Q-Learning for L_p Robust Markov Decision Processes

Navdeep Kumar navdeepkumar@campus.technion.ac.il
Electrical and Computer Engineering
Technion
Haifa, Israel

Kaixin Wang kaixin.wang@u.nus.edu
Electrical and Computer Engineering
Technion
Haifa, Israel

Kfir Levy kfirylevy@technion.ac.il
Electrical and Computer Engineering
Technion
Haifa, Israel

Shie Mannor shie@ee.technion.ac.il
Electrical and Computer Engineering
Technion
Haifa, Israel

Abstract

Robust Markov Decision Processes (MDPs) are a powerful tool to solve the sequential decision-making problem where system parameters are partially known or changing or adversarial. Recently, there have been works aimed at solving sa and s-rectangular robust MDPs. The methods are model-based that can potentially be generalized to model-free settings. We formally propose model-free algorithm for sa and s-rectangular L_p robust MDPs and provide its convergence guarantees. The proposed model-free algorithms can be combined with existing deep RL techniques such as DQN etc. to solve challenging problems.

Keywords: Markov Decision Processes, Robustness, Model Free Algorithms, Stochastic Approximation Algorithms

1. Introduction

In Markov Decision Processes (MDPs), an agent interacts with the environment and learns to optimally behave in it Sutton and Barto (2018). Nevertheless, an MDP solution may be very sensitive to the model parameters Mannor et al. (2004); Zhao et al. (2019); Packer et al. (2018), implying that one should be cautious when the model is changing or when there is uncertainty in the model parameters. Robust MDPs (RMDPs) on the other hand, allow some room for the uncertainty in the model parameters as it allows the model parameters to belong to some uncertainty set Hanasusanto and Kuhn (2013); Tamar et al. (2014); Iyengar (2005).

Solving robust MDPs is NP-hard for general uncertainty sets Nilim and Ghaoui (2005), hence the uncertainty set is popularly assumed to be *rectangular* which enables tractability Iyengar (2005); Nilim and Ghaoui (2005); Ho et al. (2020). The uncertainty set is called rectangular if the uncertainty in model parameters (*i.e.*, reward and transition kernel) in one state is not coupled with the uncertainty in model parameters in different state. Under this rectangularity assumption, many structural properties of MDPs remain intactIyengar (2005); Nilim and Ghaoui (2005) and methods such as robust value iteration Bagnell et al. (2001); Wolfram Wiesemann (2012), robust modified policy iteration Kaufman and Schaefer (2013), partial robust policy iteration Ho et al. (2020) etc can be used to solve it. It is also known that uncertainty in the reward can be easily handled, whereas handling uncertainty in transition kernel is much more challenging Derman et al. (2021). When an uncertainty set has a polyhedral structure (as in L_1/L_∞ robust MDPs) then it can be solved using nested Linear programming, that is, both policy evaluation and policy improvement are done using linear programming exactly or inexactlyKaufman and Schaefer (2013); Wolfram Wiesemann (2012). Further in

this direction, partial policy iteration approach to solve L_1-Robust MDPs by Ho et al. (2020) uses bags of tricks such as homotopy, bisection methods etc. But these methods are computationally expensive or limited to certain cases. Derman et al. (2021) developed the methodology to tackle robustness using regularization. It proposed model based robust value iteration for sa-rectangular L_p robust MDPs. And for s-rectangular case, it proposed the policy gradient. Along this line, Kumar et al. (2022), corrects the unrealistic assumption made by Derman et al. (2021) and extends the model based robust value iteration for sa-rectangular L_p robust MDPs. Wang and Zou (2021) provides Q-learning for the R-contamination uncertainty set that can be used in model-free settings, further Wang and Zou (2022) provides policy gradient method for the same. But the R-contamination uncertainty set considered by Wang and Zou (2021, 2022) is sa-rectangular. This paper extends the model-based algorithms in Kumar et al. (2022) to model free setting for both sa and s-rectangular L_p robust MDPs with convergence guarantees.

2. Preliminary

2.1 Notations

For a set $\mathcal{S}$, $|\mathcal{S}|$ denotes its cardinality. $\langle u, v\rangle := \sum_{s\in\mathcal{S}} u(s)v(s)$ denotes the dot product between functions $u, v : \mathcal{S} \to \mathbb{R}$. $\|v\|_p^q := (\sum_s |v(s)|^p)^{\frac{q}{p}}$ denotes the qth power of L_p norm of function v, and we use $\|v\|_p := \|v\|_p^1$ and $\|v\| := \|v\|_2$ as shorthand. For a set $\mathcal{C}$, $\Delta_{\mathcal{C}} := \{a : \mathcal{C} \to \mathbb{R} | a(s) \geq 0, \forall s \sum_{c\in\mathcal{C}} a_c = 1\}$ is the probability simplex over $\mathcal{C}$ $\mathbf{0}, \mathbf{1}$ denotes all zero vector and all ones vector/function respectively of appropriate dimension/domain. $\mathbb{1}(a = b) := 1$ if $a = b$, 0 otherwise, is indicator function.

2.2 Markov Decision Processes

A Markov Decision Process (MDP) is defined by $(\mathcal{S}, \mathcal{A}, P, R, \gamma, \mu)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta_{\mathcal{S}}$ is the transition kernel, $R : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is the reward function, $\mu \in \Delta_{\mathcal{S}}$ is the initial distribution over states and $\gamma \in [0, 1)$ is the discount factor Sutton and Barto (2018). A stationary policy $\pi : \mathcal{S} \to \Delta_{\mathcal{A}}$ maps states to probability distribution over actions. And $\pi(a|s), P(s'|s, a), R(s, a)$ denotes the probability of selecting action a in state s, transition probability to state s' in state s under action a, and reward in state s under action a respectively. We denote $S = |\mathcal{S}|$, and $A = |\mathcal{A}|$ as a shorthand. The objective in a MDP is to maximize the expected discounted cumulative reward, defined as

$$\mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n R(s_n, a_n) \;\middle|\; s_0 \sim \mu, a_n \sim \pi(\cdot|s_n), s_{n+1} \sim P(\cdot|s_n, a_n)\right] = \langle R, d_P^\pi\rangle = \langle \mu, v_{P,R}^\pi\rangle, \tag{1}$$

where d_P^π is occupation measure of policy π with initial distribution of sates μ, and kernel P, defined as

$$d_P^\pi(s, a) := \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n \mathbb{1}(s_n = s, a_n = a) \;\middle|\; s_0 \sim \mu, a_n \sim \pi(\cdot|s_n), s_{n+1} \sim P(\cdot|s_n, a_n)\right],$$

and $v_{P,R}^\pi$ is the value function (dual formulationPuterman (1994)) under policy π with transition kernel P and reward vector R, defined as

$$v_{P,R}^\pi(s) := \mathbb{E}\left[\sum_{n=0}^{\infty} \gamma^n R(s_n, a_n) \;\middle|\; s_0 = s, a_n \sim \pi(\cdot|s_n), s_{n+1} \sim P(\cdot|s_n, a_n)\right]. \tag{2}$$

