Robust Control via Adversarial Training

Hao-Lun Hsu Miroslav Pajic

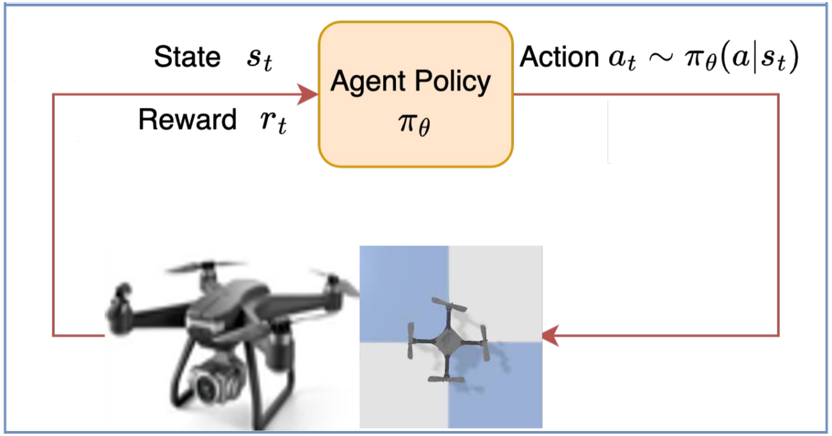
CPSL@Duke
Department of Electrical and Computer Engineering
Department of Computer Science
Department of Mechanical Engineering and Material Science
Duke University

Duke
PRATT SCHOOL of ENGINEERING

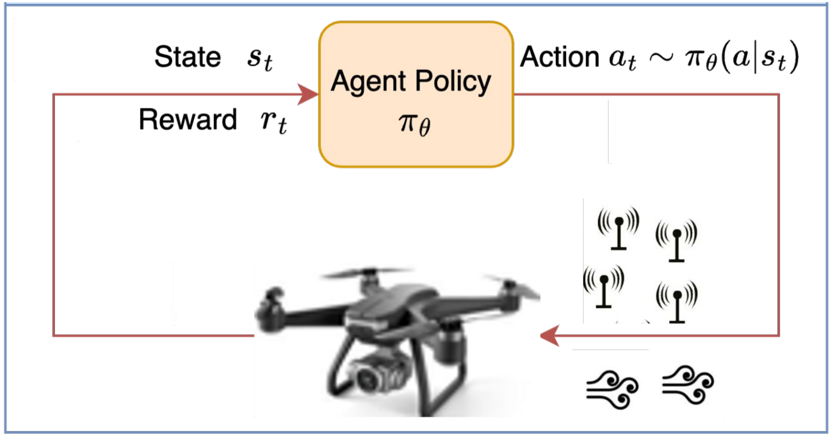
DUKE ROBOTICS

Introduction

Training



Testing

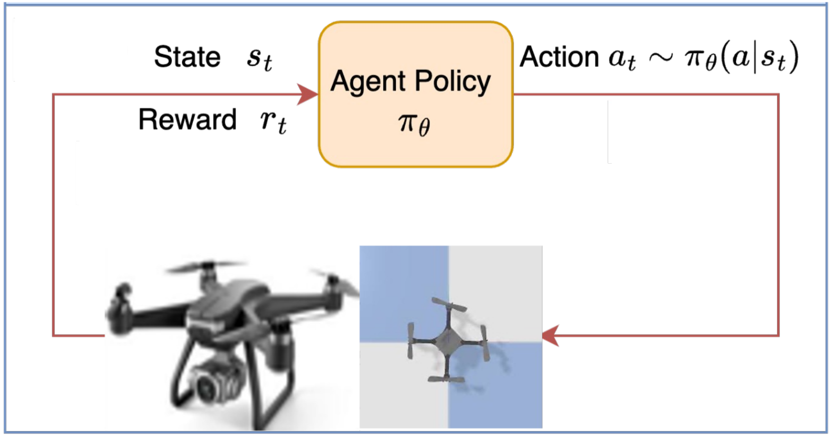


1. Train in **real world**: expensive, dangerous, and time-intensive → a limit set of training scenarios
2. Train in **simulation**: Sim-to-Real gap (reality of simulation) →not robust to modeling errors

