(A = x) = \mathbb{P}(B = x)$ for any x in the event set.

(7), we have

$$\begin{aligned}\mathbb{E}\left[\sum_{t=1}^{T}\widehat{r}_t\cdot(v-\widetilde{v}_t^{FPL})\right] &\leq \mathbb{E}\left[z\cdot(\widetilde{v}_1^{FPL}-v)\right]\\ &\leq \mathbb{E}\left[z\cdot\widetilde{v}_1^{FPL}\right]\\ &\leq k\mathbb{E}(\max_{i\in[n]}z_i)\\ &\leq \frac{k(\log n+1)}{\eta} \qquad (8)\end{aligned}$$

where the second last "$\leq$" uses the inequality $\sum_{i\in[n]}\widetilde{v}_{1,i}^{FPL}\leq k$; the last "$\leq$" uses the fact that $\mathbb{E}(\max_{i\in[n]}z_i)\leq\frac{\log n+1}{\eta}$.

The rest of the proof lies on the following basic intuitions. First, the played strategy v_t^{FPL} should not be too "far" from the hindsight strategy $\widetilde{v}_t^{FPL}$ since the reward estimations they used are only slightly different (compare Equation (5) and (6)). Second, the estimated reward $\widehat{r}_t$ hopefully is "close" to the real reward r_t. Therefore, Inequality (8) also roughly conveys that the played strategy sequence v_t^{FPL} is not too "bad" compared with *any* pure strategy v in the scenario of real reward r_t.

We first lower bound $q_{t,i}$ using $\widetilde{q}_{t,i}$ as follows.

$$\begin{aligned}q_{t,i} &= \int_{z\in[0,\infty]^n}v_{t,i}^{FPL}(z)f(z)dz\\ &= e^{-\eta||\widehat{r}_t||_1}\int_{z\in[0,\infty]^n}v_{t,i}^{FPL}(z)f(z-\widehat{r}_t)dz\\ &= e^{-\eta||\widehat{r}_t||_1}\int\cdots\int_{z_i\in[-\widehat{r}_{t,i},\infty]}v_{t,i}^{FPL}(z+\widehat{r}_t)f(z)dz\\ &= e^{-\eta||\widehat{r}_t||_1}\int\cdots\int_{z_i\in[-\widehat{r}_{t,i},\infty]}\widetilde{v}_{t,i}^{FPL}(z)f(z)dz\\ &\geq e^{-\eta||\widehat{r}_t||_1}\int\cdots\int_{z_i\in[0,\infty]}\widetilde{v}_{t,i}^{FPL}(z)f(z)dz\\ &= e^{-\eta||\widehat{r}_t||_1}\widetilde{q}_{t,i}\\ &\geq (1-\eta m)\widetilde{q}_{t,i}\end{aligned}$$

where the last inequality uses Lemma 6: $||\widehat{r}_t||_1\leq m$ for any t. Now we bound the difference between $\mathbb{E}\left[\widehat{r}_t\cdot v_t^{FPL}\right]$ and $\mathbb{E}\left[\widehat{r}_t\cdot\widetilde{v}_t^{FPL}\right]$.

$$\begin{aligned}\mathbb{E}\left[\widehat{r}_t\cdot v_t^{FPL}|\mathcal{F}_t\right] &= \sum_{i\in[n]}\widehat{r}_{t,i}q_{t,i}\\ &\geq \sum_{i\in[n]}\widehat{r}_{t,i}\widetilde{q}_{t,i}-\eta m\sum_{i\in[n]}\widehat{r}_{t,i}\widetilde{q}_{t,i}\\ &\geq \mathbb{E}\left[\widehat{r}_t\cdot\widetilde{v}_t^{FPL}|\mathcal{F}_t\right]-\eta m\min(m,k)\end{aligned}$$

where we used the fact that $\widehat{r}_t$ and $\widehat{r}_{t-1}$ are fixed given $\mathcal{F}_t$, therefore the randomness of v_t^{FPL} and $\widetilde{v}_t^{FPL}$ only comes from z. Taking expectation over $\mathcal{F}_t$, we have

$$\mathbb{E}\left[\widehat{r}_t\cdot v_t^{FPL}\right]\geq\mathbb{E}\left[\widehat{r}_t\cdot\widetilde{v}_t^{FPL}\right]-\eta m\min(m,k) \qquad (9)$$

Now we can upper bound the loss from substituting $\widetilde{v}_t^{FPL}$ in Inequality (8) by v_t^{FPL}.