The value function $v_{P,R}^\pi$ for policy π, is the fixed point of the Bellmen operator $\mathcal{T}_{P,R}^\pi$ Sutton and Barto (2018), defined as

$$(\mathcal{T}_{P,R}^\pi v)(s) = \sum_a \pi(a|s) \left[R(s, a) + \gamma \sum_{s'} P(s'|s, a)v(s') \right]. \tag{3}$$

Similarly, the optimal value function $v_{P,R}^* := \max_\pi v_{P,R}^\pi$ is well defined and is the fixed point of the optimal Bellman operator $\mathcal{T}_{P,R}^*$ Sutton and Barto (2018), defined as

$$(\mathcal{T}_{P,R}^* v)(s) = \max_{a\in\mathcal{A}} \left[R(s, a) + \gamma \sum_{s'} P(s'|s, a)v(s') \right]. \tag{4}$$

The optimal Bellman operator $\mathcal{T}^*_{P,R}$ and Bellman operators $\mathcal{T}^\pi_{P,R}$ for all policy π, are γ-contraction mapsSutton and Barto (2018).

2.3 Robust MDPs

In most practical cases, the system dynamics (transition kernel P and reward function R) are not known precisely. Instead, we have an access to a nominal reward function R_0 and a nominal transition kernel P_0 that may have some uncertainty. To capture this, the uncertainty (or ambiguity) set is defined as $\mathcal{U} := (R_0 + \mathcal{R}) \times (P_0 + \mathcal{P})$, where $\mathcal{R}, \mathcal{P}$ are the respective uncertainties in the reward function and transition kernel Iyengar (2005).The robust performance of a policy π is defined to be its worst performance on the entire uncertainty set $\mathcal{U}$. And our objective is to maximize the robust performance, that is

$$\max_\pi \min_{R,P\in\mathcal{U}} \langle R, d^\pi_P\rangle, \qquad \text{equivalently} \qquad \max_\pi \min_{R,P\in\mathcal{U}} \langle \mu, v^\pi_{P,R}\rangle. \tag{5}$$

Solving the above robust objective is NP-hard in generalIyengar (2005); Wolfram Wiesemann (2012). Hence, the uncertainty set $\mathcal{U}$ is commonly assumed to be $\mathtt{s}$-rectangular, that is $\mathcal{R}$ and $\mathcal{P}$ can be decomposed state wise as $\mathcal{R} = \times_{s\in\mathcal{S}}\mathcal{R}_s$ and $\mathcal{P} = \times_{s\in\mathcal{S}}\mathcal{P}_s$. Sometimes, the uncertainty set $\mathcal{U}$ is assumed to decompose state-action wise as $\mathcal{R} = \times_{s\in\mathcal{S},a\in\mathcal{A}}\mathcal{R}_{s,a}$ and $\mathcal{P} = \times_{s\in\mathcal{S},a\in\mathcal{A}}\mathcal{P}_{s,a}$, known as $(\mathtt{sa})$-rectangular uncertainty set. Observe that it is a special case of $\mathtt{s}$-rectangular uncertainty set. Under the $\mathtt{s}$-rectangularity assumption, many structural properties of MDPs stay intact, and the problem becomes tractable Wolfram Wiesemann (2012); Iyengar (2005); Nilim and Ghaoui (2005).Throughout the paper, we assume that the uncertainty set $\mathcal{U}$ is $\mathtt{s}$-rectangular (or $(\mathtt{sa})$-rectangular) unless stated otherwise. Under $\mathtt{s}$-rectangularity, the robust value function is well defined Nilim and Ghaoui (2005); Wolfram Wiesemann (2012); Iyengar (2005) as

$$v^\pi_\mathcal{U} := \min_{R,P\in\mathcal{U}} v^\pi_{P,R}. \tag{6}$$

Using the robust value function, robust policy performance can be rewritten as

$$\min_{R,P\in\mathcal{U}} \langle \mu, v^\pi_{P,R}\rangle = \langle \mu, \min_{R,P\in\mathcal{U}} v^\pi_{P,R}\rangle = \langle \mu, v^\pi_\mathcal{U}\rangle. \tag{7}$$

The robust value function $v^\pi_\mathcal{U}$ is the fixed point of the robust Bellmen operator $\mathcal{T}^\pi_\mathcal{U}$ Wolfram Wiesemann (2012); Iyengar (2005), defined as

$$(\mathcal{T}^\pi_\mathcal{U} v)(s) = \min_{R,P\in\mathcal{U}} \sum_a \pi(a|s) \left[R(s,a) + \gamma \sum_{s'} P(s'|s,a) v(s') \right]. \tag{8}$$

Moreover, the optimal robust value function $v^*_\mathcal{U} := \max_\pi v^\pi_\mathcal{U}$ is well defined and is the fixed point of the optimal robust Bellman operator $\mathcal{T}^*_\mathcal{U}$ Iyengar (2005); Wolfram Wiesemann (2012), defined as

$$(\mathcal{T}^*_\mathcal{U} v)(s) = \max_{\pi_s\in\Delta_\mathcal{A}} \min_{R,P\in\mathcal{U}} \sum_a \pi_s(a) \left[R(s,a) + \gamma \sum_{s'} P(s'|s,a) v(s') \right]. \tag{9}$$

The optimal robust Bellman operator $\mathcal{T}^*_\mathcal{U}$ and robust Bellman operators $\mathcal{T}^\pi_\mathcal{U}$ are γ contraction maps for all policy π (see theorem 3.2 of Iyengar (2005)), that is

$$\|\mathcal{T}^*_\mathcal{U} v - \mathcal{T}^*_\mathcal{U} u\|_\infty \le \gamma \|u - v\|_\infty, \qquad \|\mathcal{T}^\pi_\mathcal{U} v - \mathcal{T}^\pi_\mathcal{U} u\|_\infty \le \gamma \|u - v\|_\infty, \qquad \forall \pi, u, v. \tag{10}$$

So for all initial values v^π_0, v^*_0, sequences defined as

$$v^\pi_{n+1} := \mathcal{T}^\pi_\mathcal{U} v^\pi_n, \qquad v^*_{n+1} := \mathcal{T}^*_\mathcal{U} v^*_n \tag{11}$$

converges linearly to their respective fixed points, that is $v^\pi_n \to v^\pi_\mathcal{U}$ and $v^*_n \to v^*_\mathcal{U}$. This makes the robust value iteration an attractive method for solving robust MDPs.