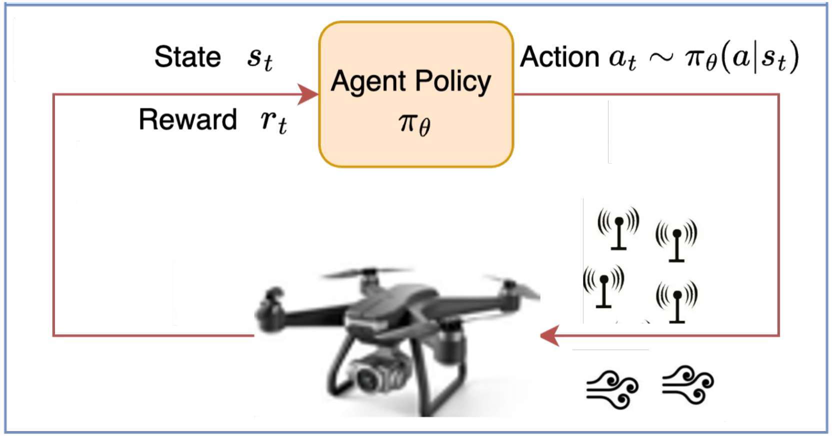
Introduction

Duke PRATT SCHOOL of ENGINEERING

Training

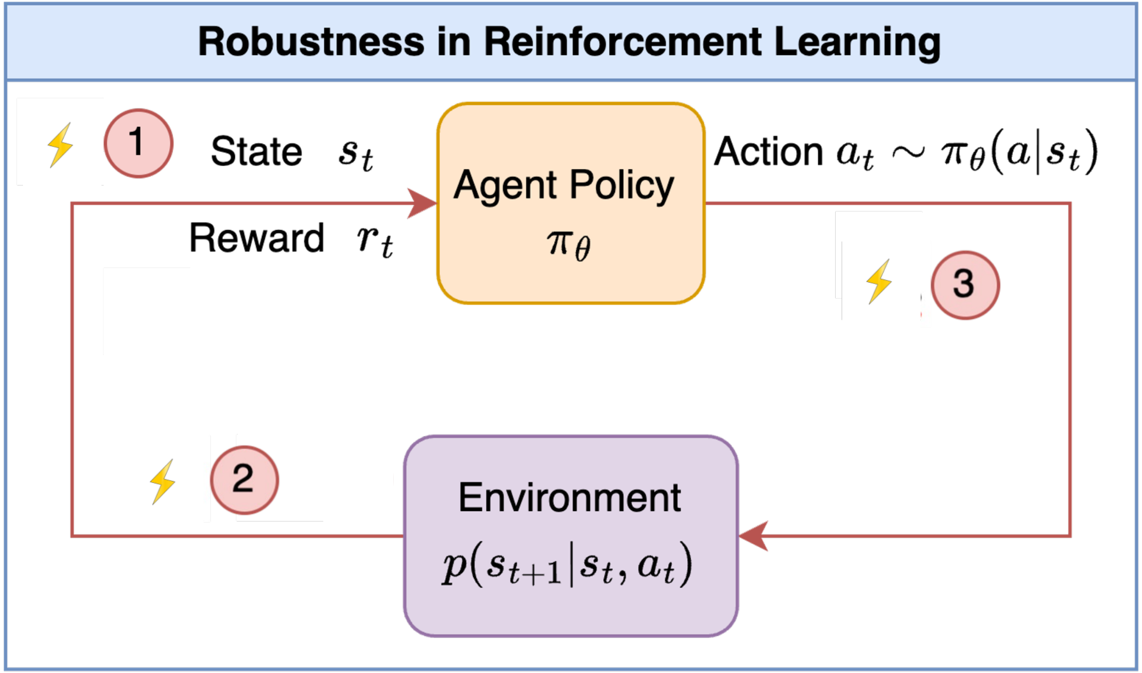


Testing



Robust RL takes the uncertainty of *internal parameters* and *external disturbances* into account

Motivation: Robust Control



Sources of uncertainty/errors:

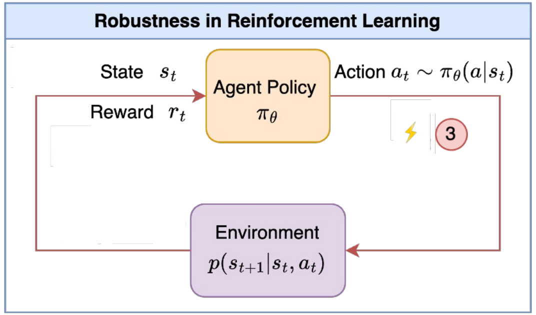
1. **Sensing:** observed states may be different from the true states
2. **Modeling errors:** Transitions dynamics may change
3. **Actuation:** Applied actions may be different from the agent's intention

Robust Control Design with 2-Player Game Design

Duke PRATT SCHOOL of ENGINEERING

$$R(\theta, \phi) \doteq \mathbb{E}_{s_0 \sim p_0} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t^p, a_t^a) \right]$$

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi)$$



Pros

1. Optimize the worst-case performance of RL agents under disturbance
2. Empirical success

Cons

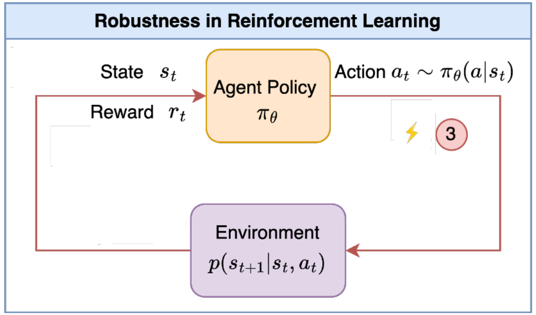
1. Inner minimization problem is difficult to solve → local-optimum
2. worst-case optimization can result in over-conservation if adversary is overly capable

Robust Control Design
with 2-Player Game Design

Duke
PRATT SCHOOL of ENGINEERING

$$R(\theta, \phi) \doteq \mathbb{E}_{s_0 \sim p_0} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, a_t^p, a_t^a) \right] \quad (1)$$

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi) \quad (2)$$



[NeurIPS24*] Adversarial herding for better approximation of the optimal adversary

[ICRA24] Adaptive adversary for unknown adversary strength

$$R(\theta, \phi_i) \doteq \mathbb{E}_{s_0 \sim p_0} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, (1-\alpha)a_t^p + \alpha a_t^a) | C \right]$$

[L4DC24] Efficient exploration via Langevin Monte Carlo with robustness

1. J Dong* and HL Hsu* et al., "Robust Reinforcement Learning through Efficient Adversarial Herding", under review, 2024.
2. HL Hsu et al., "REFORMA: Robust REinFORceMent Learning via Adaptive Adversary for Drones Flying under Disturbances" in *IEEE International Conference on Robotics and Automation* (**ICRA**), 2024.
3. HL Hsu et al., "Robust Exploration with Adversary via Langevin Monte Carlo" in *Learning for Dynamics and Control Conference* (**L4DC**), 2024

Robust Control Design
with 2-Player Game Design

Duke PRATT SCHOOL of ENGINEERING

1. Adversarial ensemble which involves a group of adversaries [1]
   a. Special case in noisy action robust MDP: Adaptive adversary for unknown adversary strengths [2]

2. Efficient exploration via Langevin Monte Carlo with robustness [3]

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi)$$

1. J Dong* and HL Hsu* et al., "Robust Reinforcement Learning through Efficient Adversarial Herding", under review, 2024.
2. HL Hsu et al., "REFORMA: Robust REinFORceMent Learning via Adaptive Adversary for Drones Flying under Disturbances" in *IEEE International Conference on Robotics and Automation* (**ICRA**), 2024.
3. HL Hsu et al., "Robust Exploration with Adversary via Langevin Monte Carlo" in *Learning for Dynamics and Control Conference* (**L4DC**), 2024

Robust Control Design
with 2-Player Game Design

1. Adversarial ensemble which involves a group of adversaries [1]
   a. Special case in noisy action robust MDP: Adaptive adversary for unknown adversary strengths [2]

2. Efficient exploration via Langevin Monte Carlo with robustness [3]

1. J Dong* and HL Hsu* et al., "Robust Reinforcement Learning through Efficient Adversarial Herding", under review, 2024.
2. HL Hsu et al., "REFORMA: Robust REinFORceMent Learning via Adaptive Adversary for Drones Flying under Disturbances" in *IEEE International Conference on Robotics and Automation* (**ICRA**), 2024.
3. HL Hsu et al., "Robust Exploration with Adversary via Langevin Monte Carlo" in *Learning for Dynamics and Control Conference* (**L4DC**), 2024