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^{T}\widehat{r}_t\cdot(v-v_t^{FPL})\right]\\ =\ &\mathbb{E}\left[\sum_{t=1}^{T}\widehat{r}_t\cdot(v-\widetilde{v}_t^{FPL})\right]+\mathbb{E}\left[\sum_{t=1}^{T}\widehat{r}_t\cdot(\widetilde{v}_t^{FPL}-v_t^{FPL})\right]\\ \leq\ &\frac{k(\log n+1)}{\eta}+\eta mT\min(m,k),\end{aligned}$$

where the "$\leq$" is due to the Inequality (8) and (9). Putting all these together, we can upper bound the regret from the exploitation part:

$$\begin{aligned}&\mathbb{E}\left[\sum_{t=1}^{T}r_t\cdot(v-v_t^{FPL})\right]\\ \leq\ &\mathbb{E}\left[\sum_{t=1}^{T}(r_t-\widehat{r}_t)\cdot(v-v_t^{FPL})\right]+\mathbb{E}\left[\sum_{t=1}^{T}\widehat{r}_t\cdot(v-v_t^{FPL})\right]\\ \leq\ &2Tk(1-\frac{\gamma}{n})^M+\frac{k(\log n+1)}{\eta}+\eta mT\min(m,k)\end{aligned}$$

Now configuring the regret from exploration, which is trivially upper bounded by m and happens with probability γ, we can upper-bound the total regret as

$$\begin{aligned}R_T &\leq \gamma mT+(1-\gamma)\Big[2Tk(1-\frac{\gamma}{n})^M\\ &\quad+\frac{k(\log n+1)}{\eta}+\eta mT\min(m,k)\Big]\\ &\leq \gamma mT+2Tke^{-M\frac{\gamma}{n}}\\ &\quad+\frac{k(\log n+1)}{\eta}+\eta mT\min(m,k)\end{aligned}$$

By taking $\eta=\sqrt{\frac{k(\log n+1)}{mT\min\{m,k\}}}$, $\gamma=\frac{\sqrt{k}}{\sqrt{mT}}$ and $M=n\sqrt{\frac{mT}{k}}\log(Tk)$, we obtain the upper bound $\mathcal{O}\left(\sqrt{kmT\min\{m,k\}\log n}\right)$.

8. CONCLUSIONS

In this paper we proposed FPL-UE, the first defender strategy for repeated security games assuming no prior knowledge about the attacker. We proved that our algorithm enjoys a number of compelling theoretical properties. In particular, we showed that FPL-UE can provably achieve low regret bounds, against both the best fixed strategy on hindsight, and the optimal adaptive strategy. In addition, we proved that our main theoretical results have a number of game theoretic implications, such as the efficient estimation of algorithmic equilibria and the minimax value of the game (for zero-sum security games). Our numerical evaluations demonstrated that FPL-UE is indeed efficient against typical attacker profiles. We also demonstrated that FPL-UE indeed outperforms the FPL version of Neu and Bartok. This justifies the usage of the additional uniform exploration steps. Furthermore, its performance is comparable to that of SUQR, a state of the art of the repeated security games literature. This result is surprising and significant, as our algorithm does not require any prior knowledge of the attacker, while SUQR relies much on the existence of such prior information. This implies that, our algorithm is very useful in real-world situations where the attacker behaviour model is not available at the beginning, or the attacker does not fully follow some rational behaviour model. Given this, we argue that our algorithm is more generic, compared to the state of the art, and thus, can be applied in many realistic scenarios of repeated security games.

Acknowledgement:

Xu is supported by MURI grant W911NF-11-1-0332 and NSF grant CCF- 1350900. Tran-Thanh and Jennings gratefully acknowledge funding from the UK Research Council for project "ORCHID", grant EP/I011587/1.

REFERENCES

[1] B. An, D. Kempe, C. Kiekintveld, E. Shieh, S. Singh, M. Tambe, and Y. Vorobeychik. Security games with limited surveillance. In *Conference on Artificial Intelligence (AAAI 2012)*, 2012.

[2] M.-F. Balcan, A. Blum, N. Haghtalab, and A. D. Procaccia. Commitment without regrets: Online learning in stackelberg security games. In *Proceedings of the Sixteenth ACM Conference on Economics and Computation*, EC 2015, pages 61–78, New York, NY, USA, 2015. ACM.

[3] A. Blum, N. Haghtalab, and A. D. Procaccia. Learning optimal commitment to overcome insecurity. In Z. Ghahramani, M. Welling, C. Cortes, N. Lawrence, and K. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 1826–1834. Curran Associates, Inc., 2014.

