Robust Reinforcement Learning using Least Squares Policy Iteration with Provable Performance Guarantees

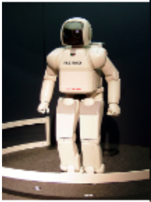
Kishan Panaganti and Dileep Kalathil

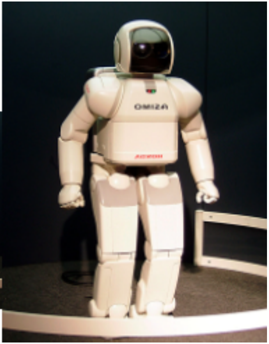
TEXAS A&M UNIVERSITY

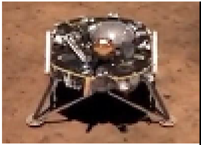
Short Presentation at ICML 2021

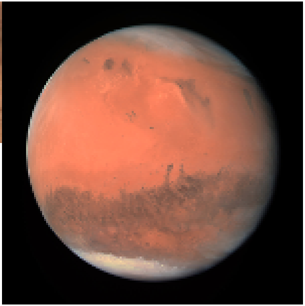
July 2021

Motivation - Why should we be robust?





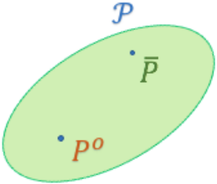




This paper: Robustness for model parameter uncertainty

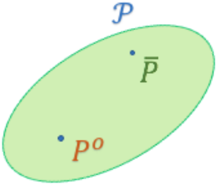
https://en.wikipedia.org/wiki/Robot, https://en.wikipedia.org/wiki/Planet

Main “informal” question



Question: Can we promise robustness when the “test” model is $\bar{P}$?

Main "informal" question



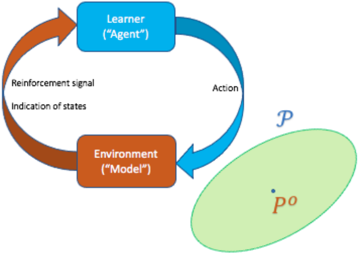
Question: Can we promise robustness when the "test" model is $\bar{P}$?

We develop a model-free RL algorithm that learns a policy that is robust against parameter uncertainty

We provide provable convergence guarantees for the proposed model-free RL algorithm (Policy Evaluation + Policy Iteration)

We verify the algorithm in simulation on OpenAIGym (Brockman et al., 2016)

Robust ~~Classical~~ MDP Formulation

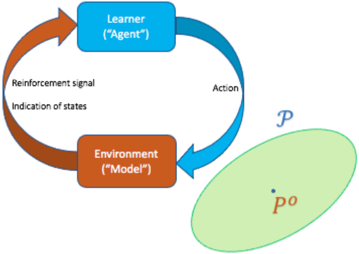


Robust MDP $= \{\mathcal{S}, \mathcal{A}, \mathcal{P}, r\}$ **This paper**

Let $\mathcal{P} = P^o + \mathcal{U}$, $\mathcal{U}$ is the parameter uncertainty set.
[indexed by (s,a)] $P^o \in \mathcal{P}$

States $\mathcal{S}$, actions $\mathcal{A}$, rewards r are known

Robust ~~Classical~~ MDP Formulation



Robust MDP $= \{\mathcal{S}, \mathcal{A}, \mathcal{P}, r\}$ **This paper**

Let $\mathcal{P} = P^o + \mathcal{U}$, $\mathcal{U}$ is the parameter uncertainty set.
[indexed by (s,a)] $P^o \in \mathcal{P}$

States $\mathcal{S}$, actions $\mathcal{A}$, rewards r are known

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 policy evaluation for fixed policy π. Robust value function: $V_\pi(s) = \min_{P \in \mathcal{P}} \mathbb{E}_P[\sum_{t=0}^{\infty} \alpha^t r(s_t, \pi(s_t)) \mid s_0 = s]$.