2.4 L_p Robust MDPs

In this section, we basically summarize the useful results from Kumar et al. (2022). We begin by making a few useful definitions as per Kumar et al. (2022). Let q be such that it satisfies the Holder's equality, i.e. $\frac{1}{p} + \frac{1}{q} = 1$. Let p-variance function $\kappa_p : \mathcal{S} \to \mathbb{R}$ and p-mean function $\omega_p : \mathcal{S} \to \mathbb{R}$ be defined as

$$\kappa_p(v) := \min_{\omega\in\mathbb{R}} \|v - \omega\mathbf{1}\|_p, \qquad \omega_p := arg\min_{\omega\in\mathbb{R}} \|v - \omega\mathbf{1}\|_p. \tag{12}$$

Table 1: p-variance

x	$\kappa_x(v)$	remark
p	$\|v - \omega_p\|_p$	ω_p can be found by binary search
∞	$\frac{1}{2}\left(\max_s v(s) - \min_s v(s)\right)$	Peak to peak difference
2	$\sqrt{\sum_s \left(v(s) - \frac{\sum_s v(s)}{S}\right)^2}$	Variance
1	$\sum_{i=1}^{\lfloor (S+1)/2 \rfloor} v(s_i) - \sum_{i=\lceil (S+1)/2 \rceil}^{S} v(s_i))$	Top half minus bottom half

where v is sorted, i.e. $v(s_i) \geq v(s_{i+1}) \quad \forall i$.

$\omega_p(v)$ can be calculated by binary search in the range $[\min_s v(s), \max_s v(s)]$ and can then be used to approximate $\kappa_p(v)$ Kumar et al. (2022). Observe that for $p = 1, 2, \infty$, the p-variance function κ_p can also be computed in closed form, see table 1 for summary Kumar et al. (2022).

2.4.1 SA-RECTANGULAR

In accordance with Derman et al. (2021); Kumar et al. (2022), we define $(\texttt{sa})$-rectangular L_p constrained uncertainty set as

$$\mathcal{U}_p^{\texttt{sa}} := (R_0 + \mathcal{R}) \times (P_0 + \mathcal{P}) \quad \text{where}$$
$$\mathcal{R} = \times_{s \in \mathcal{S}, a \in \mathcal{A}} \mathcal{R}_{s,a}, \qquad \mathcal{R}_{s,a} = \{r_{s,a} \in \mathbb{R} \mid \|r_{s,a}\|_p \leq \alpha_{s,a}\}$$
$$\mathcal{P} = \times_{s \in \mathcal{S}, a \in \mathcal{A}} \mathcal{P}_{s,a} \qquad \mathcal{P}_{s,a} = \{p_{s,a} : \mathcal{S} \to \mathbb{R} \mid \sum_{s'} p_{s,a}(s') = 0, \|p_{s,a}\|_p \leq \beta_{s,a}\},$$

and $\alpha_{s,a}, \beta_{s,a} \in \mathbb{R}$ are reward noise radius and transition kernel noise radius respectively. These are chosen small enough so that all the transition kernels in $(P_0 + \mathcal{P})$ are well defined. Let $Q^*_{\mathcal{U}_p^{\texttt{sa}}}$ denote the optimal robust Q-values associated with optimal robust value $v^*_{\mathcal{U}_p^{\texttt{sa}}}$, given as

$$Q^*_{\mathcal{U}_p^{\texttt{sa}}}(s,a) := -\alpha_{s,a} - \gamma \beta_{s,a} \kappa_q(v^*_{\mathcal{U}_p^{\texttt{sa}}}) + R_0(s,a) + \gamma \sum_{s'} P_0(s'|s,a) v^*_{\mathcal{U}_p^{\texttt{sa}}}(s'). \tag{13}$$

It is evident from theorem 1 that optimal robust value and optimal robust Q-values satisfies the following relation, same as its non-robust counterparts,

$$v^*_{\mathcal{U}_p^{\texttt{sa}}}(s') = \max_{a \in \mathcal{A}} Q^*_{\mathcal{U}_p^{\texttt{sa}}}(s,a). \tag{14}$$

Proposition 1 *(Theorem 1, Kumar et al. (2022))* $(\texttt{Sa})$*-rectangular L_p robust Bellman operator is equivalent to reward regularized (non-robust) Bellman operator. That is, using κ_p defined in (12), we have*

$$(\mathcal{T}^{\pi}_{\mathcal{U}_p^{\texttt{sa}}} v)(s) = \sum_a \pi(a|s)[-\alpha_{s,a} - \gamma \beta_{s,a} \kappa_q(v) + R_0(s,a) + \gamma \sum_{s'} P_0(s'|s,a) v(s')], \qquad \textit{and}$$
$$(\mathcal{T}^{*}_{\mathcal{U}_p^{\texttt{sa}}} v)(s) = \max_{a \in \mathcal{A}} [-\alpha_{s,a} - \gamma \beta_{s,a} \kappa_q(v) + R_0(s,a) + \gamma \sum_{s'} P_0(s'|s,a) v(s')].$$

Corollary 2 *The Q-value iteration,*

$$Q_{n+1}(s,a) = R_0(s,a) - \alpha_{sa} - \gamma \beta_{sa} \kappa_q(v_n) + \gamma \sum_{s'} P_0(s'|s,a) \max_{a \in \mathcal{A}} Q_n(s',a),$$

where $v_n(s) = \max_{a \in \mathcal{A}} Q_n(s,a)$, then Q_n converges to $Q^*_{\mathcal{U}_p^{\texttt{sa}}}$ linearly for any initial Q-value Q_0.

Proof The proof is in appendix B ■

2.4.2 S-RECTANGULAR

In accordance with Derman et al. (2021), We define s-rectangular L_p constrained uncertainty set as

$$\mathcal{U}_p^{\mathtt{s}} = (R_0 + \mathcal{R}) \times (P_0 + \mathcal{P}), \quad \text{where} \quad \mathcal{R} = \times_{s\in\mathcal{S}}\mathcal{R}_s, \quad \mathcal{R}_s = \{r_s : \mathcal{A} \to \mathbb{R} \;\Big|\; \|r_s\|_p \le \alpha_s\},$$

$$\mathcal{P} = \times_{s\in\mathcal{S}}\mathcal{P}_s, \qquad \mathcal{P}_s = \{p_s : \mathcal{S}\times\mathcal{A} \to \mathbb{R} \;\Big|\; \sum_{s'} p_s(s', a) = 0, \forall a, \|p_s\|_p \le \beta_s\},$$

and $\alpha_s, \beta_s \in \mathbb{R}$ are reward and transition kernel respectively noise radius that are chosen small enough so that all the transition kernels in $P_0 + \mathcal{P}$ are well defined. Let $Q^*_{\mathcal{U}_p^{\mathtt{s}}}$ denote the optimal robust Q-values associated with optimal robust value $v^*_{\mathcal{U}_p^{\mathtt{s}}}$, defined as

$$Q^*_{\mathcal{U}_p^{\mathtt{s}}}(s, a) := R_0(s, a) + \gamma \sum_{s'} P_0(s'|s, a) v^*_{\mathcal{U}_p^{\mathtt{s}}}(s'). \tag{15}$$

Let us define the function $\psi_p, \psi_p^\infty : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \times \mathbb{R}^{\mathcal{S}} \to \mathbb{R}^{\mathcal{S}}$ as following: $\psi_p(Q, v)(s)$ is the solution to

$$\big(\ \alpha_s + \gamma\beta_s\kappa_q(v)\ \big)^p = \sum_a \big(\ Q(s, a) - x\ \big)^p\ \mathbf{1}\ \big(\ Q(s, a) \ge x\ \big), \tag{16}$$

and $\psi_p^\infty(Q, v)(s)$ is the solution to

$$\big(\ \gamma\beta_s\kappa_q(v)\ \big)^p = \sum_a \big(\ Q(s, a) - x\ \big)^p\ \mathbf{1}\ \big(\ Q(s, a) \ge x\ \big)\ . \tag{17}$$

