

Stackelberg Security Games

Tom Morgan, Varun Sriram



Objectives

1. Background on Stackelberg Security Games
2. Repeated Security Stackelberg Game Model
3. Online Setting for Stackelberg Games
4. Understanding Results:
 - a. Full Information Upper Bound
 - b. Partial Information Upper Bound
 - c. Lower Bound

Stackelberg Security Games

- Two players: **attacker** and **defender**
- Defender first commits to a mixed strategy and then attacker plays in response
- Attacker chooses one of **n targets** to attack
- Defender has a set of **resources R**, each of which has a set of possible **schedules D** each of which is a subset of $\{1, \dots, n\}$
- A **pure defense strategy** is a mapping of resources to schedules, resulting in a subset of the targets which are covered
- A **mixed defense strategy** assigns each target a probability p_i of being covered
- Attacker and defender each have a utility function which takes as input **which target** was attacked, and **whether or not it was covered**

$$U_d(i, \mathbf{p}) = u_d^c(i)p_i + u_d^u(i)(1 - p_i)$$

Stackelberg Security Game Applications

- Milind Tambe's group at USC:
 - Airport security
 - Poaching countermeasures
 - Coast guard anti-terrorism for ports
 - Allocation of air marshals
 - Fare checking in LA & Chicago Metros
- “The most widely used application of game theory without money” - Yiling & Bo paraphrased, 2016



Online Setting - Review

- T = time horizon; M = the finite set of decisions/actions/experts to choose from; $N = |M|$
- Players: Online Learner and Adaptive Adversary
- For every t , the learner chooses a distribution D over the actions in M and picks an action from D : This is the **learning algorithm**
- The adversary best responds and the learner incurs a loss based on that response

Online Setting - Stackelberg Model

- T = time horizon; M = the finite set of mixed strategies to choose from; $N = |M|$; n = # targets
- Players: Defender and k Different Attackers (the k types are known to defender)
- For every t , the defender chooses a distribution D over the mixed strategies in M and picks a mixed strategy $p_t \in R^n$ from D : This is the **learning algorithm**
- If $p_i \in \{0, 1\}$, we call p a **pure strategy**, if $0 \leq p_i \leq 1$, we call p a **mixed strategy**
- At time t , the adversarially chosen attacker best responds (hits target $b_{a_t}(p) = \arg \max_i U_{a_t}(i, p)$) and the defender incurs a loss ($U_d(b_{a_t}(p_t), p_t)$) based on that attacker and his response
- When the defender knows the sequence of attackers up to $(t-1)$, the learning algorithm is **full information**, and when the defender knows only the attacker's best responses up to $(t-1)$, the learning algorithm is **partial information**

Discussion: Do you see any issues with this setup?

Online Setting - Narrowing Space of Mixed Strategies

Definition 4.1: $P_i^j = \{p \in \text{mixed strategy space s.t. } b_{a_j}(p) = i\}$

(1) This is a convex polytope. (2) By considering only extreme points of all these convex polytopes, we incur minimal additional loss in the regret bound.

Lemma 4.4: In a repeated security game with n targets and k attacker types,

$$|M| \in O((2^n + kn^2)^n n^k)$$

Full Information Result - Regret (1)

- L_{alg} or U_{alg} = loss/utility incurred by learner over T by following his learning algorithm
- L_{min} or U_{max} = biggest utility/ smallest loss incurred by adopting best fixed action in hindsight
- $\text{Regret} = L_{\text{alg}} - L_{\text{min}} = U_{\text{max}} - U_{\text{alg}}$
- $U_{\text{max}} = \sum_{t=1}^T U_d(b_{a_t}(p^*), p^*)$; $U_{\text{alg}} = \sum_{t=1}^T U_d(b_{a_t}(p_t), p_t)$; $p^* = \arg \max_p \sum_{t=1}^T U_d(b_{a_t}(p), p)$
- **No Regret** means average regret $(1/T * \text{Regret})$ goes to 0 as $T \rightarrow \infty$: that is, R must be sublinear in T
- PROP 1: \exists class of no regret **learning algorithms** st $R \leq \text{Sqrt}(T \log N)$

Discussion: What is an example of a learning algorithm in above class?