Robustness with Adversarial Ensembles

Duke PRATT SCHOOL of ENGINEERING

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi)$$

Update a single adversary with first-order optimization method to solve inner optimization

Robustness with Adversarial Ensembles

$$\max_{\theta\in\Theta} \min_{\phi\in\Phi} R(\theta, \phi) \rightarrow \max_{\theta\in\Theta} \min_{\phi\in\widehat{\Phi}} R(\theta, \phi)$$

Update a single adversary with first-order optimization method to solve inner optimization

Employ a set of fixed adversaries $\widehat{\Phi} \doteq \{\phi_i\}_{i=1}^{m}$ where m is the total number of adversaries and for all $i \in [m],\ \phi_i \in \Phi$

Robustness with Adversarial Ensembles

Duke PRATT SCHOOL of ENGINEERING

$$\max_{\theta\in\Theta} \min_{\phi\in\Phi} R(\theta,\phi) \quad \Rightarrow \quad \max_{\theta\in\Theta} \min_{\phi\in\widehat{\Phi}} R(\theta,\phi)$$

Update a single adversary with first-order optimization method to solve inner optimization

Employ a set of fixed adversaries $\widehat{\Phi} \doteq \{\phi_i\}_{i=1}^m$ where m is the total number of adversaries and for all $i \in [m],\ \phi_i \in \Phi$

The gradient of $R(\theta,\phi)$ with respect to the adversary's parameter is d-dimensional

1-dimensional $R(\theta,\phi)$ needs to be approximated

Robustness with Adversarial Ensembles

Duke PRATT SCHOOL of ENGINEERING

$$\max_{\theta\in\Theta} \min_{\phi\in\Phi} R(\theta, \phi)$$

$$\max_{\theta\in\Theta} \min_{\phi\in\widehat{\Phi}} R(\theta, \phi)$$

Efficiently approximate?

Update a single adversary with first-order optimization method to solve inner optimization

Employ a set of fixed adversaries $\widehat{\Phi} \doteq \{\phi_i\}_{i=1}^m$ where m is the total number of adversaries and for all $i \in [m],\ \phi_i \in \Phi$

The gradient of $R(\theta, \phi)$ with respect to the adversary's parameter is d-dimensional

1-dimensional $R(\theta, \phi)$ needs to be approximated

Definitions and Take-aways

Duke
PRATT SCHOOL *of* ENGINEERING

Definition 1: For a function $h : \mathcal{X} \to \mathbb{R}$, we define its L^∞ norm as $||h||_\infty = \sup_{x \in \mathcal{X}} |h(x)|$

Definition 2: Let $(\mathcal{U}, d)$ be a metric space where $d : \mathcal{U} \times \mathcal{U} \to \mathbb{R}^+$ is the metric function. Then a finite set $\mathcal{X} \subset \mathcal{U}$ is an ϵ- packing if no two distinct elements in $\mathcal{X}$ are ϵ-close to each other, i.e.,

$$\inf_{x, x' \in \mathcal{X} : x \neq x'} d(x, x') > \epsilon.$$

Insights from the theoretical results

- When the adversaries in the ensemble are distinct to each other, the accuracy for approximating the true worst-case performance can be improved with increased number of adversaries
- Robust optimization with an adversary ensemble solves the initial optimization problem!

Upper Bound of Almost Sure Approximation

Duke
PRATT SCHOOL *of* ENGINEERING

Let R_Φ denote a function class as $R_\Phi \doteq \{R_\phi \doteq R(\theta, \phi) : \Theta \to \mathbb{R} | \phi \in \Phi\}$.
→ The number of adversaries needed to approximate the inner optimization problem is in approximately **linear** order of the desired precision if the set of adversaries are different enough.

Assumption 1: Assume that R_Φ has finite radius under this metric, i.e., $\sup_{\phi,\phi' \in \Phi} d(R_\phi, R_{\phi'}) \leq r_{\max}$ where $r_{\max} < \infty$ is a finite number.

Interpretation of Assumption 1

- The performance of any protagonist policy in two different environments cannot vary infinitely
- The number of adversaries needs for approximation is about $O(\frac{1}{\epsilon})$

Theorem 1: Consider the metric space $(R_\Phi, ||\cdot||_\infty)$ where for any two functions $R_\phi, R_{\phi'} \in R_\Phi$, the distance between them is defined as $d(R_\phi, R_{\phi'}) \doteq ||R_\phi - R_{\phi'}||_\infty$. With assumption 1, let $\widehat{\Phi} = \{\phi_i\}_{i=1}^m \subset \Phi$, if $R_{\widehat{\Phi}}$ is a maximal ϵ- packing then $|R_{\widehat{\Phi}}| \geq \lceil \frac{r_{\max}}{\epsilon} \rceil$ so that

$$|R(\theta, \phi^*) - R(\theta, \widehat{\phi})| \leq \epsilon$$

Upper Bound of Approximation with High Probability

Duke PRATT SCHOOL of ENGINEERING

Theorem 2: Assume that Φ is a metric space with a distance function $d : \Phi \times \Phi \mapsto \mathbb{R}$. Let σ be any probability measure on Φ. Let $\widehat{\Phi} = \{\phi_i\}_{i=1}^m$ be a set of independently sampled elements from following identical measure σ . consider a fixed $\theta \in \Theta$ and assume that $R(\theta, \phi)$ is an L_ϕ -Lipschitz continuous function of with respect to the metric space (Φ, d). Let $\widehat{\phi}$ and ϕ^* be defined the same as in Theorem 1. For presentation simplicity, assume that $\sigma(\{\phi : d(\phi, \phi^*) \leq \epsilon\}) \geq L_\sigma \epsilon$. Let $0 < \delta < 1$ denote the probability of a bad event. Then with probability $1 - \delta$, the approximation error of $\widehat{\phi}$ on the inner optimization problem is bounded by ϵ if $m \geq \log(\delta) \log^{-1}(1 - \frac{L_\sigma}{L_\phi}\epsilon)$

Now let $\phi_i \in \widehat{\Phi}$ be learners (Φ is an adversary ensemble), instead of fixed adversaries.