[4] M. Brown, W. B. Haskell, and M. Tambe. Addressing scalability and robustness in security games with multiple boundedly rational adversaries. In *Conference on Decision and Game Theory for Security (GameSec)*, 2014.

[5] N. Cesa-Bianchi and G. Lugosi. *Prediction, Learning, and Games*. Cambridge University Press, New York, NY, USA, 2006.

[6] N. Cesa-Bianchi and G. Lugosi. Combinatorial bandits. *Journal of Computer and System Sciences*, 78(5):1404–1422, 2012.

[7] F. Fang, P. Stone, and 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)*, 2015.

[8] B. Ford, T. Nguyen, M. Tambe, N. Sintov, and 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*, 2015.

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

[10] Y. Gai, B. Krishnamachari, and R. Jain. Combinatorial network optimization with unknown variables: Multi-armed bandits with linear rewards and individual observations. *IEEE ACM Trans. Netw.*, 20 (5):1466–1478, 2012.

[11] W. B. Haskell, D. Kar, F. Fang, M. Tambe, S. Cheung, and E. Denicola. Robust protection of fisheries with compass. In *Proceedings of the Twenty-Eighth AAAI Conference on Artificial Intelligence, July 27 -31, 2014, Québec City, Québec, Canada.*, pages 2978–2983, 2014.

[12] M. Jain, J. Tsai, J. Pita, C. Kiekintveld, S. Rathi, M. Tambe, and F. Ordóñez. Software assistants for randomized patrol planning for the lax airport police and the federal air marshal service. *Interfaces*, 40(4):267–290, July 2010.

[13] A. Kalai and S. Vempala. Efficient algorithms for online decision problems. *J. Comput. Syst. Sci.*, 71(3):291–307, Oct. 2005.

[14] D. Kar, F. Fang, F. D. Fave, N. Sintov, and M. Tambe. Şa game of thronesŤ: When human behavior models compete in repeated stackelberg security games. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2015)*, 2015.

[15] C. Kiekintveld, J. Marecki, and M. Tambe. Approximation methods for infinite bayesian stackelberg games: Modeling distributional payoff uncertainty. In *The 10th International Conference on Autonomous Agents and Multiagent Systems - Volume 3*, AAMAS 2011, pages 1005–1012, Richland, SC, 2011. International Foundation for Autonomous Agents and Multiagent Systems.

[16] R. Klima, C. Kiekintveld, and V. Lisy. Online learning methods for border patrol resource allocation. In *GameSec*, 2014.

[17] R. Klima, V. Lisy, and C. Kiekintveld. Combining online learning and equilibrium computation in security games. In *GameSec*, 2015.

[18] B. Kveton, Z. Wen, A. Ashkan, and C. Szepesvári. Tight regret bounds for stochastic combinatorial semi-bandits. *AISTAT*, 2015.

[19] G. Neu and G. Bartók. An efficient algorithm for learning with semi-bandit feedback. *ALT*, pages 234–248, 2013.

[20] G. Neu and G. Bartók. Importance weighting without importance weights: An efficient algorithm for combinatorial semi-bandits. *CoRR*, abs/1503.05087, 2015.

[21] T. H. Nguyen, R. Yang, A. Azaria, S. Kraus, and M. Tambe. Analyzing the effectiveness of adversary modeling in security games. In *In Conf. on Artificial Intelligence (AAAI 2013)*, 2013.

[22] J. Pita, M. Jain, J. Marecki, F. Ordóñez, C. Portway, M. Tambe, C. Western, P. Paruchuri, and S. Kraus. Deployed armor protection: The application of a game theoretic model for security at the los angeles international airport. In *Proceedings of the 7th International Joint Conference on Autonomous Agents and Multiagent Systems: Industrial Track*, AAMAS 2008, pages 125–132, Richland, SC, 2008. International Foundation for Autonomous Agents and Multiagent Systems.

[23] J. Pita, M. Jain, M. Tambe, F. Ordóñez, and S. Kraus. Robust solutions to stackelberg games: Addressing bounded rationality and limited observations in human cognition. *Artificial Intelligence*, 174(15):1142 – 1171, 2010.

[24] M. Tambe. *Security and Game Theory: Algorithms, Deployed Systems, Lessons Learned*. Cambridge University Press, New York, NY, USA, 1st edition, 2011.

[25] M. Tennenholtz. Program equilibrium. *Games and Economic Behavior*, 49(2):363–373, 2004.

[26] R. Yang, B. Ford, M. Tambe, and A. Lemieux. Adaptive resource allocation for wildlife protection against illegal poachers. In *International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2014)*, 2014.