Proposition 3 *The Q-value iteration,*

$$Q_{n+1}(s, a) := R_0(s, a) + \gamma \sum_{s'} P_0(s'|s, a)\psi_p(Q_n, v_n)(s'), \tag{18}$$

$$v_{n+1} := \psi_p(Q_n, v_n). \tag{19}$$

Then Q_n, v_n converges to their optimal values, i.e. $\lim_{n\to\infty} Q_n = Q^*_{\mathcal{U}_p^{\mathtt{s}}}, \lim_{n\to\infty} v_n = v^*_{\mathcal{U}_p^{\mathtt{s}}}$ linearly, for any initial value function v_0 and $Q_0(s, a) := R_0(s, a) + \gamma P_0(s'|s, a)v_0(s')$.

Proof The above result follows from theorem 3 of Kumar et al. (2022). It is discussed in more details in section B. ■

3. Stochastic Approximation Algorithms

In this section, we outline some useful stochastic approximation algorithms results from Borkar (2008); Borkar and Soumyanatha (1997); Borkar and Meyn (2000). The stochastic approximation algorithm takes the following form,

$$X(n + 1) = X(n) + a(n)\ \big[\ h(X_n) + M(n + 1)\ \big], \quad n \ge 0, \tag{20}$$

where $X(n) \in \mathbb{R}^d$, $h : \mathbb{R}^d \to \mathbb{R}^d$, and $\{a(n)\}$ is a sequence of positive numbers. The sequence $\{M(n) : n \ge 0\}$ is noise sequence. Below are the assumptions under which the above sequence can be modeled as ordinary differential equation (ODE).

Assumption 1 *(Borkar and Meyn (2000))*

1. *The ODE*
$$\dot{x}(t) = h(x(t)), \qquad t \geq 0, \tag{21}$$
has unique a unique asymptotically stable equilibrium x^*.

2. *The function h is Lipschitz. Let the limiting ODE be*
$$h_\infty(x) := \lim_{r\to\infty} \frac{h(rx)}{r}. \tag{22}$$
The origin in $\mathbb{R}^d$ is an aysmptotically stable equilibrium for the ODE
$$\dot{x}(t) = h_\infty(x(t)), \qquad t \geq 0. \tag{23}$$

3. *The sequence $\{M(n), n \geq 0\}$, with $\mathcal{F}_n = \sigma(X(i), M(i), i \leq n)$ is a martingale difference noise. Moreover, for some constant $C_0 \leq \infty$ and any initial condition $X(0) \in \mathbb{R}^d$*
$$\mathbf{E}\left[\|M(n+1)\|^2 \,\middle|\, \mathcal{F}_n \right] \leq C_0(1 + \|X(n)\|^2), \qquad n \geq 0. \tag{24}$$

4. *The sequence $\{a(n)\}$ satisfies $0 < a(n) \leq 1, \quad n \geq 0$, and*
$$\sum_n a(n) = \infty, \qquad \sum_n a(n)^2 < \infty. \tag{25}$$

Proposition 4 *(Theorem 2.2, Borkar and Meyn (2000)) Under assumption 1,*
$$\lim_{n\to\infty} X(n) = x^*, \qquad \text{a.s.}$$
for all initial condition $X(0) \in \mathbb{R}^d$.

3.1 Asynchronous Update

Now we focus on the asynchronous version of the above stochastic approximation algorithm. The requirement of updating all components of $X(n)$ at each time step is relaxed. Let $\{Y(n) \subset \{1, 2, \cdots, d\}, n \geq 0\}$ be the sequence of components to be updated, in other words $Y(n)$ is the set of component to be updated at time n. Further, let us define the counter function as
$$\nu(i, n) := \sum_{k=0}^{n} \mathbf{1}(i \in Y(n)). \tag{26}$$
Observe that $\nu(i, n)$ is the number of times the component i is updated till time n. Now, state the additional assumption required for convergence of asynchronous version of stochastic approximation algorithm in hand.

Assumption 2

1. *All components are updated comparably often. That is, there exists some constant $\Delta > 0$, such that*
$$\liminf_{n\to\infty} \frac{\nu(i, n)}{n} \geq \Delta, \qquad i.$$

2. *The following limit exist a.s. for all $x > 0, i, j$,*
$$\lim_{n\to\infty} \frac{\sum_{m=n}^{N(n,x)} a(\nu(i, m))}{\sum_{m=n}^{N(n,x)} a(\nu(j, m))},$$
where
$$N(n, x) := \min\{m \geq n \mid \sum_{k=n+1}^{m} a(m) > x\}.$$

Proposition 5 *(Theorem 2.5, Borkar and Meyn (2000)) Under assumption 1 and 2, the asynchronous stochastic approximation algorithm,*

$$X(n+1)(i) = X(n) + a(\nu(i,n))\mathbf{1}(i \in Y(n)) \left[h(X_n) + M(n+1) \right], \quad n \geq 0, \tag{27}$$

converges to optimal value x^*, similar to synchronous case, that is

$$\lim_{n\to\infty} X(n) = x^*.$$

3.2 Two Time Scale

We consider two time scale algorithm as follows, for all $n \geq 0$

$$X(n+1) = X(n) + a(n) \left[h^1(X_n, Y_n) + M^1(n+1) \right], \tag{28}$$

$$Y(n+1) = Y(n) + b(n) \left[h^2(X_n, Y_n) + M^2(n+1) \right], \tag{29}$$

where $X(n) \in \mathbb{R}^{d_1}, Y(n) \in \mathbb{R}^{d_2}$, $h^1 : \mathbb{R}^{d_1+d_2} \to \mathbb{R}^{d_1}, h^2 : \mathbb{R}^{d_1+d_2} \to \mathbb{R}^{d_2}$, and $\{a(n)\}, \{b(n)\}$ are sequences of positive numbers. Let us state the assumption under which the above sequence can be modeled as ordinary differential equation (ODE).

Assumption 3 *1. Step size sequences*

$$\sum_n a(n) = \sum_n b(n) = \infty, \qquad \sum_n \left(a(n)^2 + b(n)^2 \right) < \infty, \qquad \frac{b(n)}{a(n)} \to 0.$$

2. The sequence $\{M^1(n) : n \geq 0\}, \{M^2(n) : n \geq 0\}$ are uncorrelated with zero mean. And

$$\mathbf{E}\left[\|M^i(n+1)\|^2 \mid \mathcal{F}_n \right] \leq K(1 + \|X(n)\|^2 + \|Y(n)\|^2), i = 1, 2,$$

where $\mathcal{F}_n := \sigma(X(m), Y(m), M^1_m, M^2_m, m \leq n)$ for $n \geq 0$.