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi) \qquad (2)$$

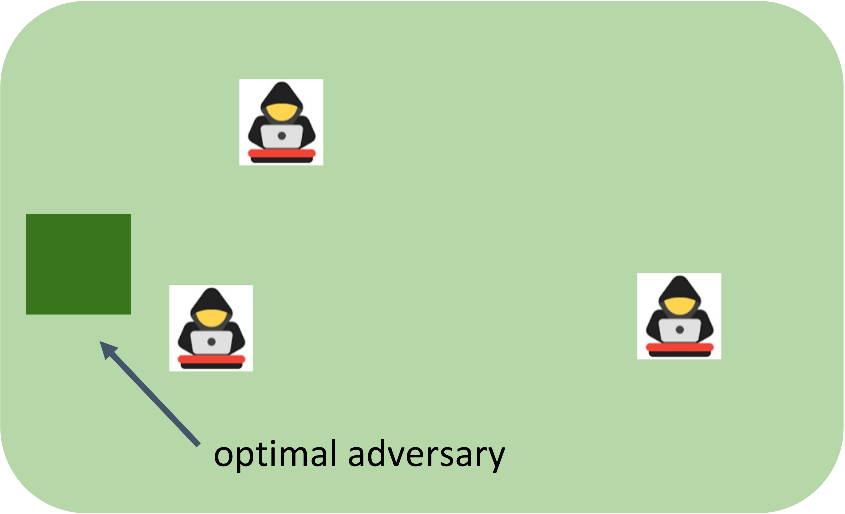
$$\max_{\theta \in \Theta} \min_{\phi_1, \ldots, \phi_m \in \Phi} \min_{\phi \in \{\phi_i\}_{i=1}^m} R(\theta, \phi) \qquad (3)$$

Lemma 1: The solution set to the optimization problem (2) is identical to the solution set of the optimization problem (3).

Adversarial Herd with Optimization Over Worst-k Adversaries

Duke
PRATT SCHOOL of ENGINEERING

$$\max_{\theta\in\Theta} \min_{\phi_1,\ldots,\phi_m\in\Phi} \min_{\phi\in\{\phi_i\}_{i=1}^m} R(\theta,\phi)$$



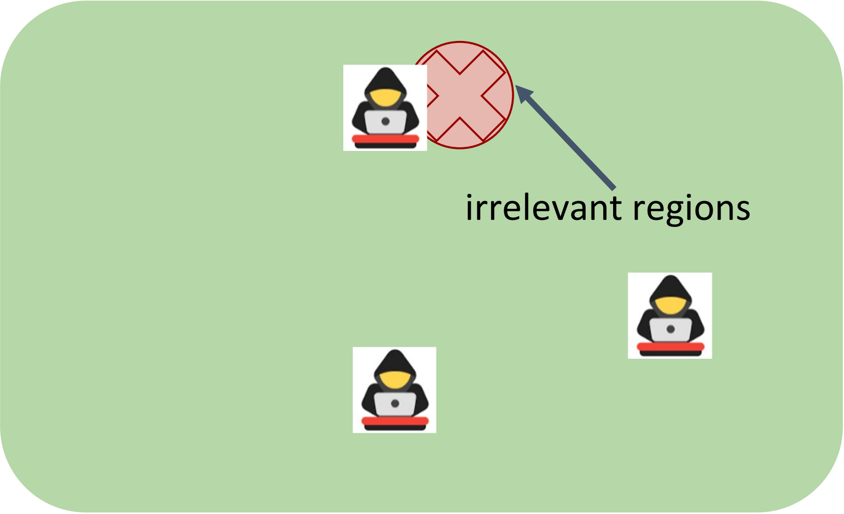
1.Efficient approximation of the inner optimization

i.e., the size of adversary herd is upper-bounded to obtain sufficient approximation precision.

Adversarial Herd with Optimization Over Worst-k Adversaries

Duke
PRATT SCHOOL of ENGINEERING

$$\max_{\theta\in\Theta} \min_{\phi_1,\ldots,\phi_m\in\Phi} \frac{1}{|I_{\theta,\widehat{\Phi},k}|} \sum_{i\in I_{\theta,\widehat{\Phi},k}} R(\theta,\phi_i)$$



2.Resolving Potential Over-Pessimism
i.e., modify the objective from optimizing its worst-case performance, to optimizing its average performance over the worst-k adversaries

Adversarial Herd with Optimization Over *Worst-k* Adversaries

Duke
PRATT SCHOOL *of* ENGINEERING

$$\max_{\theta \in \Theta} \min_{\phi_1, \ldots, \phi_m \in \Phi} \frac{1}{|I_{\theta, \widehat{\Phi}, k}|} \sum_{i \in I_{\theta, \widehat{\Phi}, k}} R(\theta, \phi_i)$$

Algorithm 1 RObust reinforcement Learning with Adversarial Herds (ROLAH)

Input: m: size of the adversarial herd ; k: the number of the worst adversaries to use; λ_p: step size for updating the agent policy; λ_a: step size for updating the adversary herd;
Output: $\widehat{\theta}$: parameter for the agent policy.

```
Randomly initialize θ and {φ_i}_{i=1}^m
t ← 0, θ^t ← θ, φ_i^t ← φ_i  ∀i ∈ [m]
for t = 0 : T − 1 do
    {Update the adversarial herd.}
    for i = 1 : m do
        Estimate R(θ^t, φ_i^t) by rolling out the agent π_{θ^t} with the adversary π_{φ_i^t}
    end for
    Construct I_{θ,Φ̂,k} with the estimations.
    φ_j^{t+1} ← φ_j^t − λ_a ∇_φ R(θ^t, φ_j^t)  ∀j ∈ I_{θ,Φ̂,k}
    {Update the agent policy.}
    for i = 1 : m do
        Estimate R(θ^t, φ_i^{t+1}) by rolling out the agent π_{θ^t} with the adversary π_{φ_i^{t+1}}
    end for
    Construct I_{θ,Φ̂,k} with the estimations.
    θ^{t+1} ← θ^t − λ_p Σ_{j∈I_{θ,Φ̂,k}} ∇_θ R(θ^t, φ_j^{t+1})
end for
θ̂ ← θ^T
```

We can use ***any*** DRL algorithms to train agent & adversary

Adversarial Herd with Optimization Over *Worst-k* Adversaries

Duke PRATT SCHOOL *of* ENGINEERING

$$\max_{\theta \in \Theta} \min_{\phi_1, \ldots, \phi_m \in \Phi} \frac{1}{|I_{\theta, \widehat{\Phi}, k}|} \sum_{i \in I_{\theta, \widehat{\Phi}, k}} R(\theta, \phi_i)$$

Algorithm 1 RObust reinforcement Learning with Adversarial Herds (ROLAH)

Input: m: size of the adversarial herd ; k: the number of the worst adversaries to use; λ_p: step size for updating the agent policy; λ_a: step size for updating the adversary herd;

Output: $\widehat{\theta}$: parameter for the agent policy.

