Sample Complexity of Robust Reinforcement Learning with a Generative Model

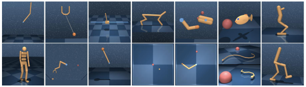
Kishan Panaganti and Dileep Kalathil

Presentation at The 25th International Conference on Artificial Intelligence and Statistics (AISTATS) 2022

March 2022

Reinforcement Learning (RL)

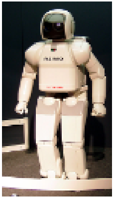
- RL algorithms have achieved some remarkable successes recently

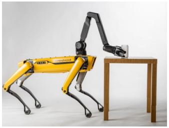


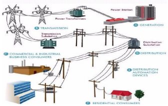
- However, most of the successful RL algorithms are limited to very structured or simulated environments

What is preventing RL from becoming the celebrated solution for real-world control systems?

Why do we need Robust RL?

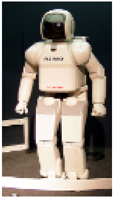


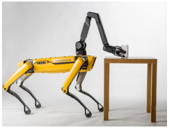


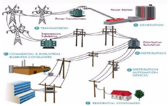


- In RL, it is nominally assumed that the testing environment is identical to the training environment (simulator model)
- However, in reality, the parameters of the simulator model can be different from the real-world systems
  - Due to the approximation errors incurred while modeling
  - Due to changes in the real-world parameters (maybe adversarial disturbances)

Why do we need Robust RL?

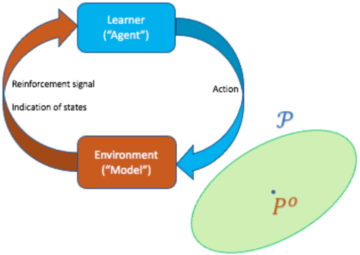






- In RL, it is nominally assumed that the testing environment is identical to the training environment (simulator model)
- However, in reality, the parameters of the simulator model can be different from the real-world systems
  - Due to the approximation errors incurred while modeling
  - Due to changes in the real-world parameters (maybe adversarial disturbances)
- In this talk: Sample complexity of Robust RL algorithm on model parameter uncertainties for real-world environments
- (Informal Robustness Gap Theorem) The worst-case performance of non-robust policy can be as bad as an arbitrary policy in an order sense

Robust ~~Classical~~ MDP Formulation



- Robust MDP $= \{\mathcal{S}, \mathcal{A}, \mathcal{P}, r\}$
  - $\mathcal{P}_{s,a} = \{P \in \Delta^{|\mathcal{S}|} \ : \ D(P_{s,a}, P^o_{s,a}) \leq c_r\}$.
  - $D =$ TV, Chi-square, KL
  - P^o (accessible simulator model)

Robust MDP objective

$$\max_\pi \ \min_{P \in \mathcal{P}} \mathbb{E}_P[\ \sum_{t=0}^\infty \alpha^t r(s_t, \pi(s_t))\ ], \qquad 0 < \alpha < 1$$

Find policy that performs best under the worst model.

Dynamic Programming for Robust MDP

- Robust value function: $V_\pi(s) = \min_{P \in \mathcal{P}} \mathbb{E}_P[\sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t)) \mid s_0 = s]$.
- To find: V^* and π^*.

Dynamic Programming for Robust MDP

- Robust value function: $V_\pi(s) = \min_{P \in \mathcal{P}} \mathbb{E}_P[\sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t)) \mid s_0 = s]$.
- To find: V^* and π^*. When $\mathcal{P}$ is known: Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary deterministic policies
- Optimal robust value function V^*, solved by iterating

$$V_{k+1}(s) = \max_a \ (r(s,a) + \alpha \min_{P \in \mathcal{P}} \sum_{s'} P_{s,a}(s') V_k(s'))$$

- Optimal robust (stationary) policy π^*, solved by

$$\pi^*(s) = \arg\max_a \ (r(s,a) + \alpha \min_{P \in \mathcal{P}} \sum_{s'} P_{s,a}(s') V^*(s'))$$