Robust Bellman operator for Robust PE

$$T_\pi(V_\pi(s)) = r(s, \pi(s)) + \alpha \min_{P \in \mathcal{P}} \sum_{s'} P_{s,\pi(s)}(s') V_\pi(s')$$

- Optimal robust policy and value: $\pi^* = \arg\max_\pi V_\pi$ and $V^* = \max_\pi V_\pi$

Dynamic Programming for Robust MDP

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

Robust Bellman operator for Robust PE

$$T_\pi(V_\pi(s)) = r(s, \pi(s)) + \alpha \min_{P \in \mathcal{P}} \sum_{s'} P_{s,\pi(s)}(s') V_\pi(s')$$

- Optimal robust policy and value: $\pi^* = \arg\max_\pi V_\pi$ and $V^* = \max_\pi V_\pi$
- Hard problem because of $\min_{P \in \mathcal{P}}$
- **Question**: How do we compute V^* and π^*?

Dynamic Programming for Robust MDP

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

Robust Bellman operator for Robust PE

$$T_\pi(V_\pi(s)) = r(s, \pi(s)) + \alpha \min_{P \in \mathcal{P}} \sum_{s'} P_{s,\pi(s)}(s') V_\pi(s')$$

- Optimal robust policy and value: $\pi^* = \arg\max_\pi V_\pi$ and $V^* = \max_\pi V_\pi$
- Hard problem because of $\min_{P \in \mathcal{P}}$
- **Question**: How do we compute V^* and π^*?
- Solved by *Robust policy iteration* (Iyengar, 2005), *Robust value iteration* (Nilim and El Ghaoui, 2005)

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*),

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under “rectangularity” condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point.

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point. Solved by iterating

$$V_{k+1} = T_\pi(V_k)$$

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point. Solved by iterating
$$V_{k+1} = T_\pi(V_k)$$
- Define $T(V) = \max_\pi T_\pi(V)$.

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point. Solved by iterating
$$V_{k+1} = T_\pi(V_k)$$
- Define $T(V) = \max_\pi T_\pi(V)$. T is a contraction in sup norm and V^* is its unique fixed point

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point. Solved by iterating
$$V_{k+1} = T_\pi(V_k)$$
- Define $T(V) = \max_\pi T_\pi(V)$. T is a contraction in sup norm and V^* is its unique fixed point
- Optimal robust (stationary) policy π^* satisfies
$$\pi^* = \arg\max_\pi T_\pi(V^*)$$

DP for Robust MDP

- Solved by *Robust policy iteration* (Iyengar, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- T_π is a contraction in sup norm and V_π is its unique fixed point. Solved by iterating
$$V_{k+1} = T_\pi(V_k)$$
- Define $T(V) = \max_\pi T_\pi(V)$. T is a contraction in sup norm and V^* is its unique fixed point
- Optimal robust (stationary) policy π^* satisfies
$$\pi^* = \arg\max_\pi T_\pi(V^*)$$
- Also solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*),

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- Optimal robust value function V^*,

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- Optimal robust value function V^*, solved by iterating

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

- Optimal robust (stationary) policy π^*,

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- 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

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

DP for Robust MDP

- Solved by *Robust value iteration* (Nilim and El Ghaoui, 2005)
- Under "rectangularity" condition (*uncorrelated uncertainties across (s,a)*), it suffices to consider stationary control policies **and** stationary nature uncertain models
- 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

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

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

Robust RL - Main Challenges

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

Robust RL - Main Challenges

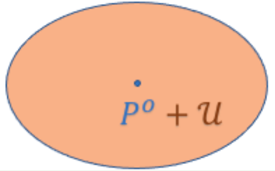
Main goal: Find robust optimal policy π^* when $\mathcal{P}$ is unknown.

Challenge: Recall $\mathcal{P} = P^o + \mathcal{U}$. When P^o is known, we can construct $\mathcal{U}$ such that $\mathcal{P}$ is a valid collection of probability vectors.

Robust RL - Main Challenges

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

Challenge: Recall $\mathcal{P} = P^o$ n, we can construct $\mathcal{U}$ such that $\mathcal{P}$ is a valid collection of p



Example (Spherical ur

$$\mathcal{U} := \{x \mid \|x\|_2 \leq 1, \sum_{s \in \mathcal{S}} x_s = 0, -P^o(s') \leq x_{s'} \leq 1 - P^o(s'), \forall s' \in \mathcal{S}\}$$

Robust RL - Main Challenges

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

Challenge: Recall $\mathcal{P} = P^o + \mathcal{U}$. When P^o is known, we can construct $\mathcal{U}$ such that $\mathcal{P}$ is a valid collection of probability vectors.

