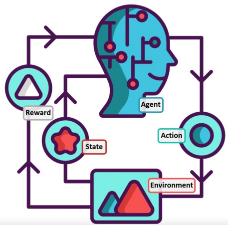
NEURAL INFORMATION PROCESSING SYSTEMS

Robust RL Adversarial Efficiency



Efficient Adversarial Training without Attacking: Worst-Case-Aware Robust Reinforcement Learning

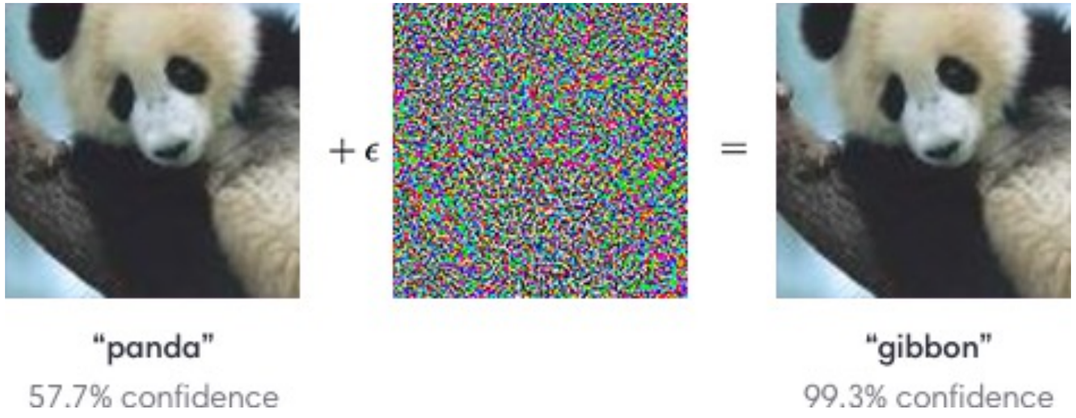
Yongyuan Liang*(cheryllLiang@outlook.com), Yanchao Sun*(ycs@umd.edu)
Ruijie Zheng, Furong Huang

SEE OUR WORK

Background: RL agents are vulnerable. Why?

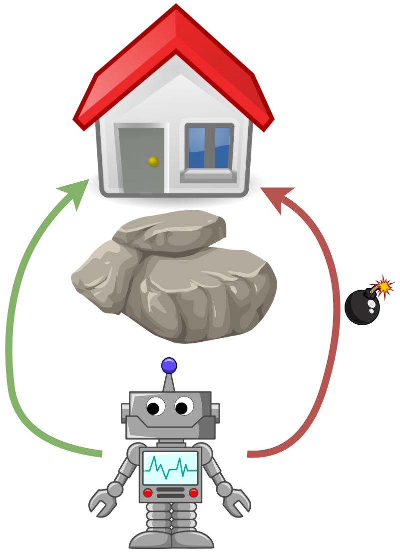
Vulnerability from DNN approximator

Deep reinforcement learning learns complex policies in large-scale tasks using DNNs. Well-trained DNNs easily fail under adversarial attacks of the input.



Intrinsic vulnerability

Intrinsic vulnerability of policies comes from the dynamics of the environment. Red policy can be dangerous under adversarial perturbations!!!



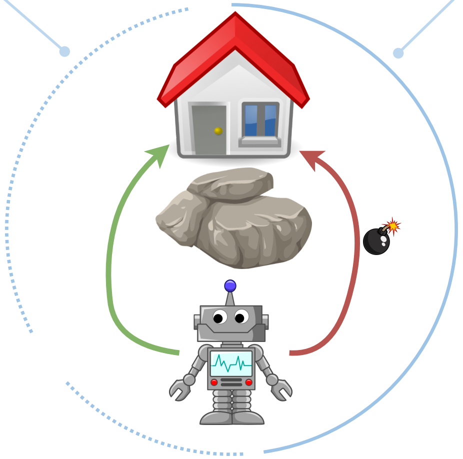
Challenge: Efficiently Enhancing Intrinsic Robustness

Problems: Long-term vulnerability

How to learn RL policies with stronger intrinsic robustness.

Difficulty: Efficiency

Efficiently robust training without requiring much more effort than vanilla training.