Randomly initialize θ and $\{\phi_i\}_{i=1}^m$

$t \leftarrow 0, \theta^t \leftarrow \theta, \phi_i^t \leftarrow \phi_i \quad \forall i \in [m]$

for $t = 0 : T - 1$ **do**

{Update the adversarial herd.}

for $i = 1 : m$ **do**

Estimate $R(\theta^t, \phi_i^t)$ by rolling out the agent π_{θ^t} with the adversary $\pi_{\phi_i^t}$

end for

Construct $I_{\theta, \widehat{\Phi}, k}$ with the estimations.

$\phi_j^{t+1} \leftarrow \phi_j^t - \lambda_a \nabla_\phi R(\theta^t, \phi_j^t) \quad \forall j \in I_{\theta, \widehat{\Phi}, k}$

Train adversary

{Update the agent policy.}

for $i = 1 : m$ **do**

Estimate $R(\theta^t, \phi_i^{t+1})$ by rolling out the agent π_{θ^t} with the adversary $\pi_{\phi_i^{t+1}}$

end for

Construct $I_{\theta, \widehat{\Phi}, k}$ with the estimations.

$\theta^{t+1} \leftarrow \theta^t - \lambda_p \sum_{j \in I_{\theta, \Phi, k}} \nabla_\theta R(\theta^t, \phi_j^{t+1})$

Train agent

end for

$\widehat{\theta} \leftarrow \theta^T$

In practice, we can ensure the adversaries are distinct enough during update.

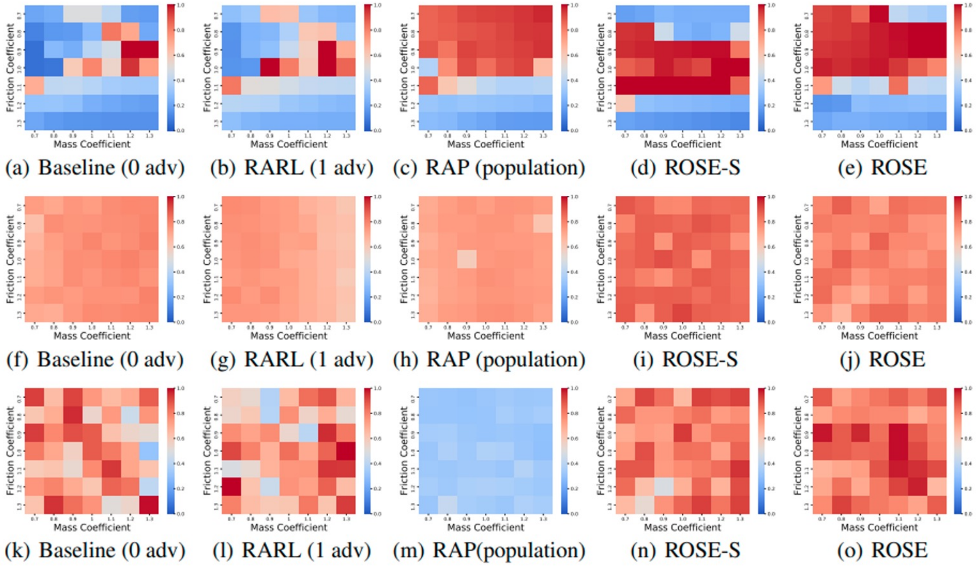
Evaluation on Standard Learning Benchmarks

1. Tasks: 5 MuJoCo environments in continuous action space
2. Core learning algorithms: TRPO (results in the slides), PPO, DDPG
3. Method comparison:
   a. Baseline (e.g., TRPO itself w/o adversarial learning) [1]
   b. RARL (1 adversary) [2]
   c. RAP (population adversaires) [3]
   d. M2TD3 (known uncertainty parameter set) [4]
   e. ROSE (ours)

1. J. Schulman et al., ”**Trust region policy optimization**”, in *ICML 2015*
2. L. Pinto et al., “**Robust Adversarial Reinforcement Learning**, in *ICML 2017*
3. E. Vinitsky et al., “**Robust reinforcement learning using adversarial populations**”, arXiv preprint arXiv:2008.01825, *2020*
4. T. Tanabe et al., “**Max-Min Off-Policy Actor-Critic Method Focusing on Worst-Case Robustness to Model Misspecification**”, in *NeurIPS, 2022*

Robustness to Test Conditions (Environmental Change)

1. Set both the friction and mass coefficients equal to 1.0 during training
2. Our method ROSE has competitive performance under varying test conditions
   a. M2TD3 is not reported because it is already provided with the uncertainty parameter set for training.
   b. Stein Variational Policy Gradient



(a) Baseline (0 adv) (b) RARL (1 adv) (c) RAP (population) (d) ROSE-S (e) ROSE

(f) Baseline (0 adv) (g) RARL (1 adv) (h) RAP (population) (i) ROSE-S (j) ROSE

(k) Baseline (0 adv) (l) RARL (1 adv) (m) RAP(population) (n) ROSE-S (o) ROSE

Robustness to Agent Disturbance

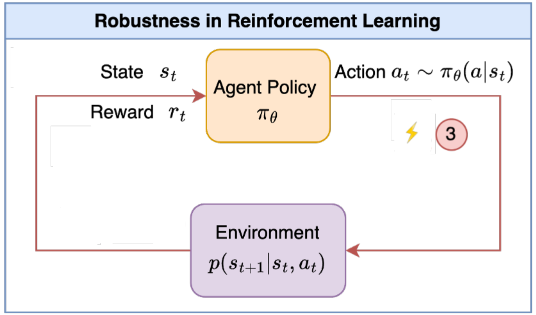
Duke PRATT SCHOOL of ENGINEERING

1. Overall, our method ROSE outperforms other methods.
2. M2TD3 is additionally provided with the uncertainty parameter set for training.
   a. ROSE still outperforms M2TD3 in most scenarios with disturbances/adversarial attacks

