Adversarial Sequential Decision Making

Part 3: Training Time Defense

Wen Sun

MAX PLANCK INSTITUTE FOR SOFTWARE SYSTEMS

WISCONSIN UNIVERSITY OF WISCONSIN–MADISON

25th July, 2022

Overview

- Introduction: robust supervised learning (linear regression)
- Robustness in offline RL [ZCZS 2021b]
- Robustness in online RL [ZCZS 2021a]
- Experiments

Robust Linear regression

Given a **clean** dataset $\tilde{D} = (x_i, y_i)_{i=1}^N$, where $x \sim \nu, \quad \|x\| \leq 1,$

$y = x^\top w^\star + \delta$, with δ being sub-gaussian and $\mathbb{E}[\delta] = 0, \mathbb{E}[\delta^2] \leq \gamma^2$

Adversary can arbitrarily corrupt ϵN many pairs from $\tilde{D}$

Then, there exists robust linear regression algorithm that returns an estimator $\hat{w}$, s.t.,

$$\mathbb{E}_{x\sim\nu}(x^\top(w^\star - \hat{w}))^2 \leq c\left(\frac{\gamma^2 \text{poly}(d)}{N} + \gamma^2\epsilon\right)$$

Is statistically robust RL possible?

Markov Decision Process (MDP)

An MDP $M = \left(S, A, R, P, \mu_0, \gamma\right)$ is defined by the following elements:

- the state space S.
- the action space A.
- the reward function $R : S \times A \rightarrow \Delta_{\mathbb{R}}$.
- the transition function $P : S \times A \rightarrow \Delta_S$.
- the initial state distribution $\mu_0 \in \Delta_S$.
- the discounting factor $\gamma \in [0,1)$.

Policy and Value

- A (stochastic) ***policy*** $\pi : S \rightarrow \Delta_A$ specify a strategy of choosing the action based on the current state, i.e. $a_t \sim \pi\left(s_t\right)$.
- The ***value function*** w.r.t. a policy π is defined as"

$$V^{\pi}(s) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \middle| \ \pi, \ \ s_1 = s\right]$$

- The ***Q function*** w.r.t. a policy π is defined as:

$$Q^{\pi}(s, a) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \middle| \ \pi, \ \ s_1 = s, \ a_1 = a\right]$$

- The advantage function is defined as: $A^{\pi}(s, a) = Q^{\pi}(s, a) - V^{\pi}(s)$

Optimal Policy Identification (OPI)

- The *objective* of RL is to learn a policy that maximize the expected discounted sum of reward:

$$J(\pi) = \mathbb{E}_{s \sim \mu_0}\left[V^{\pi}(s)\right]$$

- The *optimal policy* is defined as $\pi^* = \operatorname{argmax}_{\pi} J(\pi)$.

- The learning goal is to find a ϵ-optimal policy $\hat{\pi}$, i.e.

$$J(\pi^*) - J(\hat{\pi}) \le \epsilon.$$

Overview

- Introduction
- Robustness in offline RL [ZCZS 2021b]
- Robustness in online RL [ZCZS 2021a]
- Experiments

The ϵ-Contamination model in Offline RL

1. A clean dataset $D \sim \mu(s, a)$ of size T: $\{(s_t, a_t, r_t, s_t')\}_{t=1:T}$
2. An adversary replace an ϵ fraction of D with arbitrary transitions $(s, a, r, s') \in S \times A \times \mathbb{R} \times S$.
3. The learner observes the contaminated dataset and try to find a $poly(\epsilon)$-optimal policy.

Assumptions

Assumption 1 (Tabular MDP and Exploratory Behavior Policy):

- *We assume that both state and action spaces are finite, with size S and A respectively.*
- *We also assume that the dataset D is collected by an exploratory behavior policy, such that each (s, a) is visited with some non-zero probability $p(s, a)$.*

Finding 1: the statistical limit of robustness in offline RL.