Ignoring the worst case may fail

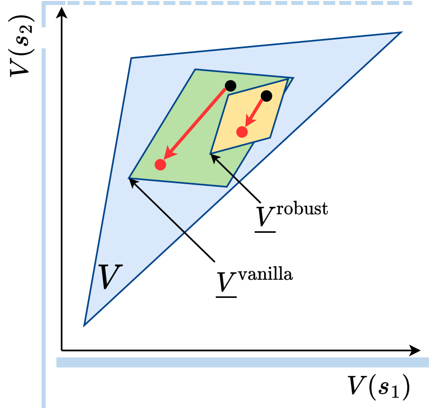
Regularization-based methods[1] neglecting the intrinsic vulnerability, fail under strong attacks.

Very expensive robust training

SOTA Alternating Training with Learned Adversaries (ATLA)[2] doubles the computational cost.

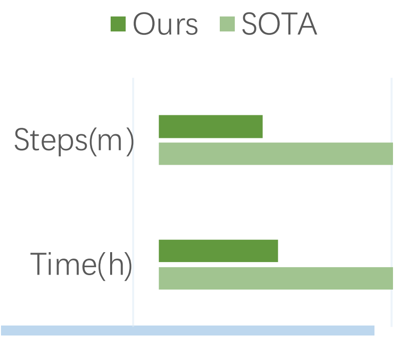
Prior Solutions

Contributions



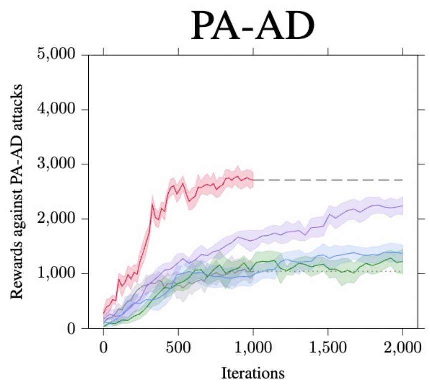
Training Framework: WocaR-RL

Worst-case-aware Robust RL: directly optimizes the worst-case values



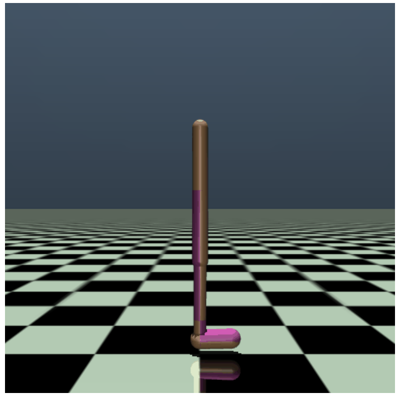
Efficiency

saves about 50% training samples and 50% time



Improve Robustness

obtain 20% more rewards under the strongest attacker



Interpretable Behaviors

learns to lower down its body, which is more intuitive and interpretable

Our Methods

Mechanism 1: Worst-attack Value Estimation

01

💡 Worst-attack Bellman Operator as a contraction:

$$(\underline{\mathcal{T}}^{\pi}Q)(s,a) \coloneqq \mathbb{E}_{s' \sim P(s,a)}[R(s,a) + \gamma min_{a' \epsilon \mathcal{A}_{adv}(s',\pi)} Q(s',a')]$$

02

💡 Estimating worst-attack value by minimizing the estimation loss:

$$\mathcal{L}_{est}\left(\underline{Q}_{\phi}^{\pi}\right) \coloneqq \frac{1}{N}\sum_{t=1}^{N}(\underline{y}_t - \underline{Q}_{\phi}^{\pi}(s_t, a_t))^2,$$

$$where\ \underline{y}_t = r_t + \gamma min_{a' \epsilon \mathcal{A}_{adv}(s_{t+1}, a')} \underline{Q}_{\phi}^{\pi}(s_{t+1}, a')$$

$\mathcal{A}_{adv}$ denotes the set of actions an adversary can mislead the victim π into selecting by perturbing the state s_{t+1} into a neighboring state $\tilde{s}_{t+1}$.