3. The ODE

$$\dot{x}(t) = h^1(x(t), y)$$

has a global asymptotically stable equilibrium $\lambda(y)$ (uniformy in y), where λ is a Lipschitz map.

4. The ODE

$$\dot{y}(t) = h^2(\lambda(y(t)), y(t))$$

has a globally asymptotically stable equilibrium y^*.

5. The function

$$h^1_r(x, y) := \frac{h^1(rx, ry)}{r}, \qquad r \geq 1,$$

satisfy $h^1_r \to h^1_\infty$ as $r \to \infty$, uniformly on compacts for some h^1_∞. Also the limiting ODE $\dot{x}(t) = h^1_\infty(x(t), y)$ has unique globally asymptotically stable equilibrium $\lambda_\infty(y)$, where λ_∞ is a Lipschitz map. Further $\lambda_\infty(0) = 0$, i.e., the ODE $\dot{x}(t) = h^1_\infty(x(t), 0)$ has the origin as its unique globally asymptotically stable equilibrium.

6. The function

$$h^2_r(y) := \frac{h^2(r\lambda_\infty(y), ry)}{r}, \qquad r \geq 1,$$

satisfy $h^2_r \to h^2_\infty$ as $r \to \infty$, uniformly on compacts for some h^2_∞. Also the limiting ODE $\dot{y}(t) = h^2_\infty(y(t))$ has the origin as its unique globally asymptotically stable equilibrium.

Proposition 6 *(Theorem 2, chapter 6, Borkar (2008)) Under assumption 3, $(X(n), Y(n)) \to (\lambda(y^*), y^*)$ a.s.*

4. Synchronous Q-Learning

4.1 The sa-rectangularity case

In this section, we will prove that the Algorithm 1 converges to optimal Q-values under appropriated step size sequences. We will use techniques from Borkar and Meyn (2000) for the same. We start with the synchronous case first then we will see that the analysis extends to the asynchronous case just as easily as its non-robust counterpart. We have the following stochastic approximation

$$Q_{n+1} = Q_n + \eta_n \left[h(Q) + M_{n+1} \right], \tag{30}$$

where

$$h(Q)(s,a) \coloneqq R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(Q) + \gamma \sum_{s'} P_0(s'|s,a) \max_{a\in\mathcal{A}} Q(s',a) - Q(s,a) \tag{31}$$

and

$$M_{n+1}(s,a) \coloneqq \gamma \max_{a\in\mathcal{A}} Q_n(s',a) - \gamma \sum_{s'} P_0(s'|s,a) \max_{a\in\mathcal{A}} Q_n(s',a), \qquad s' \sim P_0(\cdot \mid s,a). \tag{32}$$

Theorem 7 *The Q-value iteration, defined as*

$$Q_{n+1}(s,a) = Q_n(s,a) + \eta_n \left[R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(Q) + \gamma \max_{a'} Q_n(s',a') - Q_n(s,a) \right], \qquad s' \sim P_0(\cdot|s,a),$$

converges to the optimal Q-value, i.e

$$\lim_{n\to\infty} Q_n = Q^*_{\mathcal{U}_p^{sa}},$$

for any initial Q-value Q_0.

Proof We show in appendix C.1 that the above Q-value iterations satisfies Assumption 1, hence the result follows from theorem 2.2 of (Borkar and Meyn, 2000). ■

Synchronous sample based algorithm based on the above result is presented in Algorithm 1.

Algorithm 1 Synchronous Q-Learning Algorithm for sa-Rectangular L_p Robust MDP

1: Choose appropriate step sizes η_n for $n \geq 0$. Randomly initialize Q_0.
2: **while** not converged **do**
3: Calculate value variance

$$\sigma = \kappa_q(v_n), \quad \text{where } v_n(s) \coloneqq \max_a Q_n(s,a). \tag{33}$$

4: **for** $s, a \in \mathcal{S} \times \mathcal{A}$ **do**
5: Get the next state $s' \sim P_0(\cdot|s,a)$ from the environment.
6: Update Q-value as

$$Q_{n+1}(s,a) = Q_n(s,a) + \eta_n \left[R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\sigma + \gamma \max_{a'} Q_n(s',a') - Q_n(s,a) \right]. \tag{34}$$

7: **end for**
8: **end while**

4.2 The s-rectangularity case

In this section, we will prove that the algorithm 2 converges to optimal Q-values under appropriate step size sequences. We will use techniques from Borkar (2008)(chapter six) for two time scale, for the same.

Theorem 8 *The Q-value and value iteration*

$$\begin{aligned} Q_{n+1}(s,a) &:= Q_n(s,a) + \eta_n^1 \left[R_0(s,a) + \gamma v_n(s') - Q_n(s,a) \right], \qquad s' \sim P_0(\cdot|s,a), \\ v_{n+1}(s) &:= v_n(s) + \eta_n^2 \left[\psi_p(Q_n, v_n)(s) - v_n(s) \right] \end{aligned} \tag{35}$$

converges to their optimal values. That is

$$Q_n \to Q^*_{\mathcal{U}_p^s}, \qquad \textit{and} \qquad v_n \to v^*_{\mathcal{U}_p^s},$$

for any initial Q-value Q_0 and value v_0.

Proof We show in appendix C.2 that the above Q-value iterations satisfies assumption 3, then the result follows directly from Lakshminarayanan and Bhatnagar (2017) that builds up on theorem 2, chapter 6, Borkar (2008). ■

Synchronous sample based algorithm based on the above result is presented in algorithm 2.

Algorithm 2 Synchronous Algorithm for s-rectangular L_p Robust MDP

Choose appropriate step sizes sequences $\{\eta_n^1\}, \{\eta_n^2\}$. Take initial Q-values Q_0 and value function v_0 randomly.
while not converged **do**
  Update value function:

$$v_{n+1}(s) = v_n(s) + \eta_n^2 \left[\psi_p(Q_n, v_n) - v_n(s)\right]. \tag{36}$$

  for $a \in \mathcal{A}$ **do**
    Get the next state $s' \sim P_0(\cdot|s,a)$ from the environment.
    Update Q-value as

$$Q_{n+1}(s,a) = Q_n(s,a) + \eta_n^1 [R_0(s,a) + \gamma v_n(s') - Q_n(s,a)]. \tag{37}$$

  end for
end while

5. Asynchronous Q-Learning

Now we move our attention to the asynchronous case. Recall from theorem 2.5 of Borkar and Meyn (2000) that asynchronous case requires additional assumption on the relative frequency of updates in different components (states and action). In its non-robust counterpart, the required assumption (assumption 2) is satisfied through appropriate exploratory action strategies (such as ϵ greedy action). Robust counterpart has no addition complications, hence the same exploratory strategies (such as ϵ greedy action) would be sufficient to ensure that the assumption 2 is satisfied, and consequently the asynchronous version of the Q-value iteration for sa-rectangular L_p robust MDPs, converges to their optimal values. And κ_∞, κ_2 can be easily computed asynchronously that paves the path for asynchronous model free algorithm for sa-rectangular L_p robust MDPs, for $p = 1, 2$. For s-rectanglur case, we need to find the conditions that is required to extend synchronous stochastic algorithm to asynchronous stochastic algorithm. We believe, this extention may be very similar to single time scale case, and consequently everything above for sa-rectangular case, hold for s-rectangular case too. Asynchronous algorithms are presented in appendix D for sa/s-rectangular L_1/L_2 robust MDPs.

