Manipulating a Learning Defender and Ways to Counteract

Jiarui Gan[1], Qingyu Guo[2], Long Tran-Thanh[3], Bo An[2], Michael Wooldridge[1]

[1]University of Oxford, [2]Nanyang Technological University, [3]University of Southampton

1 Learning Optimal Commitment

Stackelberg Security Game (SSG)

- **Defender**: allocate m **resources** to protect n **targets**
  → coverage $\mathbf{c} = (c_1, \dots, c_n)$, $c_i \in [0,1]$, $\sum_i c_i \le m$
- **Attacker**: select a target $i \in T = \{1, \dots, n\}$ to attack

Utilities: $u^{\mathrm{d}}(\mathbf{c}, i) = c_i \cdot r_i^{\mathrm{d}} + (1 - c_i) \cdot p_i^{\mathrm{d}}$

$u^{\mathrm{a}}(\mathbf{c}, i) = (1 - c_i) \cdot r_i^{\mathrm{a}} + c_i \cdot p_i^{\mathrm{a}}$

Strong Stackelberg equilibrium (SSE):

- Optimal defender commitment assuming best attacker response
- $(\hat{\mathbf{c}}, \hat{\imath}) = \mathrm{argmax}_{\mathbf{c},\, i \in \mathrm{BR}(\mathbf{c})}\ u^d(\mathbf{c}, i)$, where $\mathrm{BR}(\mathbf{c}) \coloneqq \underset{i \in T}{\mathrm{argmax}}\, u^a(\mathbf{c}, i)$

When attacker type (payoffs) is uncertain…

👉 **Learn** optimal commitment by observing **attacker best responses** [Letchford et al., 2009; Blum et al., 2014; Haghtalab et al., 2016; Roth et al., 2016; Peng et al., 2019]

	1^{a}	2^{a}
1^{d}	*1, -1*	*-1, 1/3*
2^{d}	*-1, 3*	*0.9, -1*

Type A

	1^{a}	2^{a}
1^{d}	*1, -1*	*-1, 1*
2^{d}	*-1, 1*	*0.9, -1*

Type B

Example: A *defender* (*row player*) wants to defend two areas 1 and 2, which a *poacher* (*column player*) wants to attack. The poacher may be of Types A or B as his payoffs depend on animal prices on the black market, which fluctuate and are held private by the poacher.

To learn attacker type, play (0.6, 0.4):

- If best response 1^{a}, *Type A*; o.w., *Type B*

(More generally: learn optimal commitment in a continuous type space [Blum et al., 2014; Peng et al., 2019])

Key assumption: *truthful* attacker responses. *What if not?* 🤔

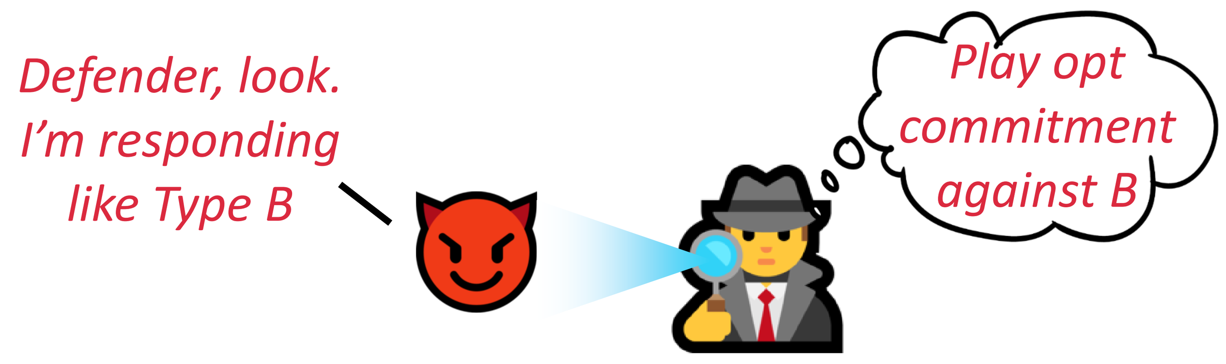
2 Manipulating a Learning Defender

When attacker is **truthful**

	Type A	*Type B*
Optimal commit:	(0.75, 0.25)	(0.5, 0.5)
Induced best response:	1^{a}	1^{a}
Defender utility:	0.5	0
Attacker utility:	0	0

When attacker is **untruthful**…

- ***Type-A* attacker**: manipulate by best responding like Type B
- Defender plays opt commit against *Type B*, obtaining utility 0



Theorem. When attacker can report an arbitrary type, it is always optimal to report the **zero-sum** type. Defender learns the **maximin** strategy as her optimal commitment as a result.