Impossibility Result

Theorem 1. For any given $\epsilon \in (0, 2/SA]$ and exploratory data distribution $p(s, a)$, under ϵ-contamination, no offline RL algorithm can find a better than $SA\epsilon/2$-optimal policy with probability more than 1/2 on all MDPs.

<u>Key Idea:</u>

- A sparse reward structure: only (s^*, a^*) has positive reward Bernoulli($SA\epsilon/2$).
- There exists an (s, a) pair has at most $\dfrac{T}{SA}$ data points.
- The attacker can concentrate on (s, a), and flip the reward to 1 on ϵT data points of (s, a).
- Then, (s, a) will look as good as Bernoulli($SA\epsilon$).

Interpretation of the result

- Unlike high-dimensional robust statistics, here our optimality gap has an SA dependence

- Thus robustness of offline rl is not possible for high-dimensional setting, i.e., large-scale MDPs.

Any remedy?

- Q: Can we avoid an explicit SA scaling, i.e., achieve dimension-independent optimality gap?
- A: On-policy policy gradient!

Overview

- Introduction
- Robustness in offline RL [ZCZS 2021b]
- Robustness in online RL [ZCZS 2021a]
- Experiments

The ϵ-Contamination model in Online RL

1. *At any timestep t, the adversary observes (s_t, a_t) and decides whether to supersede the environment to provide any $r_t^\dagger \in \mathbb{R}$ and $s_{t+1}^\dagger \in S$.*
2. *The adversary cannot contaminate in more than ϵK episodes, K being the total number of interaction episodes.*

Remarks:

- Strictly stronger than the adversary model in existing online learning literatures.

The classic Policy Gradient Algorithms

- Policy Gradient [Williams 1992]:
  1. Denote

$$d_\nu^\pi(s) = \left(1-\gamma\right)\sum_{t=1}^{\infty}\gamma^{t-1}Pr^\pi\left(s_t = s \;\middle|\; s_0 \sim \nu\right).$$

  1. Policy gradient:

$$\nabla_\theta J\left(\pi_\theta\right) = \frac{1}{1-\gamma}\mathbb{E}_{s\sim d_{\mu_0}^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(s)}\left[\nabla_\theta \log\pi_\theta(a\,|\,s)A^{\pi_\theta}(s,a)\right].$$

The classic Policy Gradient Algorithms

- Natural Policy Gradient (NPG) [Kakade, 2001]:
  1. Fisher Information Matrix:

$$F(\theta) = \mathbb{E}_{s\sim d_{\mu_0}^{\pi_\theta}}\mathbb{E}_{a\sim\pi_\theta(s)}\left[\nabla_\theta \log\pi_\theta(a\,|\,s)\left(\nabla_\theta \log\pi_\theta(a\,|\,s)\right)^\top\right]$$

  2. Gradient ascent: $\theta^{(t+1)} = \theta^{(t)} + \eta F\left(\theta^{(t)}\right)^{-1}\nabla_\theta J\left(\pi_\theta\right).$

$$w := \arg\min_{w} \mathbb{E}_{s,a\sim d_{\mu_0}^{\pi_\theta}}\left(w^\top \nabla \ln \pi_\theta(a\,|\,s) - A^{\pi_\theta}(s,a)\right)^2$$

Least square from feature $\phi(s,a) := \nabla \ln \pi_\theta(a\,|\,s)$ to $A^{\pi_\theta}(s,a)$

Sample-based Filtered NPG (FPG)

Possibly corrupted by adversary already!

- In each iteration t,

1. run $\pi^{(t)}$ to collect a dataset $\left(s_i, a_i, \hat{A}(s_i, a_i)\right)_{i=1:M}$ where $(s_i, a_i) \sim d_\nu^{\pi^{(t)}}$.
2. Solve the robust linear regression problem:

$$w^{(t)} = \text{Robust LS}\left((s, a, \hat{A})_{i=1:M},\ \ \phi(s, a) := \nabla \ln \pi_{\theta^t}(a \mid s)\right)$$

3. Policy gradient update:

$$\theta^{(t+1)} = \theta^{(t)} + \eta\, w^{(t)}$$

Robustness of FPG

Assumption 1 (Linear Advantage Function): We assume that there exists a feature map $\phi: S \times A \to \mathbb{R}^d$, *such that for any* (s, a, π), *we have*

$$A^\pi(s,a) = \phi(s,a)^\top w^\pi, \text{ for some } w^\pi \in \mathbb{R}^d.$$

We assume in addition that, for all (s, a), $\mathbb{E}[r(s,a)] \in [0,1]$, $\mathbb{V}ar[r(s,a)] \leq \sigma^2$ *and* $\left|\left|\phi(s,a)\right|\right| \leq 1$.