6. Discussion and Future Works

The work provide covergence guarantees for the model free algorithms for L_p robust MDPs. It would be interesting to study the rate of convergence. We plan to apply these algorithms with deep learning to solve real world problems. It would be interesting to find a way too estimate κ_p asynchronously, that will pave the path for asynchronous algorithms for sa/s-rectangular L_p robust MDPs.

Acknowledgments

Acknowledgements

References

J. Andrew Bagnell, Andrew Y. Ng, and Jeff G. Schneider. Solving uncertain markov decision processes. Technical report, Carnegie Mellon University, 2001.

V. S. Borkar and S. P. Meyn. The o.d.e. method for convergence of stochastic approximation and reinforcement learning. *SIAM Journal on Control and Optimization*, 38(2):447–469, 2000. doi: 10.1137/S0363012997331639. URL https://doi.org/10.1137/S0363012997331639.

Vivek Borkar. *Stochastic Approximation: A Dynamical Systems Viewpoint*. 01 2008. ISBN 978-81-85931-85-2. doi: 10.1007/978-93-86279-38-5.

V.S. Borkar and K. Soumyanatha. An analog scheme for fixed point computation. i. theory. *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, 44(4):351–355, 1997. doi: 10.1109/81.563625.

Esther Derman, Matthieu Geist, and Shie Mannor. Twice regularized mdps and the equivalence between robustness and regularization, 2021.

Grani Adiwena Hanasusanto and Daniel Kuhn. Robust data-driven dynamic programming. In C.J. Burges, L. Bottou, M. Welling, Z. Ghahramani, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems*, volume 26. Curran Associates, Inc., 2013. URL https://proceedings.neurips.cc/paper/2013/file/ef575e8837d065a1683c022d2077d342-Paper.pdf.

Chin Pang Ho, Marek Petrik, and Wolfram Wiesemann. Partial policy iteration for l1-robust markov decision processes, 2020. URL https://arxiv.org/abs/2006.09484.

Garud N. Iyengar. Robust dynamic programming. *Mathematics of Operations Research*, 30(2):257–280, May 2005. ISSN 1526-5471. doi: 10.1287/moor.1040.0129. URL http://dx.doi.org/10.1287/MOOR.1040.0129.

David L. Kaufman and Andrew J. Schaefer. Robust modified policy iteration. *INFORMS J. Comput.*, 25:396–410, 2013.

Navdeep Kumar, Kfir Levy, Kaixin Wang, and Shie Mannor. Efficient policy iteration for robust markov decision processes via regularization, 2022. URL https://arxiv.org/abs/2205.14327.

Chandrashekar Lakshminarayanan and Shalabh Bhatnagar. A stability criterion for two timescale stochastic approximation schemes. 79(C), 2017. ISSN 0005-1098. doi: 10.1016/j.automatica.2016.12.014. URL https://doi.org/10.1016/j.automatica.2016.12.014.

Shie Mannor, Duncan Simester, Peng Sun, and John N. Tsitsiklis. Bias and variance in value function estimation. In *Proceedings of the Twenty-First International Conference on Machine Learning*, ICML '04, page 72, New York, NY, USA, 2004. Association for Computing Machinery. ISBN 1581138385. doi: 10.1145/1015330.1015402. URL https://doi.org/10.1145/1015330.1015402.

Arnab Nilim and Laurent El Ghaoui. Robust control of markov decision processes with uncertain transition matrices. *Oper. Res.*, 53:780–798, 2005.

Charles Packer, Katelyn Gao, Jernej Kos, Philipp Krähenbühl, Vladlen Koltun, and Dawn Song. Assessing generalization in deep reinforcement learning, 2018. URL https://arxiv.org/abs/1810.12282.

Martin L. Puterman. Markov decision processes: Discrete stochastic dynamic programming. In *Wiley Series in Probability and Statistics*, 1994.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

Aviv Tamar, Shie Mannor, and Huan Xu. Scaling up robust mdps using function approximation. In Eric P. Xing and Tony Jebara, editors, *Proceedings of the 31st International Conference on Machine Learning*, volume 32 of *Proceedings of Machine Learning Research*, pages 181–189, Bejing, China, 22–24 Jun 2014. PMLR. URL https://proceedings.mlr.press/v32/tamar14.html.

Yue Wang and Shaofeng Zou. Online robust reinforcement learning with model uncertainty, 2021. URL https://arxiv.org/abs/2109.14523.

Yue Wang and Shaofeng Zou. Policy gradient method for robust reinforcement learning, 2022.

Berç Rustem Wolfram Wiesemann, Daniel Kuhn. Robust markov decision processes. *Mathematics of Operations Research 38(1):153-183*, 2012.

Chenyang Zhao, Olivier Sigaud, Freek Stulp, and Timothy M. Hospedales. Investigating generalisation in continuous deep reinforcement learning, 2019. URL https://arxiv.org/abs/1902.07015.

Appendix A. Helper Results

Proposition 9 *Let the function $f : \mathbb{R}^A \to \mathbb{R}$ be defined as*

$$f(b, v) = x \qquad \textit{s.t.} \qquad (\alpha + \gamma\beta\kappa_q(v))^p = \sum_{i=1}^{A}(b_i - x)^p \mathbf{1}(b_i \geq x),$$

then the function is Lipschitz in b and v.

Proof Follows from proposition 10 and 11. ■

Proposition 10 *Let the function $f : \mathbb{R}^A \to \mathbb{R}$ be defined as*

$$f(b) = x \qquad \textit{s.t.} \qquad \alpha^p = \sum_{i=1}^{A}(b_i - x)^p \mathbf{1}(b_i \geq x),$$

then the function is Lipschitz in b.