- Also solved by *Robust policy iteration* (Iyengar, 2005)

Robust RL Problem

- Main goal: Find robust optimal policy π^* when $\mathcal{P}$ is unknown.

$$\pi^* = \arg\max_{\pi} \min_{P \in \mathcal{P}} \mathbb{E}_P[\ \sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t))\ ]$$

Robust RL Problem

- Main goal: Find robust optimal policy π^* when $\mathcal{P}$ is unknown.

$$\pi^* = \arg\max_{\pi} \min_{P \in \mathcal{P}} \mathbb{E}_P[\ \sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t))\ ]$$

- Challenge: Generating samples according to each and every P in $\mathcal{P}$ is clearly infeasible.

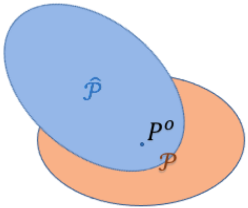
Robust RL Problem

- Main goal: Find robust optimal policy π^* when $\mathcal{P}$ is unknown.

$$\pi^* = \arg\max_{\pi} \min_{P \in \mathcal{P}} \mathbb{E}_P[\sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t))]$$

- Challenge: Generating samples according to each and every P in $\mathcal{P}$ is clearly infeasible.

- Algorithm only has access to a simulator (nominal) model P^o



Robust RL Problem

- Main goal: Find robust optimal policy π^* when $\mathcal{P}$ is unknown.

$$\pi^* = \arg\max_{\pi} \min_{P \in \mathcal{P}} \mathbb{E}_P[\ \sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t))\ ]$$

- Challenge: Generating samples according to each and every P in $\mathcal{P}$ is clearly infeasible.
- Algorithm only has access to a simulator (nominal) model P^o
    - Solution: We use generative sampling model to approximate P^o and thereby approximating $\mathcal{P}$
        - For all (s, a), simulator model gives $s' \sim P^o_{s,a}(\cdot)$ and $r(s, a)$
        - With N samples for each (s, a), estimate $P^o_{s,a}$ as

$$\widehat{P}^o_{s,a}(s') = \frac{N(s, a, s')}{N}$$

REVI Algorithm

Denote $\sigma_{\widehat{\mathcal{P}}_{s,a}}(v) = \min\{u^\top v : u \in \widehat{\mathcal{P}}_{s,a}\}$.

$$\widehat{\mathcal{P}}_{s,a} = \{P \in \Delta^{|\mathcal{S}|} \ : \ D(P_{s,a}, \widehat{P}^o_{s,a}) \leq c_r\}.$$

Robust Empirical Value Iteration (REVI) Algorithm

Input: Loop termination number k.
Initialize: $Q_0 = 0$
for $i = 0, \cdots, k-1$ **do**
    $\forall\ (s,a),\ Q_{i+1}(s,a) = r(s,a) + \gamma \sigma_{\widehat{\mathcal{P}}_{s,a}}(V_i)$, where $V_i(s) = \max_a Q_i(s,a)$
end for
Output: π_k, where $\pi_k(s) = \arg\max_a Q_k(s,a), \forall s \in \mathcal{S}$

REVI Result

PAC guarantee: $\|V^* - V^{\pi_K}\| \leq \epsilon$ with probability at least $1-\delta$.
ϵ-range is $(0, \frac{\mathcal{O}(1)}{1-\gamma})$

Uncertainty set	Sample Complexity
TV	$\mathcal{O}(\frac{\lvert\mathcal{S}\rvert^2 \lvert\mathcal{A}\rvert}{\epsilon^2 (1-\gamma)^4} \log \frac{\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert}{\delta\epsilon})$
Chi-square	$\mathcal{O}(\frac{\lvert\mathcal{S}\rvert^2 \lvert\mathcal{A}\rvert}{\epsilon^2 (1-\gamma)^4} \log \frac{\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert}{\delta\epsilon})$
KL	$\mathcal{O}(\frac{\lvert\mathcal{S}\rvert^2 \lvert\mathcal{A}\rvert e^{1/(1-\gamma)}}{\epsilon^2 (1-\gamma)^4} \log \frac{\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert}{\delta\epsilon})$
Non-robust	$\mathcal{O}(\frac{\lvert\mathcal{S}\rvert \lvert\mathcal{A}\rvert}{\epsilon^2 (1-\gamma)^3} \log \frac{\lvert\mathcal{S}\rvert\lvert\mathcal{A}\rvert}{\delta\epsilon})$