Full Information Result - Regret (2)

Theorem 5.1:

$$U_{\max} - U_{\text{alg}} \leq O(\sqrt{Tn^2k \log nk})$$

Partial Information Result

THEOREM 6.1. *Given a repeated security game with partial information feedback, n targets, k attacker types, and time horizon T , there is an online algorithm that for any unknown sequence of attackers, \mathbf{a} , at time t plays a randomly chosen mixed strategy \mathbf{p}_t , and has the property that*

$$\mathbb{E} \left[\sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}_t), \mathbf{p}_t) \right] \geq \sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}^*), \mathbf{p}^*) - O \left(T^{2/3} nk \log^{1/3}(nk) \right),$$

where the expectation is taken over algorithm's internal randomization.

Partial Information Algorithm

- Balance **exploration** with **exploitation**
- Explore by using an **unbiased estimator** of the loss
- Divide T into blocks, for each block:
 - For some (randomly chosen) timesteps in the block, **explore** by evaluating the unbiased estimator
 - Otherwise, **exploit** by using the full information algorithm with the information gathered by the previous block's loss estimation

Unbiased Estimators

- Need to be able to **estimate the loss of any mixed strategy**
- Bad idea: try every mixed strategy
- Sufficient to estimate the **frequency of each attacker type**
- Easy if there was one mixed strategy which every attacker type responded to differently
- Use **barycentric spanners** to find a set of k mixed strategies from which we can infer attacker type frequency

Partial Information Algorithm

- (1) $Z \leftarrow n (T^2 \log nk)^{1/3}$
- (2) Find a barycentric spanner $\mathcal{B} = \{\mathbf{b}_1, \dots, \mathbf{b}_k\}$
- (3) Let \mathbf{q}_1 be the uniform distribution over \mathcal{E}
- (4) For $\tau = 1, \dots, Z$
 - (a) Choose a random permutation π over $[k]$ and t_1, \dots, t_k time steps at random from $[T/Z]$
 - (b) For $t = (\tau - 1)(T/Z) + 1, \dots, \tau(T/Z)$
 - i. If $t = t_j$ for some $j \in [k]$, then $\mathbf{p}_t \leftarrow \mathbf{p}_{\mathbf{b}_{\pi(j)}}$
 - ii. Else, draw \mathbf{p}_t at random from distribution \mathbf{q}_τ
 - (c) For all $\mathbf{p} \in \mathcal{E}$, for σ such that $\mathbf{p} \in \mathcal{P}_\sigma$

$$\hat{c}_\tau(\mathbf{p}) \leftarrow \sum_{i=1}^n \sum_{j=1}^k \lambda_j(\mathbb{I}_{\sigma=i}) \hat{p}_\tau(\mathbf{b}_j) U_d(i, \mathbf{p})$$

- (d) Call FULL-INFORMATION(\hat{c}_τ). And receive $\mathbf{q}_{\tau+1}$ as a distribution over all mixed strategies in \mathcal{E}

Partial Information Result

THEOREM 6.1. *Given a repeated security game with partial information feedback, n targets, k attacker types, and time horizon T , there is an online algorithm that for any unknown sequence of attackers, \mathbf{a} , at time t plays a randomly chosen mixed strategy \mathbf{p}_t , and has the property that*

$$\mathbb{E} \left[\sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}_t), \mathbf{p}_t) \right] \geq \sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}^*), \mathbf{p}^*) - O \left(T^{2/3} nk \log^{1/3}(nk) \right),$$

where the expectation is taken over algorithm's internal randomization.

Discussion Break

How do the full information and partial information results compare in:

- Practicality of model
- Practicality of algorithm
- Adversarial model

Lower Bound

THEOREM 7.1. *For any T there is a repeated security game in the full-information feedback setting with a set K of attacker types such that $|K| < 2^{T+1}$ and any online algorithm that at time t (possibly at random) returns strategy \mathbf{p}_t has expected utility*

$$\mathbb{E} \left[\sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}_t), \mathbf{p}_t) \right] \leq \sum_{t=1}^T U_d(b_{a_t}(\mathbf{p}^*), \mathbf{p}^*) - \frac{T}{2},$$

where the expectation is over the algorithm's internal randomization.

Lower Bound Proof

- There are $n = 3$ targets
- Regardless of coverage, defender gets:
 - 0 utility for target 2
 - -1 utility for targets 1 and 3
- Defender can cover any one target
- Defender prefers 2 get attacked, so never covers it: coverage probabilities are $(p, 0, 1-p)$
- Construct attackers with two parameters $0 \leq a \leq b \leq 1$, best response is:
 - 1 if $p \in [0, a]$
 - 2 if $p \in (a, b]$
 - 3 if $p \in (b, 1]$
- (See board for the rest)

Open Discussion