Proof Observe the following inequality, that we will use next

$$(y - \epsilon_1)^p \mathbf{1}(y \geq \epsilon_1) \leq y^p \mathbf{1}(y \geq 0) \leq (y + \epsilon_2)^p \mathbf{1}(y \geq -\epsilon_2), \qquad \forall p, \epsilon_1, \epsilon_2 \geq 0 \quad \forall y.$$

Let b' be a perturbation of b such that $\|b' - b\| \leq \epsilon$. Now using the above inequality, we have

$$(b_i - (x + \epsilon_1))^p \mathbf{1}(b_i \geq x + \epsilon_1) \leq (b_i - x)^p \mathbf{1}(b_i \geq x) \leq (b_i - (x - \epsilon_2))^p \mathbf{1}(b'_i \geq x - \epsilon_2),$$

where $\epsilon_1 = \epsilon + b_i - b'_i, \epsilon_2 = \epsilon + b'_i - b_i$. Note that $\epsilon_1, \epsilon_2 \geq 0$. Putting back the values of ϵ_1, ϵ_2 above, we get

$$(b'_i - (x + \epsilon))^p \mathbf{1}(b'_i \geq x + \epsilon) \leq (b_i - x)^p \mathbf{1}(b_i \geq x) \leq (b'_i - (x - \epsilon))^p \mathbf{1}(b'_i \geq x - \epsilon).$$

Summing over i, we get

$$\sum_i^A (b'_i - (x + \epsilon))^p \mathbf{1}(b'_i \geq x + \epsilon) \leq \sum_i^A (b_i - x)^p \mathbf{1}(b_i \geq x) \leq \sum_i^A (b'_i - (x - \epsilon))^p \mathbf{1}(b'_i \geq x - \epsilon).$$

Putting $x = f(b)$, we have

$$\sum_i^A (b'_i - (f(b) + \epsilon))^p \mathbf{1}(b'_i \geq f(b) + \epsilon) \leq \sum_i^A (b_i - f(b))^p \mathbf{1}(b_i \geq f(b)) \leq \sum_i^A (b'_i - (f(b) - \epsilon))^p \mathbf{1}(b'_i \geq f(b) - \epsilon).$$

$$\implies \sum_i^A (b'_i - (f(b) + \epsilon))^p \mathbf{1}(b'_i \geq f(b) + \epsilon) \leq \alpha^p \leq \sum_i^A (b'_i - (f(b) - \epsilon))^p \mathbf{1}(b'_i \geq f(b) - \epsilon).$$

We know that the function $g(x) := \sum_{i=1}^A (b'_i - x)^p \mathbf{1}(b'_i \geq x)$ is monotonically decreasing in x. From the above relation, we have

$$g(f(b) + \epsilon) \leq \alpha^p \leq g(f(b) - \epsilon),$$

and by definition $g(f(b')) = \alpha^p$. From continuity of g, we conclude

$$f(b) - \epsilon \leq f(b') \leq f(b) + \epsilon.$$

So conclude, that

$$|f(b) - f(b')| \leq \|b - b'\|,$$

hence Lipschitz of function f is established. ■

Proposition 11 *Let the function $f : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}$ be defined as*

$$f(v) = x \qquad s.t. \qquad (\alpha + \gamma\beta\kappa_q(v))^p = \sum_{i=1}^{A}(b_i - x)^p \mathbf{1}(b_i \geq x),$$

then the function is Lipschitz in v.

Proof Let the function $g : [\min_i b_i, \max_i b_i] \to \mathbb{R}$ be defined as

$$g(x) := \Big(\sum_{i=1}^{A}(b_i - x)^p \mathbf{1}(b_i \geq x) \Big)^{\frac{1}{p}} = \Big(\sum_{b_i \geq x}(b_i - x)^p \Big)^{\frac{1}{p}} .$$

Its derivative is given by

$$\frac{dg(x)}{dx} = -\Big(\sum_{b_i \geq x} |b_i - x|^p\Big)^{\frac{1}{p}-1} \sum_{b_i \geq x} |b_i - x|^{p-1} \tag{38}$$

$$= -\frac{\sum_{b_i \geq x} |b_i - x|^{p-1}}{\big(\sum_{b_i \geq x} |b_i - x|^p\big)^{\frac{p-1}{p}}} \tag{39}$$

$$= -\Big[\frac{\big(\sum_{b_i \geq x} |b_i - x|^{p-1}\big)^{\frac{1}{p-1}}}{\big(\sum_{b_i \geq x} |b_i - x|^p\big)^{\frac{1}{p}}}\Big]^{p-1} \tag{40}$$

$$\leq -1. \tag{41}$$

The inequality follows from the following relation between L_p norm,

$$\|x\|_a \geq \|x\|_b, \qquad \forall 0 \leq a \leq b.$$

It is easy to see that the function g is strictly monotone in the range $[\min_i b_i, \max_i b_i]$, so its inverse is well defined in the same range.Then derivative of the inverse of the function g is bounded as

$$0 \geq \frac{d}{dx} g^-(x) \geq -1.$$

Observe that

$$g^-(\alpha + \gamma\beta\kappa_q(v)) = f(v)$$

and from proposition 12, we know the q-variance function $\kappa_p(v)$ is Lipschitz in v. That implies that function f is Lipschitz. ■

Proposition 12 *The p-variance function $\kappa_p(v)$ is Lipschitz in v.*

Proof Let us first prove the mean function

$$\omega_p(v) = arg\min_{w \in \mathbb{R}} \|v - \omega\|_p,$$

is Lipschitz. Note that the norm $\|\cdot\|$ is convex, hence any local minima is the global minima. Taking its derivative,

$$\frac{\partial}{\partial\omega}\|v - \omega\|_p = \frac{\partial}{\partial\omega}\Big(\sum_s |v(s) - \omega|^p\Big)^{\frac{1}{p}} \tag{42}$$

$$= \Big(\sum_s |v(s) - \omega|^p\Big)^{\frac{1}{p}-1} \sum_s sign(v(s) - \omega)|v(s) - \omega|^{p-1} \tag{43}$$

$$\tag{44}$$

The derivative must be zero at $\omega_p(v)$, so we have

$$\big(\sum_s |v(s)-\omega_p(v)|^p\big)^{p-1}\sum_s sign(v(s)-\omega_p(v))|v(s)-\omega_p(v)|^{p-1}=0 \tag{45}$$

$$\implies \sum_s sign(v(s)-\omega_p(v))|v(s)-\omega_p(v)|^{p-1}=0. \tag{46}$$

$$\tag{47}$$

Let u be an ϵ perturbation of v, that is $\|v-u\|\le\epsilon$. And let the function g be defined as

$$g(x):=\sum_s sign(u(s)-x)|u(s)-x|^{p-1}.$$

Now we will prove $g(\omega_p(v)-\epsilon)\ge 0\ge g(\omega_p(v)+\epsilon)$. Observe that

$$sign(u(s)-\omega_p(v)-\epsilon)|u(s)-\omega_p(v)-\epsilon|^{p-1}\le sign(v(s)-\omega_p(v))|v(s)-\omega_p(v)|^{p-1}=0$$

as $u(s)-\epsilon\le v(s),\forall s$. And similarly

$$sign(u(s)-\omega_p(v)+\epsilon)|u(s)-\omega_p(v)+\epsilon|^{p-1}\ge sign(v(s)-\omega_p(v))|v(s)-\omega_p(v)|^{p-1}=0,$$

as $u(s)+\epsilon\ge v(s),\forall s$. Summing over all states in the above inequalities, we get

$$g(\omega_p(v)-\epsilon)\ge 0\ge g(\omega_p(v)+\epsilon),$$

that implies root of g lies in $[\omega_p(v)-\epsilon,\omega_p(v)+\epsilon]$. That implies,

$$\|\omega_p(v)-\omega_p(v')\|\le 2\epsilon.$$

This implies the Lipschitzcity of ω_p.