$K = \mathcal{O}(\frac{1}{\log(1/\gamma)} \log(\frac{\gamma}{\epsilon(1-\gamma)^2}))$

REVI Result

PAC guarantee: $\|V^* - V^{\pi_K}\| \leq \epsilon$ with probability at least $1-\delta$.
ϵ-range is $(0, \frac{\mathcal{O}(1)}{1-\gamma})$

Uncertainty set	Sample Complexity
TV	$\mathcal{O}(\frac{\|\mathcal{S}\|^2\|\mathcal{A}\|}{\epsilon^2(1-\gamma)^4}\log\frac{\|\mathcal{S}\|\|\mathcal{A}\|}{\delta\epsilon})$
Chi-square	$\mathcal{O}(\frac{\|\mathcal{S}\|^2\|\mathcal{A}\|}{\epsilon^2(1-\gamma)^4}\log\frac{\|\mathcal{S}\|\|\mathcal{A}\|}{\delta\epsilon})$
KL	$\mathcal{O}(\frac{\|\mathcal{S}\|^2\|\mathcal{A}\|e^{1/(1-\gamma)}}{\epsilon^2(1-\gamma)^4}\log\frac{\|\mathcal{S}\|\|\mathcal{A}\|}{\delta\epsilon})$
Non-robust	$\mathcal{O}(\frac{\|\mathcal{S}\|\|\mathcal{A}\|}{\epsilon^2(1-\gamma)^3}\log\frac{\|\mathcal{S}\|\|\mathcal{A}\|}{\delta\epsilon})$

$K = \mathcal{O}(\frac{1}{\log(1/\gamma)}\log(\frac{\gamma}{\epsilon(1-\gamma)^2}))$

- Open question: Can we achieve this in the Robust RL setting?

Proof Idea

- We split $\|V^* - V^{\pi_K}\|$ into three terms as
$\|V^* - V^{\pi_K}\| \leq \underbrace{\|\widehat{V}^* - \widehat{V}^{\pi_K}\|}_{\text{I}} + \underbrace{\|V^* - \widehat{V}^*\|}_{\text{II}} + \underbrace{\|\widehat{V}^{\pi_K} - V^{\pi_K}\|}_{\text{III}}$
- Bounding I: From the contraction property of the robust Bellman operator, we can show that $\|\widehat{V}^* - \widehat{V}^{\pi_{k+1}}\| \leq \gamma \|\widehat{V}^* - \widehat{V}^{\pi_k}\|$ for any k.
- This exponential convergence, with some additional results from the MDP theory, gets us $\|\widehat{V}^* - \widehat{V}^{\pi_K}\| \leq 2\gamma^{K+1}/(1-\gamma)^2$.

Proof Idea

- We split $\|V^* - V^{\pi_K}\|$ into three terms as
  $\|V^* - V^{\pi_K}\| \leq \underbrace{\|\widehat{V}^* - \widehat{V}^{\pi_K}\|}_{\text{I}} + \underbrace{\|V^* - \widehat{V}^*\|}_{\text{II}} + \underbrace{\|\widehat{V}^{\pi_K} - V^{\pi_K}\|}_{\text{III}}$
- Bounding II: For any state s, and denoting $a = \pi^*(s)$, we get

$$V^*(s) - \widehat{V}^*(s) \leq \underbrace{\gamma(\sigma_{\mathcal{P}_{s,a}}(V^*) - \sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*))}_{\leq \gamma \|V^* - \widehat{V}^*\|} + \underbrace{\gamma(\sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*) - \sigma_{\widehat{\mathcal{P}}_{s,a}}(\widehat{V}^*))}_{\leq \mathcal{O}(\frac{1}{(1-\gamma)^2}\sqrt{\frac{|\mathcal{S}|}{N}\log\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^2\delta\epsilon}})}$$