Example (Spherical uncertainty set)

$$\mathcal{U} := \{x \mid \|x\|_2 \leq 1, \sum_{s \in \mathcal{S}} x_s = 0, -P^o(s') \leq x_{s'} \leq 1 - P^o(s'), \forall s' \in \mathcal{S}\}$$

But, we do not know P^o. So, we approximate the uncertainty set as $\widehat{\mathcal{U}}$.

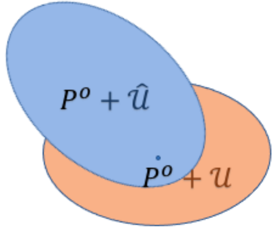
Robust RL - Main Challenges

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

Challenge: Recall $\mathcal{P} = P^o$ n, we can construct $\mathcal{U}$ such that $\mathcal{P}$ is a valid collection of p

Example (Spherical un

$\mathcal{U} := \{x \mid \|x\|_2 \leq$ $, \leq 1 - P^o(s'), \forall s' \in \mathcal{S}\}$



But, we do not know P^o. So, we approximate the uncertainty set as $\widehat{\mathcal{U}}$.

Example (Spherical “approximate” uncertainty set)

$$\widehat{\mathcal{U}} := \{x \mid \|x\|_2 \leq 1, \textstyle\sum_{s \in \mathcal{S}} x_s = 0\}$$

Robust RL - Main Challenges

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

Challenge: Recall $\mathcal{P} = P^o + \mathcal{U}$. When P^o is known, we can construct $\mathcal{U}$ such that $\mathcal{P}$ is a valid collection of probability vectors.

Example (Spherical uncertainty set)

$$\mathcal{U} := \{x \mid \|x\|_2 \leq 1, \sum_{s \in \mathcal{S}} x_s = 0, -P^o(s') \leq x_{s'} \leq 1 - P^o(s'), \forall s' \in \mathcal{S}\}$$

But, we do not know P^o. So, we approximate the uncertainty set as $\widehat{\mathcal{U}}$.

Example (Spherical “approximate” uncertainty set)

$$\widehat{\mathcal{U}} := \{x \mid \|x\|_2 \leq 1, \sum_{s \in \mathcal{S}} x_s = 0\}$$

Challenge: We only get samples from P^o, and not from every $P \in \mathcal{P}$.

Key tools used in this paper

- Additional challenge: Large scale problems incur **“curse of dimensionality”**

Key tools used in this paper

- Additional challenge: Large scale problems incur **"curse of dimensionality"**
- Two totems to address this curse **This paper**

Key tools used in this paper

- Additional challenge: Large scale problems incur **"curse of dimensionality"**
- Two totems to address this curse **This paper**

Linear function approximation for $V_\pi(s)$

Given state-dependent features $\phi(s) \in \mathbb{R}^L, L << |\mathcal{S}|$: $\bar{V}_\pi(s) = \phi(s)^\top w_\pi$

generalization capabilities (Tamar et al., 2014; Lim and Autef, 2019; Panaganti and Kalathil, 2020)

Robust TD(λ) operator

$$T_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty} \lambda^m T_\pi^{m+1}(V), \qquad \lambda \in [0,1)$$

multi-step boosting (Van Seijen et al., 2016; Altahhan, 2020; Panaganti and Kalathil, 2020)

Robust Policy Evaluation Challenge

- $T_{\pi}^{(\lambda)}$ is nonlinear and very difficult to estimate
  Denoting $\sigma_{\mathcal{B}}(v) = \min\{u^{\top} v : u \in \mathcal{B}\}$,

$$T_{\pi}^{(\lambda)}(V) = (1-\lambda) \sum_{m=0}^{\infty} \lambda^{m} \left\{ \sum_{k=0}^{m} (\alpha P_{\pi}^{o})^{k} r_{\pi} + (\alpha P_{\pi}^{o})^{m+1} V + \alpha \sum_{k=0}^{m} (\alpha P_{\pi}^{o})^{k} \sigma_{\mathcal{U}_{\pi}} (T_{\pi}^{(m-k)} V) \right\}$$

Robust Policy Evaluation Challenge

- $T_\pi^{(\lambda)}$ is nonlinear and very difficult to estimate
  Denoting $\sigma_{\mathcal{B}}(v) = \min\{u^\top v : u \in \mathcal{B}\}$,

$$T_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m \left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(T_\pi^{(m-k)}V)\right\}$$

- We propose an "approximate" robust TD(λ) operator:

$$\widetilde{T}_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m \left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(\cancel{T_\pi^{(m-k)}}V)\right\}$$