Our Methods

Mechanism 2: Worst-case-aware Policy Optimization

01

💡 Minimizing the worst-attack policy loss below:

$$\mathcal{L}_{wst}\left(\pi_\theta; \underline{Q}_\phi^\pi\right) := -\frac{1}{N}\sum_{t=1}^{N}\sum_{a\epsilon\mathcal{A}}\pi_\theta(a|s_t)\,\underline{Q}_\phi^\pi(s_t,a)\,,$$

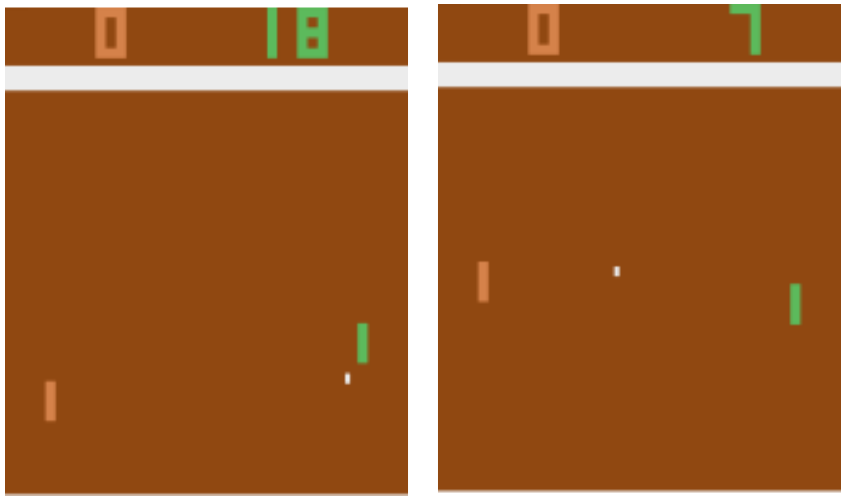
$$\textit{where } \underline{Q}_\phi^\pi \textit{ is the worst attack critic learn via } \mathcal{L}_{est}$$

02

💡 We illustrate how to implement $\mathcal{L}_{wst}$ for PPO and DQN

Our Methods

Mechanism 3: Value-enhanced State Regularization



(left) high weight $w(s)$ and (right) low weight $w(s)$

01

💡 Characterize state importance $s \in \mathcal{S}$

$$w(s) = max_{a_1 \in \mathcal{A}} Q^{\pi}(s, a_1) - min_{a_2 \in \mathcal{A}} Q^{\pi}(s, a_2)$$

02

💡 By incorporating the state importance weight, regularize the policy network loss:

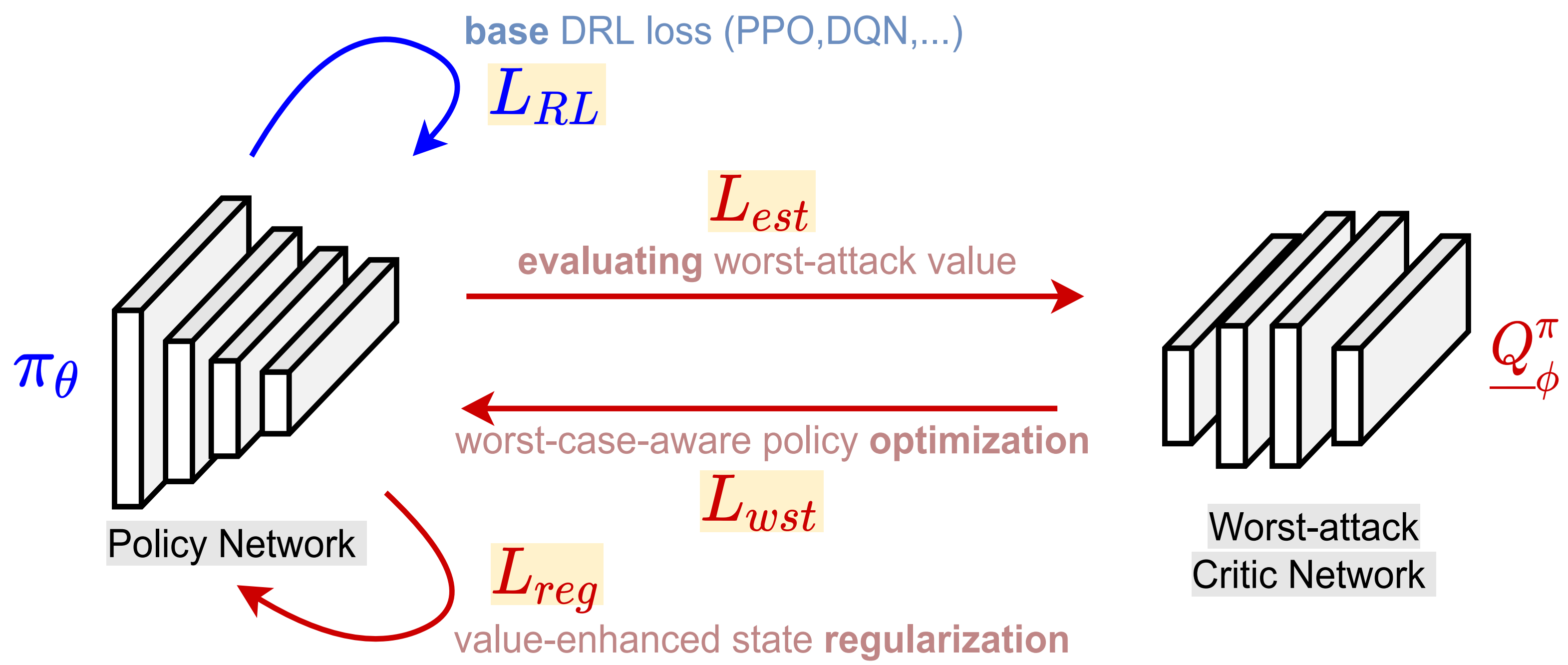
$$\mathcal{L}_{reg}(\pi_\theta) := \frac{1}{N} \sum_{t=1}^{N} w(s_t) max_{\tilde{s}_t \in B_\epsilon(s_t)} Dist(\pi_\theta(s_t), \pi_\theta(\tilde{s}_t)),$$

WocaR: Generic Training Framework

💡 Training architecture:

We train an extra worst-attack critic network $\underline{Q}_\phi^\pi$:

$$\mathcal{L}_{\underline{Q}_\phi^\pi} := \mathcal{L}_{est}\,(\underline{Q}_\phi^\pi)$$