Method	Baseline (0 adv)	RARL (1 adv)	RAP	M2TD3 (extra info)	ROSE-S (ours)	ROSE (ours)
Ant (No disturbance)	0.77±0.16	0.81±0.12	0.83±0.08	0.84±0.22	**0.87±0.13**	0.84±0.14
Ant (Action noise)	0.66±0.19	0.67±0.16	0.67±0.09	0.66±0.16	**0.70±0.14**	0.69±0.15
Ant (Adversary)	0.21±0.18	0.25±0.17	0.30±0.14	0.29±0.11	0.38±0.16	**0.44±0.23**
InvertedPendulum (No disturbance)	**1.00±0**	0.96±0.11	0.99±0.04	**1.00±0**	0.99±0.03	0.99±0.08
InvertedPendulum (Action noise)	0.91±0.13	0.91±0.15	0.95±0.10	**0.97±0.16**	0.96±0.13	0.96±0.11
InvertedPendulum (Adversary)	0.86±0.16	0.88±0.18	0.90±0.19	0.90±0.21	0.92±0.12	**0.94±0.15**
Hopper (No disturbance)	0.78±0.003	0.79±0.02	0.84±0	0.97±0.11	0.95±0.01	**0.98±0.07**
Hopper(Action noise)	0.71±0.001	0.74±0.004	0.80±0	0.77±0.07	**0.91±0.006**	0.87±0.01
Hopper (Adversary)	0.42±0.03	0.54±0.04	0.70±0.007	0.83±0.25	0.84±0.14	**0.85±0.09**
Half-Cheetah (No disturbance)	0.77±0.05	0.72±0.03	0.76±0.02	0.81±0.06	**0.87±0.05**	0.82±0.08
Half-Cheetah(Action noise)	0.59±0.2	**0.76±0.04**	0.67±0.1	0.68±0.13	**0.76±0.16**	0.73±0.13
Half-Cheetah (Adversary)	0.16±0.1	0.19±0.05	0.24±0.36	0.50±0.10	0.52±0.21	**0.58±0.30**
Walker2d (No disturbance)	0.85±0.27	0.84±0.43	0.43±0.02	**0.88±0.31**	0.84±0.44	0.86±0.38
Walker2d (Action noise)	0.78±0.31	0.80±0.28	0.36±0.04	0.79±0.21	0.83±0.37	**0.84±0.23**
Walker2d (Adversary)	0.36±0.26	0.34±0.12	0.34±0.22	0.21±0.43	0.68±0.23	**0.70±0.17**

Robust Control Design
with 2-Player Game Design

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi)$$



$$\max_{\theta \in \Theta} \min_{\phi_1, \ldots, \phi_m \in \Phi} \frac{1}{|I_{\theta, \widehat{\Phi}, k}|} \sum_{i \in I_{\theta, \widehat{\Phi}, k}} R(\theta, \phi_i)$$

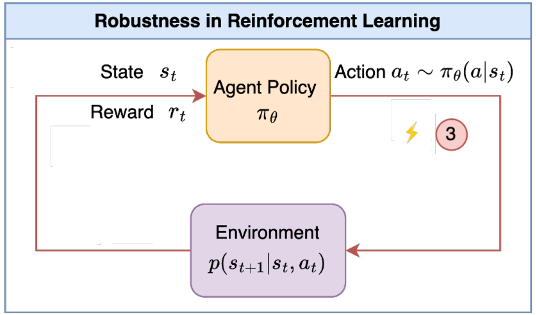
ROSE/RARL: Adversaries that incorporate domain knowledge

→ action space can be different between protagonist and adversary

Robust Control Design
with 2-Player Game Design

$$\max_{\theta \in \Theta} \min_{\phi \in \Phi} R(\theta, \phi)$$

$$\max_{\theta \in \Theta} \min_{\phi_1, \ldots, \phi_m \in \Phi} \frac{1}{|I_{\theta, \widehat{\Phi}, k}|} \sum_{i \in I_{\theta, \widehat{\Phi}, k}} R(\theta, \phi_i)$$



What if we do not have ***any*** domain knowledge for the action space?

Noisy Action Robust MDPs (NR-MDPs)

$R(\theta, \phi_i) \doteq \mathbb{E}_{s_0 \sim p_0} \left[\sum_{t=0}^{\infty} \gamma^t r(s_t, (1-\alpha) a_t^p + \alpha a_t^a) | C \right]$, where $C = \{a_t^p \sim \pi_\theta, a_t^a \sim \pi_{\phi_i}\}$

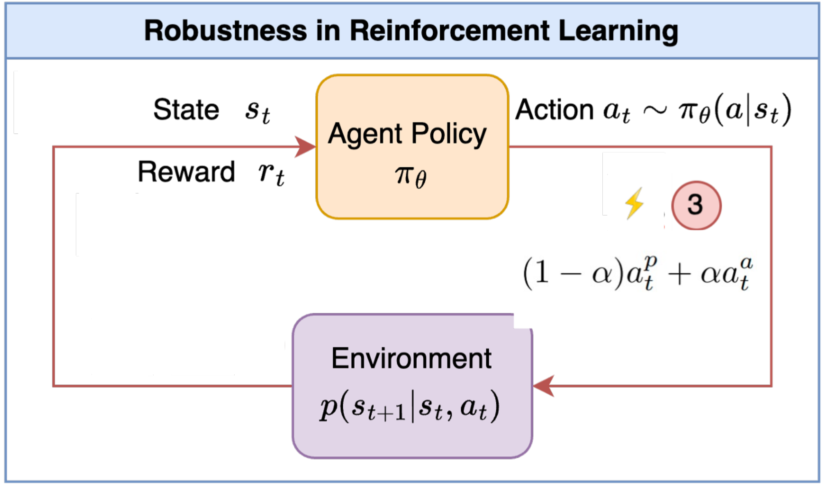
protagonist and adversary action

C. Tessler et al., "**Action Robust Reinforcement Learning and Applications in Continuous Control**", in *ICML 2019*

Noisy Action Robust MDPs (NR-MDPs)

$$R(\theta, \phi_i) \doteq \mathbb{E}_{s_0 \sim p_0}\left[\sum_{t=0}^{\infty} \gamma^t r(s_t, (1-\alpha)a_t^p + \alpha a_t^a) \middle| C\right], \text{ where } C = \{a_t^p \sim \pi_\theta, a_t^a \sim \pi_{\phi_i}\}$$