<u>Remarks</u>:

1. Assumption 1 is satisfied in, for example, tabular MDPs and linear MDP.

Robustness of FPG

Assumption 2 (Exploratory Reset Distribution [Agarwal et al. '20a]*): With respect to any state-action distribution ν, define*

$$\Sigma_\nu = \mathbb{E}_{s,a\sim\nu}\left[\phi_{s,a}\phi_{s,a}^\top\right]$$

and define the <u>relative condition number</u> as

$$\sup_{w\in\mathbb{R}^d} \frac{w^\top \Sigma_{d^*} w}{w^\top \Sigma_\nu w} = \kappa, \text{ where } d^*(s,a) = d_{\mu_0}^{\pi^*}(s,a).$$

We assume that κ is finite and small w.r.t. a reset distribution ν available to the algorithm.

<u>Remarks</u>:

1. An assumption that alleviate the challenge of exploration.
2. We will use the reset distribution ν as our initial distribution μ_0

Finding 2: online RL can be robust.

Robustness of FPG

Theorem 2. Under assumptions 1,2, and under ϵ-contamination there exists a set of hyperparameters agnostic to ϵ, such that FPG with $\mathrm{poly}(d, \frac{1}{\epsilon}, \frac{1}{1-\gamma})$ sample complexity returns a policy $\hat{\pi}$ such that

$$\mathbb{E}\left[J(\pi^*) - J(\hat{\pi})\right] \le \tilde{O}\left(\sqrt{\frac{\kappa}{(1-\gamma)^5}}\epsilon^{1/4}\right).$$

Remarks:

- κ can be as small as 1 for a good reset distribution (e.g., ν is an expert demonstration distribution).

Proof of Theorem 2

Lemma 1 (NPG Regret Lemma [Even-Dar et al. '09, Agarwal et al. '20]*). Under assumptions 1,2, assume that* π_0 *is the uniform policy and the iterates* $w^{(t)}$ *satisfies*

$$\mathbb{E}\left[\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi^{(t)}}(s,a)-\phi(s,a)^\top w^{(t)}\right)^2\right]\right]\leq \epsilon_{stat}^{(t)}$$

Then, NPG satisfies

$$\mathbb{E}\left[\frac{1}{T}\sum_{t=1}^{T}\left(J\left(\pi^*\right)-J(\pi^{(t)})\right)\right]\leq \frac{W}{1-\gamma}\sqrt{\frac{2\log|A|}{T}}+\frac{1}{T}\sum_{t=1}^{T}\sqrt{\frac{4\kappa\epsilon_{stat}^{(t)}}{\left(1-\gamma\right)^3}}$$

Proof of Theorem 2

Lemma 2 (Robust linear regression under adaptive ϵ*-contamination). For a given iteration* t*, suppose the adversary corrupt this iteration with contamination level* $\epsilon^{(t)}$*, then with* M *large enough it is guaranteed that with high probability,*

$$\mathbb{E}_{s,a\sim d^{(t)}}\left[\left(Q^{\pi(t)}(s,a)-\phi(s,a)^\top w^{(t)}\right)^2\right]\leq O\left(\frac{\sqrt{\epsilon^{(t)}}}{\left(1-\gamma\right)^2}\right)$$

- Importantly, note that $\frac{1}{\mathrm{T}}\sum_{t=1}^{T}\epsilon^{(t)}=\epsilon.$
- The result follows by plugging Lemma 2 into Lemma 1 and apply Cauchy–Schwarz:

$$\frac{1}{T}\sum_{t=1}^{T}(\epsilon^{(t)})^{1/4}\leq\frac{1}{T}\sum_{t=1}^{T}\epsilon^{1/4}=\epsilon^{1/4}$$

Lower bound

Theorem 3. For any algorithm, there exists an MDP such that the algorithm fails to find an $O\left(\frac{\epsilon}{2(1-\gamma)}\right)$-optimal policy under ε-contamination with probability at least 1/2.