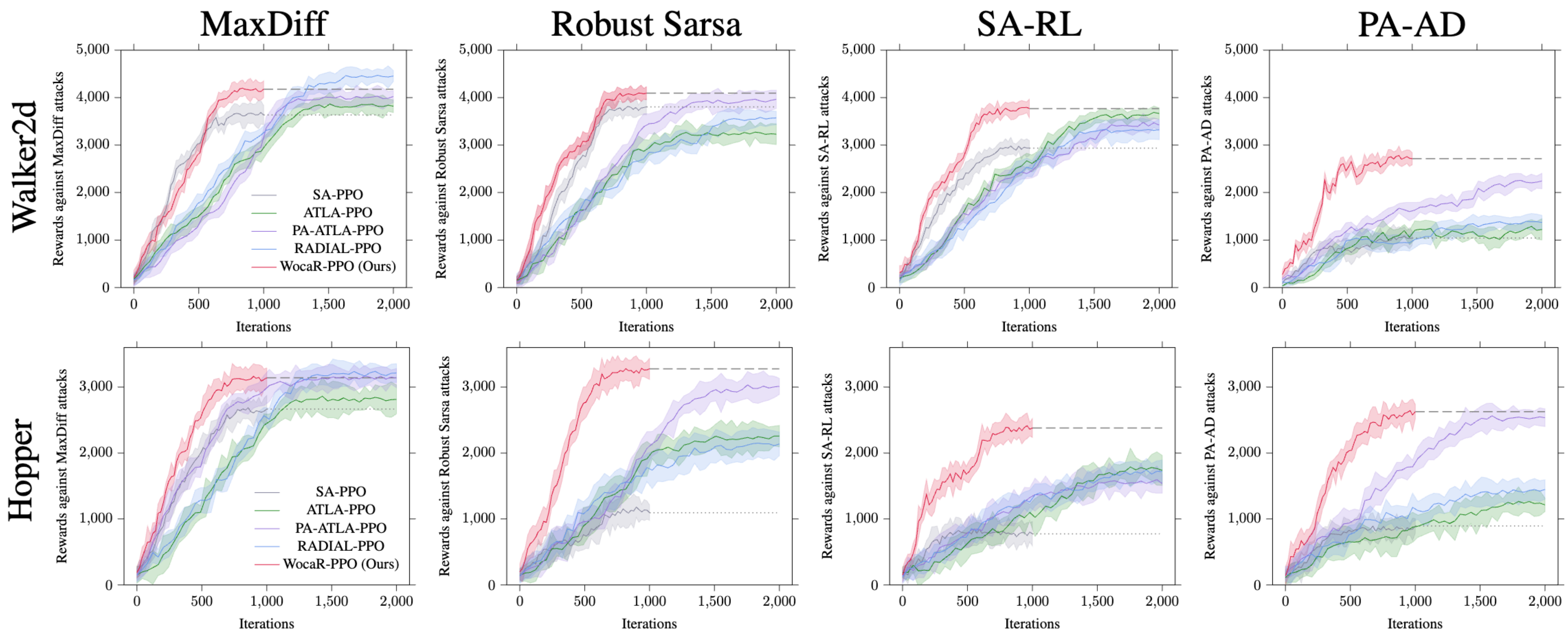


💡 Optimize the policy network π_θ by minimizing the combined loss:

$$\mathcal{L}_{\pi_\theta} := \mathcal{L}_{RL} + \kappa_{wst}\,\mathcal{L}_{wst} + \kappa_{reg}\mathcal{L}_{reg}$$

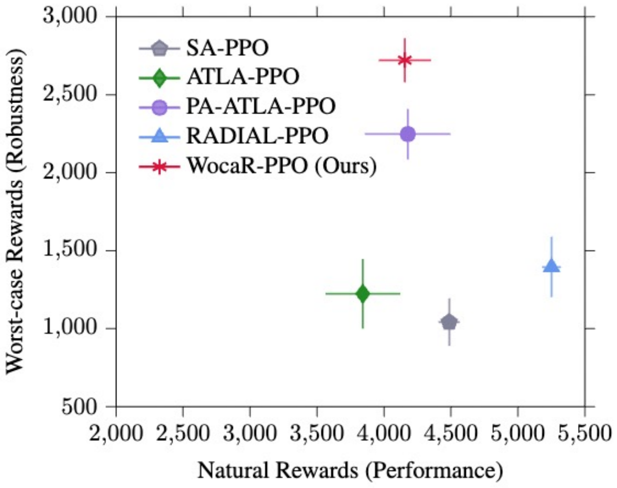
Experiments

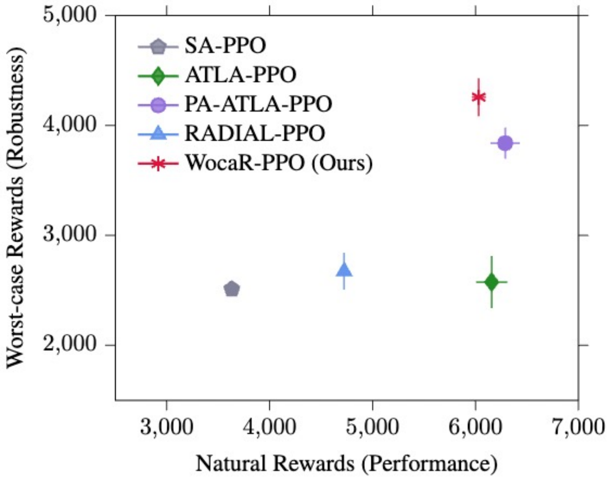
State-of-the-art Robustness of WocaR-PPO