deployed action



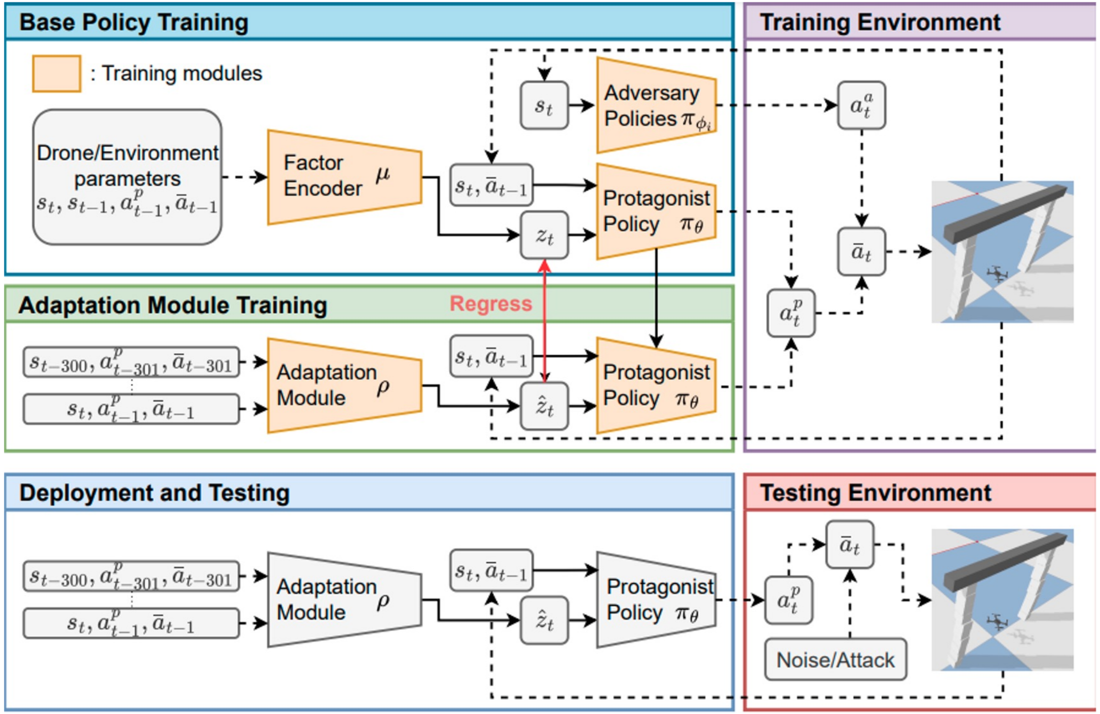
Robust Control Design
with 2-Player Game Design

1. Adversarial ensemble which involves a group of adversaries [1]
    a. Special case in noisy action robust MDP: Adaptive adversary for unknown adversary strengths [2]

2. Efficient exploration via Langevin Monte Carlo with robustness [3]

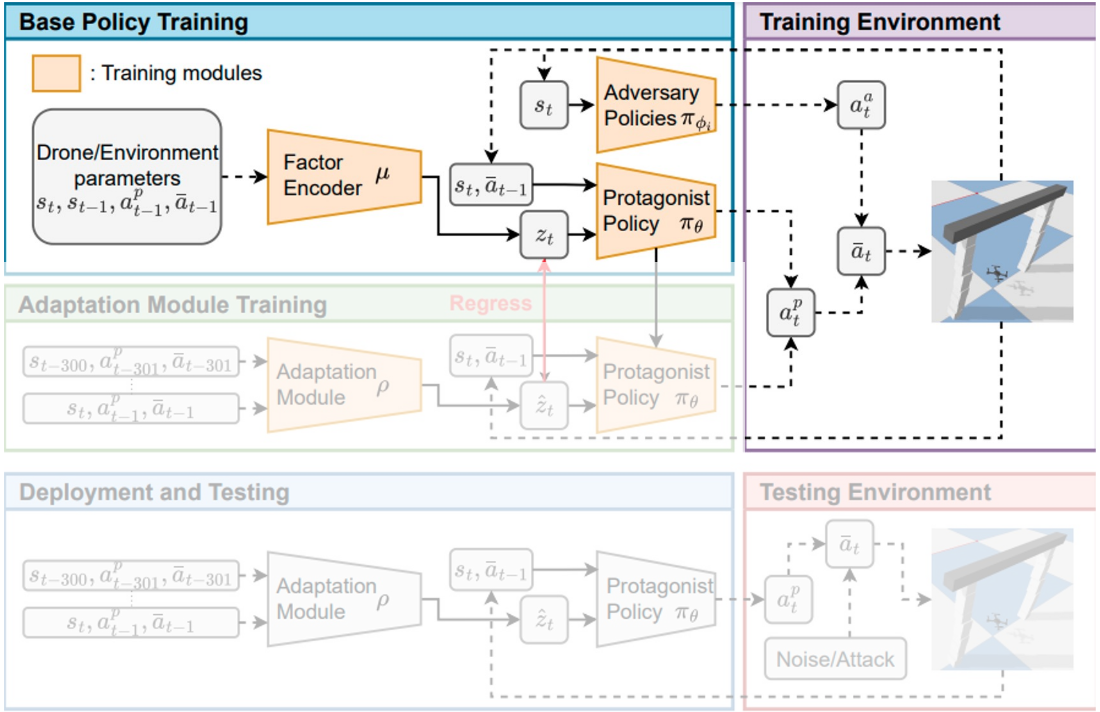
1. J Dong* and HL Hsu* et al., "Robust Reinforcement Learning through Efficient Adversarial Herding", under review, 2024.
2. HL Hsu et al., "REFORMA: Robust REinFORceMent Learning via Adaptive Adversary for Drones Flying under Disturbances" in *IEEE International Conference on Robotics and Automation* (**ICRA**), 2024.
3. HL Hsu et al., "Robust Exploration with Adversary via Langevin Monte Carlo" in *Learning for Dynamics and Control Conference* (**L4DC**), 2024

REFORMA: Robust RL via Adaptive Adversary (ICRA24)