- Key idea: an adaptive ε-contamination adversary can with large probability "mimic" a different MDP M', and no policy is more than $O(\frac{\epsilon}{2(1-\gamma)})$-optimal in both M and M'.

Summary of Theoretical Results

- Under adaptive ϵ-contamination,

1. Offline RL suffer a worst-case $O(SA\epsilon)$ optimality gap.
2. FPG can find an $O\left(\epsilon^{1/4}\right)$-optimal policy.
3. No algorithm can find better than $O(\epsilon)$-optimal policy.

RL in time-varying MDP

- Several lines of related work:

1. Adversarial MDPs: stochastic transition P, adversarial reward R_k.

   $O(\sqrt{T})$ regret can be achieved.

   [Even-Dar et al. ’09, Neu et al. ’10, ’12, ’13, ’20, Rosenberg and Mansour ’19, Jin et al. ’20, Lee et al. ’20, …]

   - Impossibility result: sublinear regret impossible when both transition and reward are adversarial at the same time. [Yadkori et al., 2013]

RL in time-varying MDP

- Several lines of related work:

2. Online/non-stationary MDPs: the MDP slowly changes over time with total variation Δ. $O(poly(S, A)\Delta^c T^{1-c})$ regret can be achieved.

   [Cheung et al. ’19, Ornik and Topcu ’19, Ortner et al. ’19, Domingues et al. ’20, …]

   - Regret bound blows up when the $\Delta = \epsilon T$.

RL in time-varying MDP

- Several lines of related work:

3. Corruption-robust RL [Lykouris et al., 2019]: at most C episodes are adversarial.

    - finds $O(poly(S, A)C/\sqrt{T})$-optimal policy in tabular MDPs and $O(poly(d)C^2/\sqrt{T})$ in linear MDPs.

    - the bound blows up when $C = \epsilon T$.

RL in time-varying MDP

- Highlights of our work compared to existing works:

  1. We handles both adversarial reward and adversarial transitions.

  2. We are the first to provide meaningful guarantees when the amount of change is linear in T.

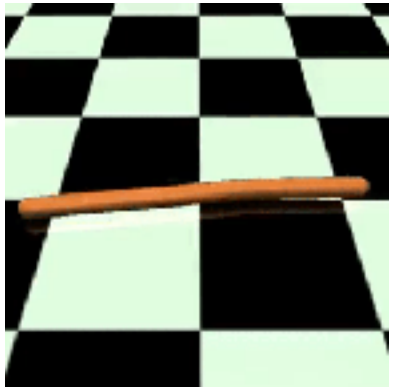
  3. Our algorithm FPG also performs well in practice (to be seen).

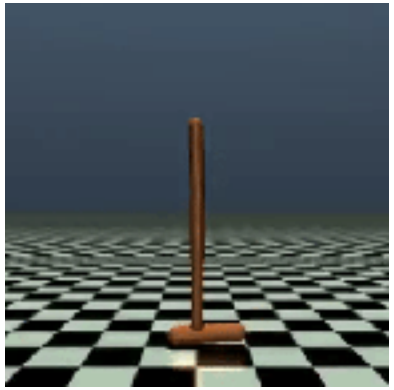
Overview

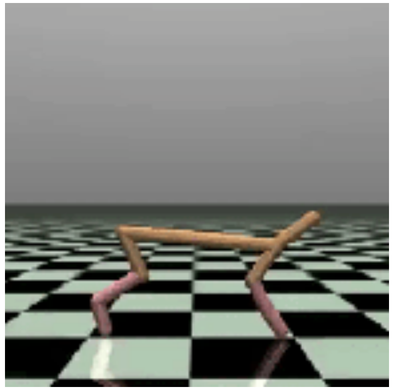
- Introduction
- Robustness in offline RL [ZCZS 2021b]
- Robustness in online RL [ZCZS 2021a]
- Experiments