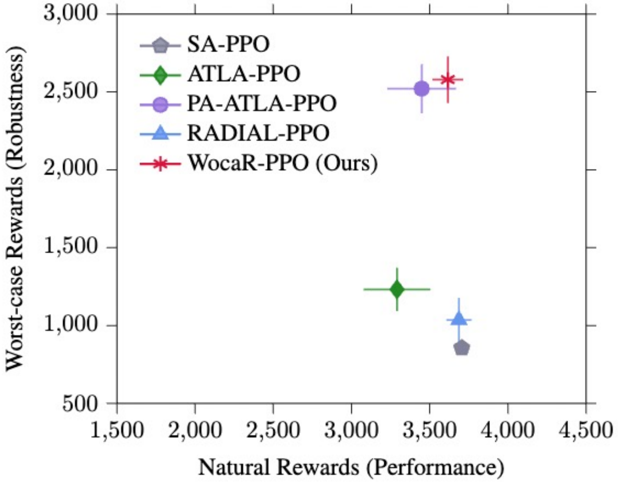


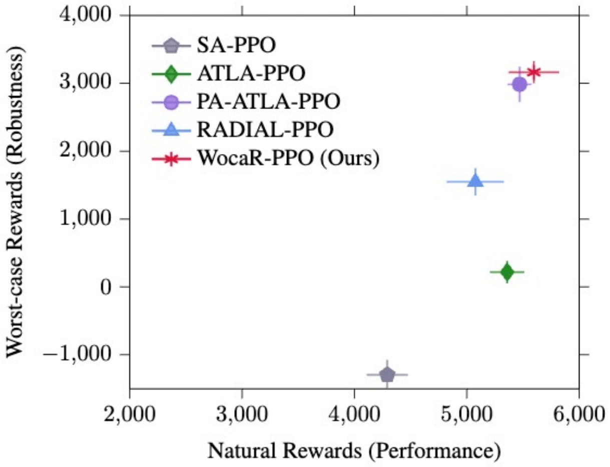
Experiments

Natural performance v.s. Robustness









WocaR-RL maintains competitive natural rewards under no attack,

which successfully gains more robustness without losing too much natural performance.