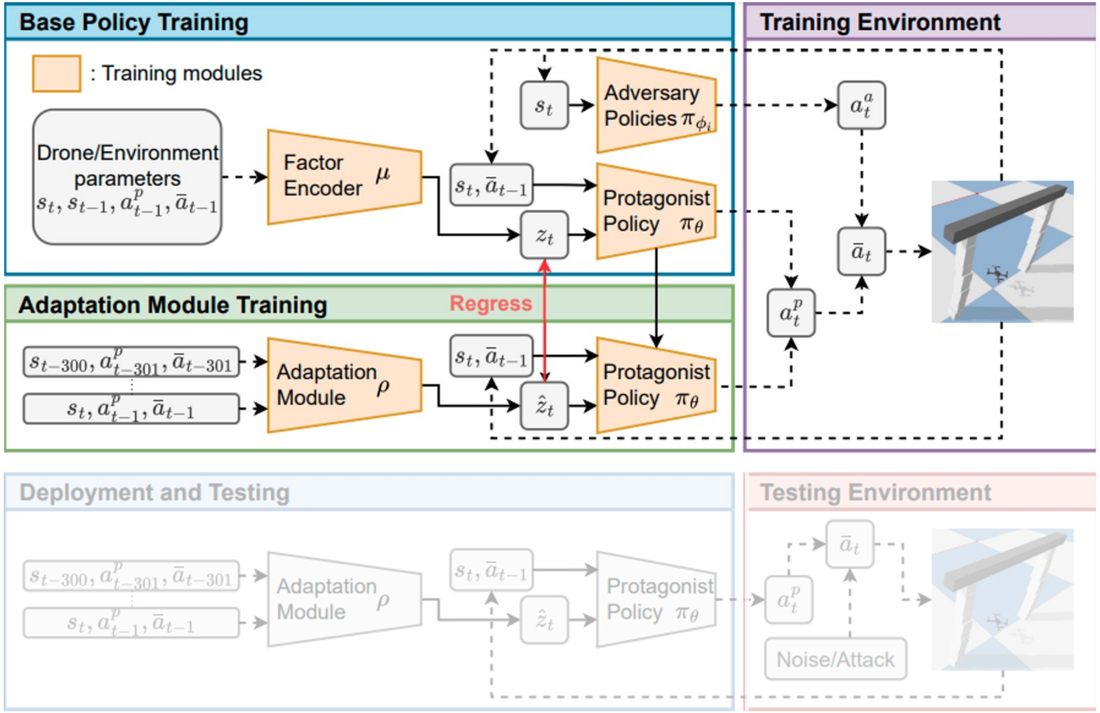
Problem: adversarial strength is unknown during evaluation

REFORMA: Robust RL via Adaptive Adversary (ICRA24)



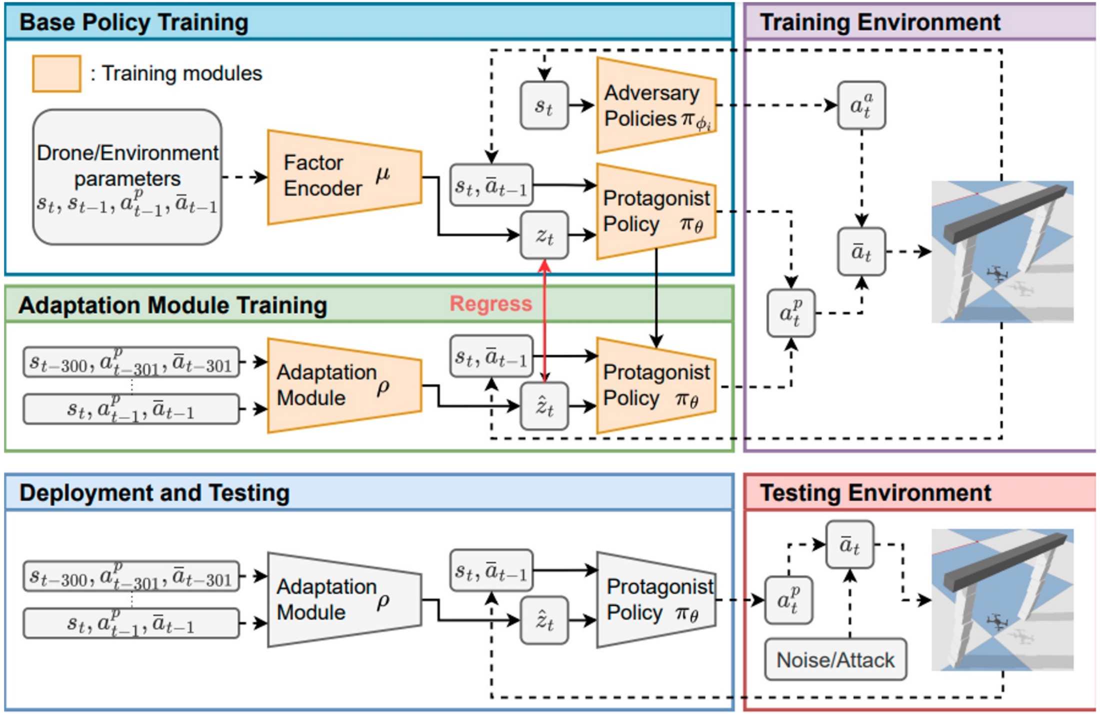
Base policy:
train protagonist and adversary policies, and factor encoder with attacked actions

REFORMA: Robust RL via Adaptive Adversary (ICRA24)



Adaptation module:
learn an adaptation module that takes state/actions history to capture drone and environment parameters.

REFORMA: Robust RL via Adaptive Adversary (ICRA24)



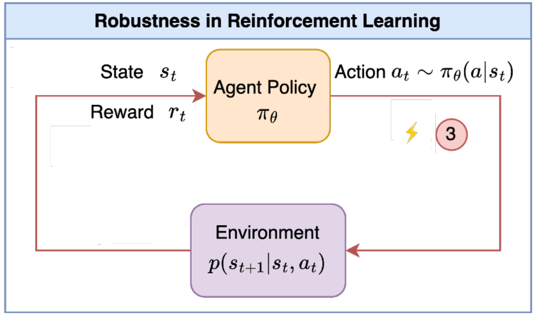
Deployment:
protagonist policy can be deployed with the inputs of the current state, previous attacked action and the latent space from adaptation module with unknown noise or attack.

Robust Control Design
with 2-Player Game Design

Duke
PRATT SCHOOL of ENGINEERING

$$\max_{\theta\in\Theta} \min_{\phi\in\Phi} R(\theta, \phi)$$

$$\max_{\theta\in\Theta} \min_{\phi_1,...,\phi_m\in\Phi} \frac{1}{|I_{\theta,\hat{\Phi},k}|} \sum_{i\in I_{\theta,\hat{\Phi},k}} R(\theta, \phi_i)$$



If the action space is discrete, improvement due to the use a group of adversaries is not obvious.

[L4DC] We also studied the better exploration strategy under adversarial training.

Summary

Duke
PRATT SCHOOL *of* ENGINEERING

- Propose robust RL via adversarial training with a group of adversaries
- Extend attackable actions in NR-MDP to adapt to a range of adversary strength
- Improve exploration under adversarial training for discrete action space using LMC

Thank you

Duke UNIVERSITY | PRATT SCHOOL of ENGINEERING