3 Handling Attacker Manipulation

A policy-based playbook

- **Stage 1**: **Defender** commits to **policy** $\pi: \Theta \to \mathcal{C} \times T$, specifying a strategy $\pi(\mathbf{c})$ to play for each reported/learned attacker type $\theta \in \Theta$, and a response $t \in BR_\theta(\mathbf{c})$ to induce the attacker to take.
- **Stage 2**: **Attacker** (of true type θ) choose optimally a type $\beta = \underset{\theta' \in \Theta}{\mathrm{argmax}}\, u^{\mathrm{a}}(\pi(\theta'))$ and behaves like this type, i.e., **report** type β.
- **Stage 3**: Outcome $(\mathbf{c}, t) = \pi(\beta)$ realized: defender plays $\mathbf{c}$ and attacker best responds $t \in \mathrm{BR}_\beta(\mathbf{c})$, obtaining $u^{\mathrm{d}}(\mathbf{c}, t)$ and $u_\theta^{\mathrm{a}}(\mathbf{c}, t)$.

Example:

- Play $\mathbf{c}^A = \left(\frac{3}{4}, \frac{1}{4}\right)$ and induce $1^{\mathrm{a}} \in \mathrm{BR}_A(c^A)$ if att. behaves like A;
- Play $\mathbf{c}^B = \left(\frac{1}{2}, \frac{1}{2}\right)$ and induce $2^{\mathrm{a}} \in \mathrm{BR}_B(c^B)$ if att. behaves like B.

👉 ***Type-A attacker* no longer has incentive to misreport *Type B*!**

Optimal policy to commit to? What quality measure?

Worst-case defender utility? Unable to distinguish quality of many polices, however (see Proposition 5 in paper).

- **Efficiency of a Policy (EoP):** an alternative measure

$$\mathrm{EoP}(\pi) = \min_{\theta \in \Theta} \frac{u^{\mathrm{d}} \text{ when } \theta \text{ reports } \mathbf{optimall} \text{ against } \pi}{u^{\mathrm{d}} \text{ when } \theta \text{ reports } \mathbf{truthfully}}$$

- Higher EoP, less utility loss due to manip. $\mathrm{EoP}(\pi) \in [0,1]$.

4 Computing the Optimal Policy

A polynomial-time algorithm for a *finite* set Θ of attacker types

Algorithm 1: Decide if there exists a policy π such that $\mathrm{EoP}(\pi) \ge \xi$