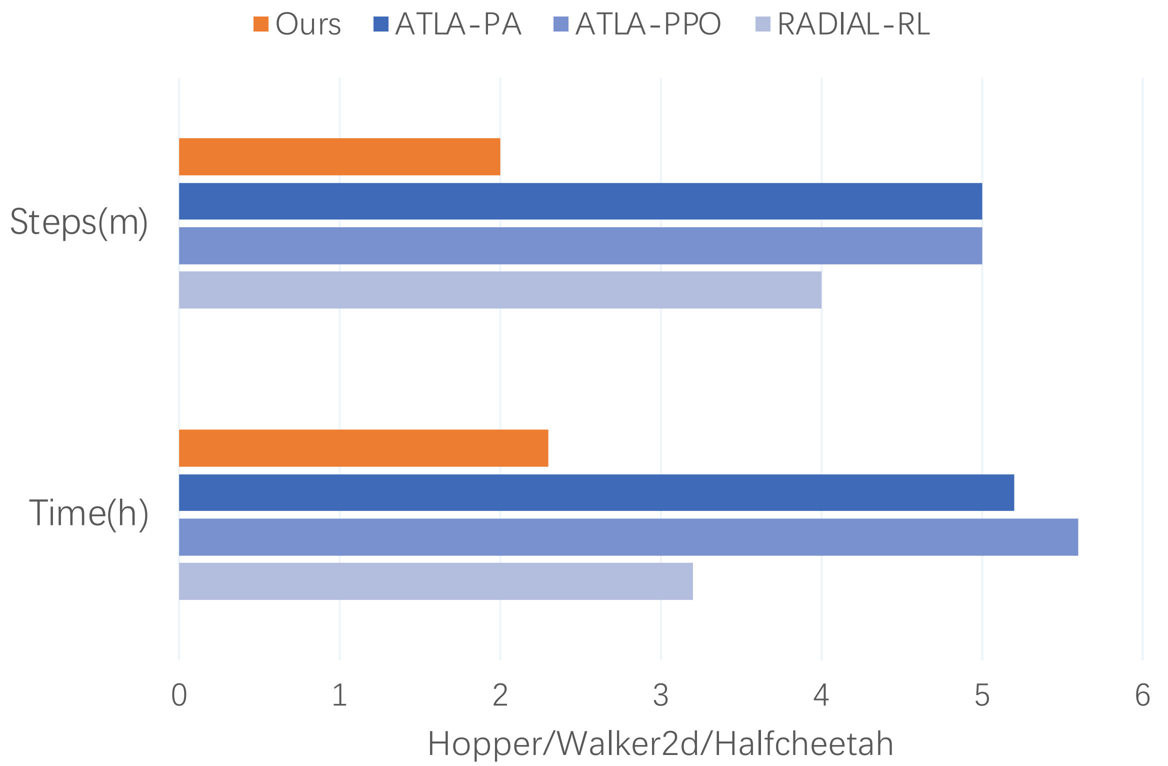
Experiments

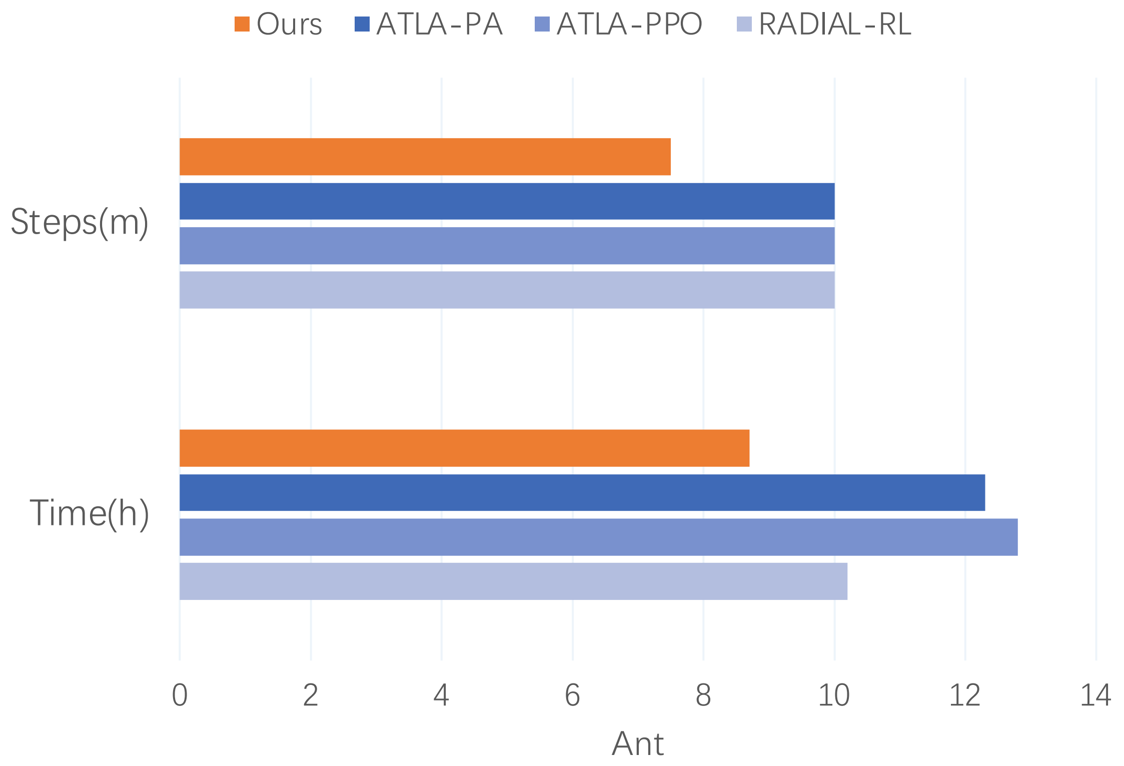
State-of-the-art Robustness of WocaR-DQN

Model	BankHeist ($\epsilon = 3/255$)				RoadRunner ($\epsilon = 3/255$)			
	Clean	PGD	MinBest	PA-AD	Clean	PGD	MinBest	PA-AD
DQN	**1308**	0	119	102	45527	0	2985	203
SA-DQN	1245	1176	1024	489	44638	20678	4214	5516
RADIAL-DQN	1178	1176	928	508	44675	38576	8476	1290
Ours	1220	**1214**	**1045**	**754**	44156	**38720**	**10545**	**8239**