- This is a tractable and "good" approximation

Robust Policy Evaluation Challenge

- $T_\pi^{(\lambda)}$ is nonlinear and very difficult to estimate

  Denoting $\sigma_{\mathcal{B}}(v) = \min\{u^\top v : u \in \mathcal{B}\}$,

$$T_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m\left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(T_\pi^{(m-k)}V)\right\}$$

- We propose an "approximate" robust TD(λ) operator:

$$\widetilde{T}_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m\left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(\cancel{T_\pi^{(m-k)}}V)\right\}$$

- This is a tractable and "good" approximation
- We still have $\widetilde{T}_\pi^{(\lambda)}(V_\pi) = V_\pi$!

Robust Policy Evaluation Challenge

- $T_\pi^{(\lambda)}$ is nonlinear and very difficult to estimate
  Denoting $\sigma_{\mathcal{B}}(v) = \min\{u^\top v : u \in \mathcal{B}\}$,

$$T_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m\left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(T_\pi^{(m-k)}V)\right\}$$

- We propose an "approximate" robust TD(λ) operator:

$$\widetilde{T}_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m\left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\mathcal{U}_\pi}(\cancel{T_\pi^{(m-k)}}V)\right\}$$

- This is a tractable and "good" approximation
- We still have $\widetilde{T}_\pi^{(\lambda)}(V_\pi) = V_\pi$!
- For the RL setting:

$$\widetilde{T}_\pi^{(\lambda)}(V) = (1-\lambda)\sum_{m=0}^{\infty}\lambda^m\left\{\sum_{k=0}^{m}(\alpha P_\pi^o)^k r_\pi + (\alpha P_\pi^o)^{m+1}V + \alpha\sum_{k=0}^{m}(\alpha P_\pi^o)^k \sigma_{\widehat{\mathcal{U}}_\pi}(\cancel{T_\pi^{(m-k)}}V)\right\}$$

RLSPI Algorithm gist

Π does a "projection operation" onto the subspace spanned by the columns of Φ under a weighted norm described by the steady state distribution of $P^o_{\pi_k}$

Robust Least Squares Policy Iteration (RLSPI) Algorithm

1. **(Initialization)** Initial policy π_0 and weights w_0
2. **(Robust Least Squares Policy Evaluation (RLSPE))** Given the policy π_k, solve for the approximate robust value function $\bar{V}_{\pi_k} = \Phi w_{\pi_k}$ using

$$\Phi w_{\pi_k} = \Pi \widetilde{T}^{(\lambda)}_{\pi_k} \Phi w_{\pi_k}$$

3. **(Robust Least Squares Policy Iteration (RLSPI))** Obtain a new policy

$$\pi_{k+1} = \arg\max_{\pi} \widetilde{T}^{(\lambda)}_{\pi}(\Phi w_{\pi_k})$$

repeat...

RLSPI Algorithm

RLSPE from Stochastic Approximation theory

$$w_{t+1} = w_t + \gamma_t B_t^{-1}(A_t w_t + b_t + C_t(w_t)),$$

where,

$$A_t = \frac{1}{t+1}\sum_{\tau=0}^{t} z_\tau \; (\alpha\phi^\top(s_{\tau+1}) - \phi^\top(s_\tau)), \qquad B_t = \frac{1}{t+1}\sum_{\tau=0}^{t} \phi(s_\tau)\phi^\top(s_\tau),$$

$$C_t(w) = \frac{\alpha}{t+1}\sum_{\tau=0}^{t} z_\tau \; \sigma_{\widehat{\mathcal{U}}_{s_\tau, \pi(s_\tau)}}(\Phi w),$$

$$b_t = \frac{1}{t+1}\sum_{\tau=0}^{t} z_\tau r(s_\tau, \pi(s_\tau)), \qquad z_\tau = \sum_{m=0}^{\tau} (\alpha\lambda)^{\tau-m}\phi(s_m)$$

RLSPI

$$\pi_{k+1} = \arg\max_{\pi} \widetilde{T}_\pi^{(\lambda)}(\Phi w_{k+1}) \xrightarrow{\phi(s) \to \phi(s,a)} \pi_{k+1}(.) = \arg\max_{a \in \mathcal{A}} \phi(., a)^\top w_{\pi_k}$$