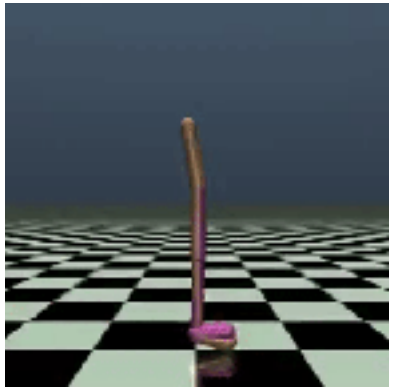
Finding 3: FPG is also robust in practice.

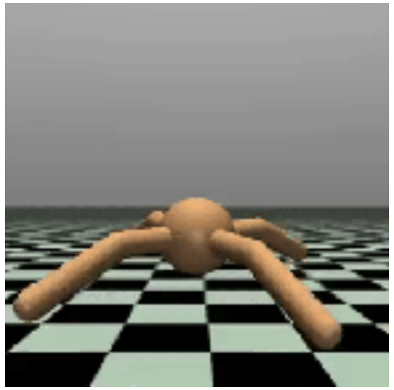
MuJoCo Continuous Control Benchmarks


Swimmer


Hopper


Half-Cheetah

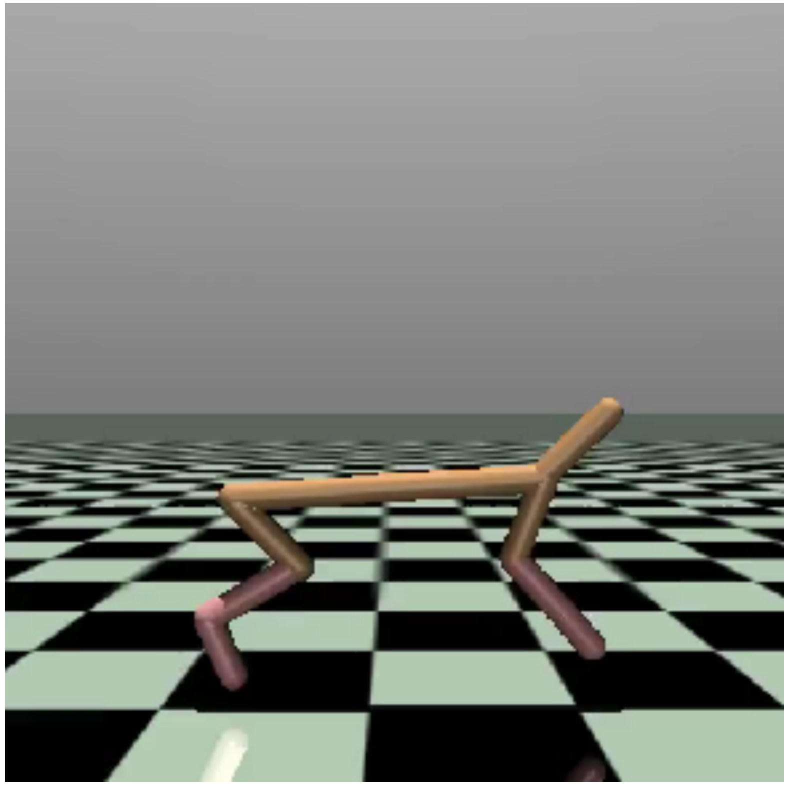

Walker


Ant


Humanoid

Attack Strategy

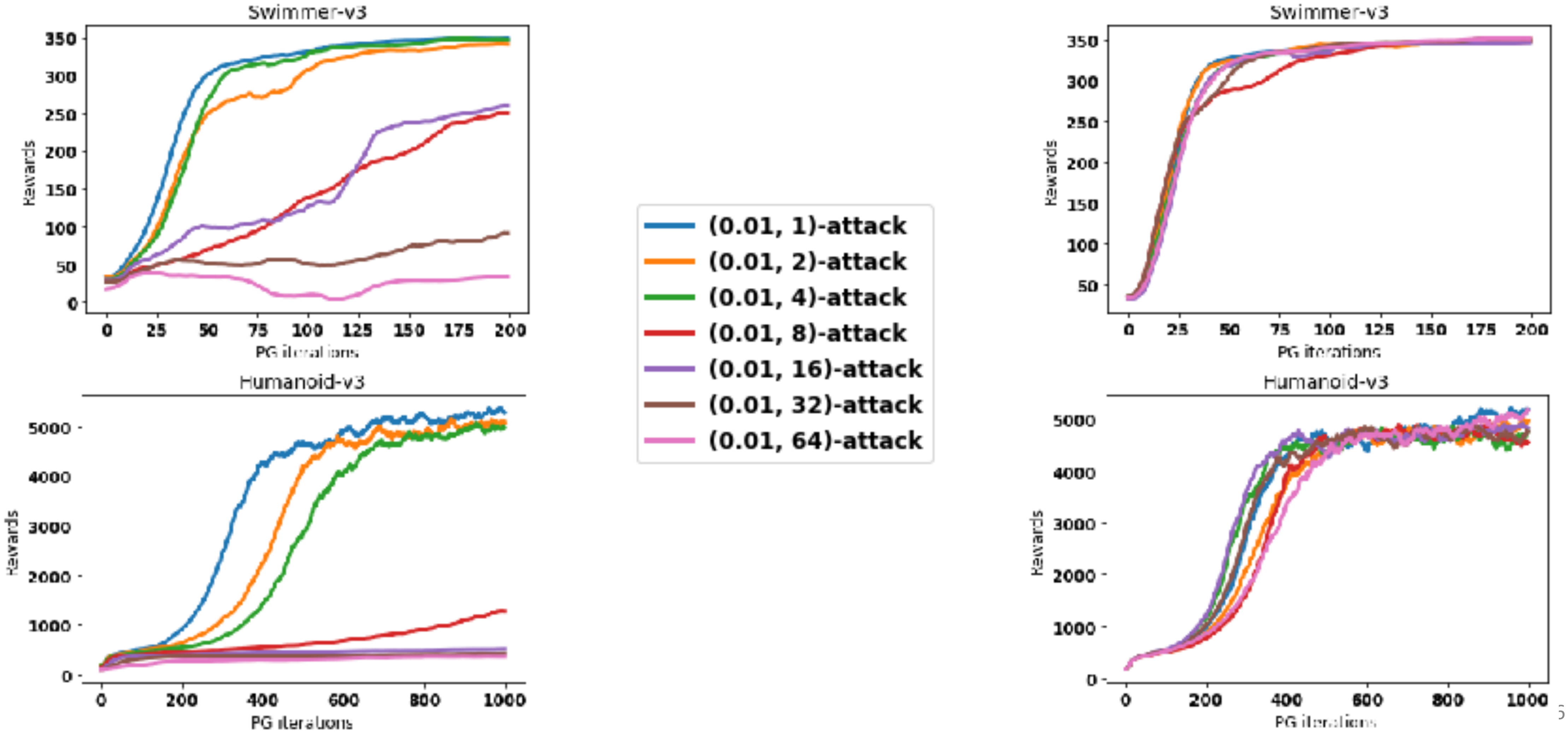
- Policy Gradient Methods: $\theta^{(t+1)} = \theta^{(t)} + g^{(t)}$
- Goal: Perturb $\hat{g}^{(t)}$ to point in the $-g^{(t)}$ direction.
- Simple strategy: flip the rewards and multiply by a big constant!
- (ϵ, δ)-attack: Among the M episodes in each PG iteration, perturb the reward to be $r_t' = -\delta r_t$ in ϵM episodes.