- Bounding the last term is non-trivial that requires more work than the non-robust setting since $\mathbb{E}[\sigma_{\widehat{\mathcal{P}}_{s,a}}(\widehat{V}^*)] \neq \sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*)$

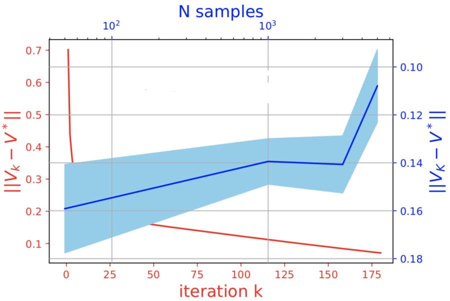
Proof Idea

- We split $\|V^* - V^{\pi_K}\|$ into three terms as
$\|V^* - V^{\pi_K}\| \le \underbrace{\|\widehat{V}^* - \widehat{V}^{\pi_K}\|}_{\text{I}} + \underbrace{\|V^* - \widehat{V}^*\|}_{\text{II}} + \underbrace{\|\widehat{V}^{\pi_K} - V^{\pi_K}\|}_{\text{III}}$
- Bounding II: For any state s, and denoting $a = \pi^*(s)$, we get

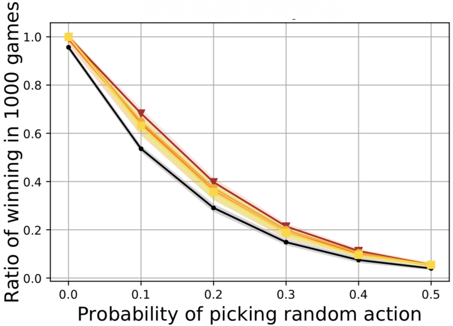
$$V^*(s) - \widehat{V}^*(s) \le \underbrace{\gamma(\sigma_{\mathcal{P}_{s,a}}(V^*) - \sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*))}_{\le \gamma\|V^* - \widehat{V}^*\|} + \underbrace{\gamma(\sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*) - \sigma_{\widehat{\mathcal{P}}_{s,a}}(\widehat{V}^*))}_{\le \mathcal{O}(\frac{1}{(1-\gamma)^2}\sqrt{\frac{|\mathcal{S}|}{N}\log\frac{|\mathcal{S}||\mathcal{A}|}{(1-\gamma)^2\delta\epsilon}})}$$

- Bounding the last term is non-trivial that requires more work than the non-robust setting since $\mathbb{E}[\sigma_{\widehat{\mathcal{P}}_{s,a}}(\widehat{V}^*)] \neq \sigma_{\mathcal{P}_{s,a}}(\widehat{V}^*)$
- Bounding III: Following II, we need a uniform bound on the bounded value functional class with $\|V\| \le 1/(1-\gamma)$.

- We show convergence of our algorithm on FrozenLake8x8 environment in OpenAI Gym with default parameters
- We test the above policy on a test environment



Learning curve of the error $\|V_k - V^*\|$.
Performance curve versus N.



Thank you for listening!

References I

Iyengar, Garud N (2005). "Robust dynamic programming". In: *Mathematics of Operations Research* 30.2, pp. 257–280.

Nilim, Arnab and Laurent El Ghaoui (2005). "Robust control of Markov decision processes with uncertain transition matrices". In: *Operations Research* 53.5, pp. 780–798.

Panaganti, Kishan and Dileep Kalathil (2021). "Robust Reinforcement Learning using Least Squares Policy Iteration with Provable Performance Guarantees". In: *Proceedings of the 38th International Conference on Machine Learning*, pp. 511–520.