RLSPI Algorithm

Pseudocode

```
Initialization: Policy evaluation weights error $\epsilon_0$, initial policy $\pi_0$.
for $k = 0 \dots K$ do
    Initialize the policy weight vector $w_0$. Initialize time step $t \leftarrow 0$.
    repeat
        Observe $s_t$, take $a_t = \pi_k(s_t)$, observe $r_t$ and $s_{t+1}$.
        Update the weight vector $w_t$
        $t \leftarrow t + 1$
    until $\|w_t - w_{t-1}\|_2 < \epsilon_0$
    $w_{\pi_k} \leftarrow w_t$
    Update the policy $\pi_{k+1}(s) = \arg\max_{a \in \mathcal{A}} \phi(s, a)^\top w_{\pi_k}$
end for
```

Convergence of RLSPE: Results

Assumptions

(i) $\alpha P_{s,\pi(s)}(s') \leq \beta P^{o}_{s,\pi(s)}(s')$ (ii) steady-state distribution $d > 0$ on P^{o}_{π}

Define

$$\rho = \text{distance}(\mathcal{U}, \widehat{\mathcal{U}}), \text{ an Unknown uncertainty error}$$
$$c(\alpha, \beta, \rho, \lambda) = (\beta(2-\lambda) + \rho\alpha)/(1-\beta\lambda)$$

Theorem (Convergence of RLSPE for policy π)

Let V_π be the true robust value function for policy π. Let Φw_π be the approximate robust value function for policy π.

- *if $c(\alpha, \beta, \rho, \lambda) < 1$, $\exists\ !\ w_\pi$ for $\Phi w_\pi = \Pi \widetilde{T}^{(\lambda)}_{\pi}(\Phi w_\pi)$*
- *(Stochastic Approximation theory) w_t converges to w_π w.p. 1.*

Convergence of RLSPE: Results

Assumptions

(i) $\alpha P_{s,\pi(s)}(s') \leq \beta P^{o}_{s,\pi(s)}(s')$ (ii) steady-state distribution $d > 0$ on P^{o}_{π}

Define

$$\rho = \text{distance}(\mathcal{U}, \widehat{\mathcal{U}}), \text{ an Unknown uncertainty error}$$
$$c(\alpha, \beta, \rho, \lambda) = (\beta(2-\lambda) + \rho\alpha)/(1-\beta\lambda)$$

Theorem (Convergence of RLSPE for policy π)

Let V_{π} be the true robust value function for policy π. Let Φw_{π} be the approximate robust value function for policy π.

- *if $c(\alpha, \beta, \rho, \lambda) < 1$, $\exists\, !\ w_{\pi}$ for $\Phi w_{\pi} = \Pi \widetilde{T}^{(\lambda)}_{\pi}(\Phi w_{\pi})$*
- *(Stochastic Approximation theory) w_t converges to w_{π} w.p. 1. Moreover,*

$$\|V_{\pi} - \Phi w_{\pi}\|_{d} \leq \frac{1}{1-c(\alpha,\beta,\rho,\lambda)} \left(\|V_{\pi} - \Pi V_{\pi}\|_{d} + \frac{\beta\rho\|V_{\pi}\|_{d}}{1-\beta\lambda} \right).$$

Linear FA error $\leq$ Projection error + Unknown uncertainty error

Convergence of RLSPI: Results

Assumptions

(i) $\max_{\pi} \|V_{\pi} - \Pi_{d_{\pi}} V_{\pi}\|_{d_{\pi}} < \delta$ (ii) $d_{\pi} \geq \bar{\mu}/C_2 \geq \mu H(\pi, P^o)/C_1 C_2$

Theorem (Asymptotic convergence of PI)

Let V^* be the optimal robust value function. $\{\pi_k\}$ policy sequence of algorithm. Let V_{π_k} be the true robust value function for policy π_k. Let $\rho = 0$.

$$\limsup_{k \to \infty} \|V^* - V_{\pi_k}\|_{\mu} \leq \frac{2\sqrt{C_1 C_2}\; c(\alpha, \beta, 0, \lambda)}{(1 - c(\alpha, \beta, 0, \lambda))^3}\, \delta.$$

Convergence of RLSPI: Results

Assumptions

(i) $\max_{\pi} \|V_{\pi} - \Pi_{d_{\pi}} V_{\pi}\|_{d_{\pi}} < \delta$ (ii) $d_{\pi} \geq \bar{\mu}/C_2 \geq \mu H(\pi, P^{o})/C_1 C_2$

Theorem (Asymptotic convergence of PI)

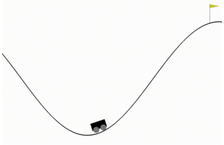
Let V^* be the optimal robust value function. $\{\pi_k\}$ policy sequence of algorithm. Let V_{π_k} be the true robust value function for policy π_k. Let $\rho = 0$.