Experiments

Significant training efficiency of WocaR-PPO





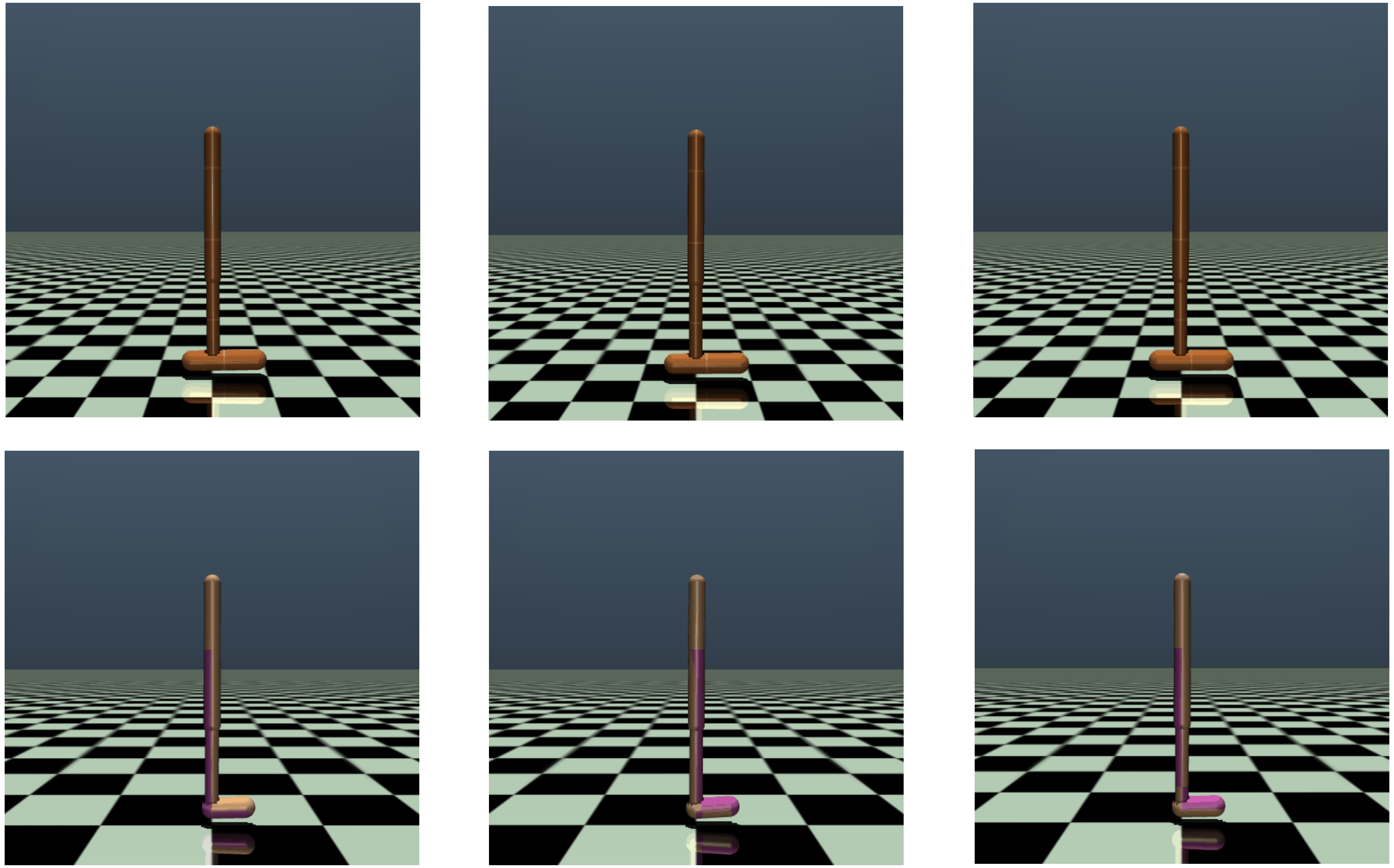
Sampling: requires only 50% or 75% steps for reliably convergence

Time: achieves 1.5 or 2× faster training

Experiments

WocaR-RL learns more interpretable behaviors than SOTA robust methods

ATLA ATLA-PA *Smart* WocaR-RL



THANK YOU FOR WATCH

GOODBYE