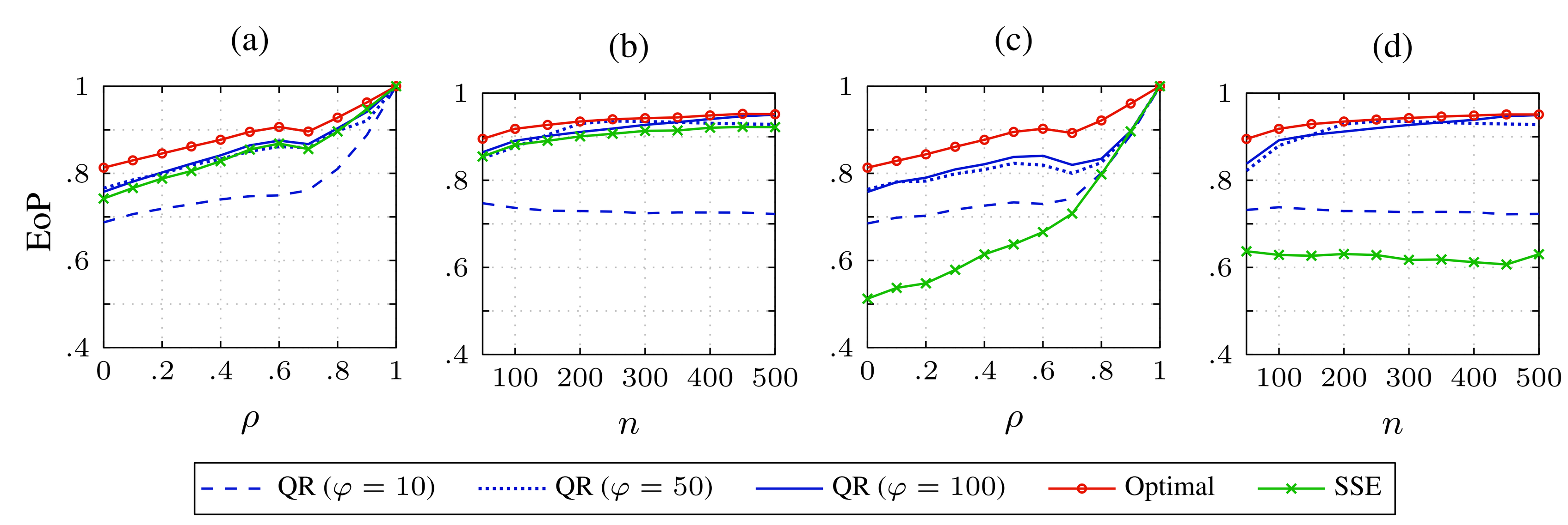
1. For each $\theta \in \Theta$, compute an SSE $(\hat{\mathbf{c}}^\theta, \hat{\imath}^\theta)$ on type θ. Let $\hat{u}(\theta) = u^{\mathrm{d}}(\hat{\mathbf{c}}^\theta, \hat{\imath}^\theta)$.
2. Sort attacker types in Θ by $\hat{u}(\theta)$, so that $\hat{u}(\theta_1) \ge \hat{u}(\theta_2) \ge \dots \ge \hat{u}(\theta_\lambda)$, $\lambda = |\Theta|$
3. For each $\ell = 1, \dots, \lambda$, let $\pi(\theta_\ell) = (\mathbf{z}, t)$, where $z_i = \min\{\hat{c}_i^{\theta_\ell}, h_i\}$, $t = \mathrm{BR}_{\theta_\ell}(\mathbf{h})$, and $h_i = \max\left\{0, \frac{\xi \cdot \hat{u}(\theta_\ell) - p_i^{\mathrm{d}}}{r_i^{\mathrm{d}} - p_i^{\mathrm{d}}}, \max_{\theta \in \{\theta_1, \dots, \theta_{\ell-1}\}} \frac{u_\theta^{\mathrm{a}}(\pi(\theta)) - r_i^\theta}{p_i^\theta - r_i^\theta}\right\}$.
4. If $\mathrm{EoP}(\pi) \ge \xi$, return π as a satisfying policy; o.w., claim no such policy exists.

Theorem. In polynomial time, Algorithm 1 either outputs a policy π with $\mathrm{EoP}(\pi) \ge \xi$, or decides correctly that no such policy exists. The policy generated is *incentive compatible* (IC).

QR policy for an *infinite* or *unknown* Θ

- **QR policy:** when θ is reported, play the SSE strategy $\hat{\mathbf{c}}^\theta$ against θ and induce attacker best response in a QR manner, with probability $\sigma(i) = \dfrac{e^{\varphi \cdot u^d(\hat{\mathbf{c}}^\theta, i)}}{\sum_{j \in \mathrm{BR}_\theta(\hat{\mathbf{c}}^\theta)} e^{\varphi \cdot u^d(\hat{\mathbf{c}}^\theta, j)}}$ for each $i \in \mathrm{BR}_\theta(\hat{\mathbf{c}}^\theta)$.

5 Empirical Evaluation



EoP comparison of different policies. In (a), other parameters are set to $\lambda = 100$, $m = 10$, and $n = 50$; and in (b), $m = n/5$, $\rho = 0.5$, and $\lambda = 100$. Figs. (c) and (d) repeat (a) and (b), respectively, with the difference that the zero-sum attacker type is always included in Θ.

Presented at NeurIPS'19, Vancouver