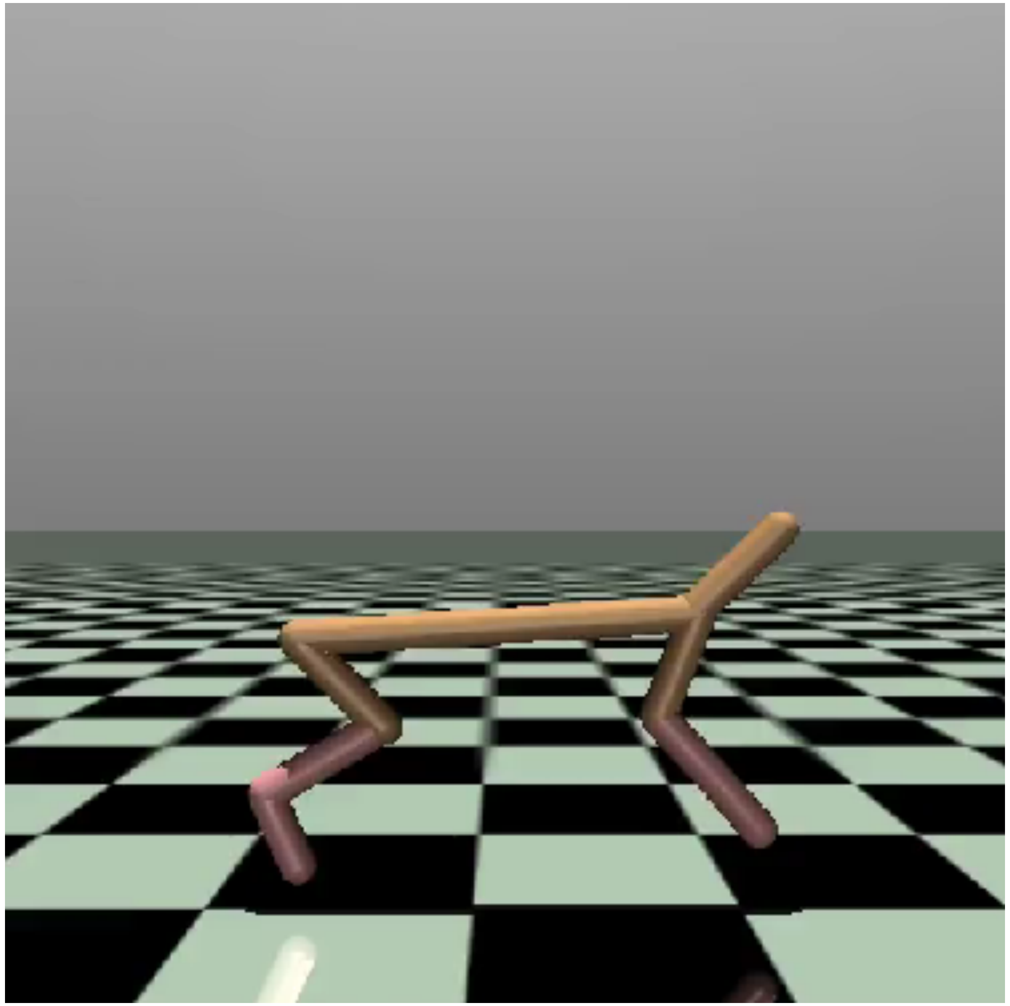
$(0.01, 100)$-attack

TRPO

FPG


Swimmer-v3
Humanoid-v3
Rewards
PG iterations
(0.01, 1)-attack
(0.01, 2)-attack
(0.01, 4)-attack
(0.01, 8)-attack
(0.01, 16)-attack
(0.01, 32)-attack
(0.01, 64)-attack

Happy Cheetah!



References

Corruption Robust Offline Reinforcement Learning, Xuezhou Zhang, Yiding Chen, Jerry Zhu, Wen Sun, AISTATS 2022

Robust Policy Gradient against Strong Data Corruption, Xuezhou Zhang, Yiding Chen, Jerry Zhu, Wen Sun, ICML 2021