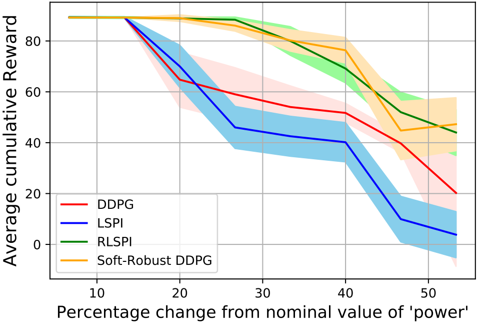
$$\limsup_{k \to \infty} \|V^* - V_{\pi_k}\|_{\mu} \leq \frac{2\sqrt{C_1 C_2}\ c(\alpha, \beta, 0, \lambda)}{(1 - c(\alpha, \beta, 0, \lambda))^3}\ \delta.$$

For some large enough k, V_{π_k} is ϵ-optimal w.r.t V^* under $\|.\|_{\mu}$

RLSPI Simulation Performance

- We train our algorithm on MountainCarContinuous environment in OpenAI Gym with default parameters
- We test for robustness by changing the power parameter

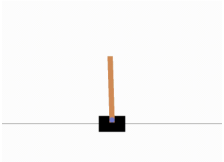


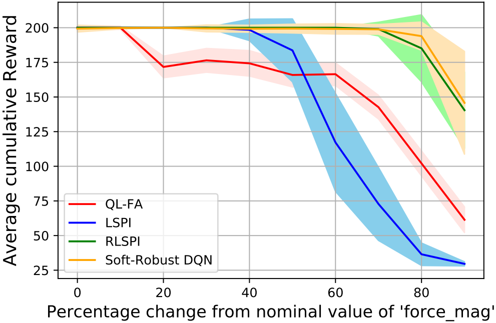

Average cumulative Reward
80
60
40
20
0
10
20
30
40
50
Percentage change from nominal value of 'power'
DDPG
LSPI
RLSPI
Soft-Robust DDPG

Soft-Robust DDPG (Derman et al., 2018)

RLSPI Simulation Performance

- We train our algorithm on CartPole environment in OpenAI Gym with default parameters
- We test for robustness by changing the force-magnitude parameter




Average cumulative Reward
200
175
150
125
100
75
50
25
0
20
40
60
80
Percentage change from nominal value of 'force_mag'
QL-FA
LSPI
RLSPI
Soft-Robust DQN

Soft-Robust DQN (Derman et al., 2018)

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.

Tamar, Aviv, Shie Mannor, and Huan Xu (2014). "Scaling up robust MDPs using function approximation". In: *International Conference on Machine Learning*, pp. 181–189.

Brockman, Greg, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba (2016). "Openai gym". In: *arXiv preprint arXiv:1606.01540*.

Van Seijen, Harm, A Rupam Mahmood, Patrick M Pilarski, Marlos C Machado, and Richard S Sutton (2016). "True online temporal-difference learning". In: *The Journal of Machine Learning Research* 17.1, pp. 5057–5096.

Derman, Esther, Daniel J Mankowitz, Timothy A Mann, and Shie Mannor (2018). "Soft-robust actor-critic policy-gradient". In: *AUAI press for Association for Uncertainty in Artificial Intelligence*, pp. 208–218.

Lim, Shiau Hong and Arnaud Autef (2019). "Kernel-based reinforcement learning in robust markov decision processes". In: *International Conference on Machine Learning*, pp. 3973–3981.

Altahhan, Abdulrahman (2020). "True Online TD (λ)-Replay An Efficient Model-free Planning with Full Replay". In: *2020 International Joint Conference on Neural Networks (IJCNN)*. IEEE, pp. 1–7.

Panaganti, Kishan and Dileep Kalathil (2020). "Model-Free Robust Reinforcement Learning with Linear Function Approximation". In: *arXiv preprint arXiv:2006.11608*.