Now we are in the position to prove our main claim. We have

$$\Big|\ \kappa_p(v)-\kappa_p(v')\ \Big| = \Big|\ \|v-\omega_p(v)\|_p-\|v'-\omega_p(v')\|_p\ \Big| \tag{48}$$

$$\le \big|\|v-v'-\omega_p(v)+\omega_p(v')\|_p, \qquad \text{(from reverse triangle inequality)} \tag{49}$$

$$\le \|v-v'\|_p-\|\omega_p(v)+\omega_p(v')\|_p, \qquad \text{(from triangle inequality)} \tag{50}$$

$$\le 3\|v-v'\|_p\le 3\epsilon. \tag{51}$$

Hence, we establish the Lipschitzcity of p-variance function κ_p. ■

Proposition 13 *Let the function $f:\mathbb{R}^A\to\mathbb{R}$ be defined as*

$$f(b)=x \qquad \textit{s.t.} \qquad \alpha^p=\sum_{i=1}^{A}(b_i-x)^p\mathbf{1}(b_i\ge x).$$

Then

$$\lim_{r\to\infty}\frac{f(rb)}{r}=\max_i b_i.$$

Proof Let $b_1:=\max_i b_i$ be the largest component of b, and let $b_2=\max\{b_i\ne b_1\mid 1\le i\le A\}$ be the second largest component. And let $\rho=|\{b_i=b_1\mid 1\le i\le A\}$ be the number of largest component.
Now, observe that there exist r_0 such that

$$r(b_1-b_2), \qquad \forall r\ge r_0.$$

This implies that

$$f(rb)=rb_1-\frac{\alpha}{\rho^{\frac{1}{p}}}, \qquad \forall r\ge r_0.$$

Taking the limit above, we get the desired result. ■

Proposition 14 $\lim_{r\to\infty} \frac{\psi_p(rb,rv)}{r}$ *is the solution to the following,*

$$(\gamma\beta\kappa_q(v))^p = \sum_{i=1}^{A}(b_i - x)^p \mathbf{1}(b_i \geq x).$$

This is equivalent of having reward uncertainty (α) zero.

Proof By definition, $\psi_p(rb, rv)$ for $r > 0$, satisfies the following

$$(\alpha + \gamma\beta\kappa_q(rv))^p = \sum_{i=1}^{A}(rb_i - \psi_p(rb,rv))^p \mathbf{1}(rb_i \geq \psi_p(rb,rv)) \tag{52}$$

$$\implies (\alpha + r\gamma\beta\kappa_q(v))^p = r^p \sum_{i=1}^{A}(b_i - \psi_p(rb,rv))^p \mathbf{1}(b_i \geq \psi_p(rb,rv)) \tag{53}$$

$$\implies (\frac{\alpha}{r} + \gamma\beta\kappa_q(v))^p = \sum_{i=1}^{A}(b_i - \psi_p(rb,rv))^p \mathbf{1}(b_i \geq \psi_p(rb,rv)). \tag{54}$$

$$\tag{55}$$

Taking the limit, we get the desired result. ■

Appendix B. L_p robust MDPs

Corollary 15 *The Q-value iteration,*

$$Q_{n+1}(s,a) = R_0(s,a) - \alpha_{sa} - \gamma\beta_{sa}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)\max_{a\in\mathcal{A}} Q_n(s',a), \tag{56}$$

where $v_n(s) = \max_{a\in\mathcal{A}} Q_n(s,a)$, then Q_n converges to $Q^*_{\mathcal{U}_p^{\text{sa}}}$ linearly for any initial Q-value Q_0.

Proof Let value iteration be define as follows

$$v_0(s) := \max_a Q_0(s,a), \qquad \text{and} \qquad v_{n+1} := \mathcal{T}^*_{\mathcal{U}_p^{\text{sa}}} v_n.$$

We will proceed by induction. Now, let us assume $v_n(s) = \max_a Q_n(s,a)$. We have

$$Q_{n+1}(s,a) := R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)\max_{a\in\mathcal{A}} Q_n(s',a) \tag{57}$$

$$= R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)v_n(s') \tag{58}$$

$$\implies \max_a Q_{n+1}(s,a) = \max_a \Big[R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)v_n(s') \Big] \tag{59}$$

$$= (\mathcal{T}^*_{\mathcal{U}_p^{\text{sa}}} v_n)(s) = v_{n+1}(s). \tag{60}$$

So from inductive arguments, we conclude

$$v_n(s) = \max_a Q_n(s,a), \qquad \forall s, n.$$

Now we have,

$$Q_{n+1}(s,a) = R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)v_n(s') \tag{61}$$

$$\implies \lim_{n\to\infty} Q_{n+1}(s,a) = \lim_{n\to\infty}\Big[\, R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v_n) + \gamma\sum_{s'} P_0(s'|s,a)v_n(s')\,\Big] \tag{62}$$

$$= R_0(s,a) - \alpha_{s,a} - \gamma\beta_{s,a}\kappa_q(v^*_{\mathcal{U}^s_p}) + \gamma\sum_{s'} P_0(s'|s,a)v^*_{\mathcal{U}^s_p}(s') \tag{63}$$

$$= Q^*_{\mathcal{U}^s_p}(s,a). \tag{64}$$

■

Proposition 16 *The Q-value iteration,*

$$Q_{n+1}(s,a) := R_0(s,a) + \gamma\sum_{s'} P_0(s'|s,a)\psi_p(Q_n,v_n)(s'), \tag{65}$$

$$v_{n+1} := \psi_p(Q_n,v_n). \tag{66}$$

Then Q_n, v_n converges to their optimal values, i.e. $\lim_{n\to\infty} Q_n = Q^*_{\mathcal{U}^s_p}, \lim_{n\to\infty} v_n = v^*_{\mathcal{U}^s_p}$ linearly, for any initial value function v_0 and $Q_0(s,a) := R_0(s,a) + \gamma P_0(s'|s,a)v_0(s')$.

Proof The above is just restatement of theorem 3 of Kumar et al. (2022). As it states

$$v_{n+1} = \mathcal{T}^*_{\mathcal{U}^s_p} v_n = \psi_p(Q_n,v_n)$$

where

$$Q_n(s,a) = R_0(s,a) + \gamma\sum_{s'} P_0(s'|s,a)v_n(s'), \tag{67}$$

$$\implies Q_{n+1}(s,a) = R_0(s,a) + \gamma\sum_{s'} P_0(s'|s,a)v_{n+1}(s'), \tag{68}$$

$$\implies Q_{n+1}(s,a) = R_0(s,a) + \gamma\sum_{s'} P_0(s'|s,a)\psi_p(Q_n,v_n)(s'). \tag{69}$$

$$\tag{70}$$

Now, lets study the convergence rate. Linear convergence rate of v_{n+1} follows from the contraction property of optimal robust Bellman operator. Now we move for attention to Q_n. Observe we have

$$\|Q_n - Q^*_{\mathcal{U}^s_p}\|_\infty = \|R_0 + \gamma P_0^T v_n - R_0 - \gamma P_0^T v^*_{\mathcal{U}^s_p}\|_\infty \tag{71}$$

$$= \gamma\|P_0^T v_n P_0^T v^*_{\mathcal{U}^s_p}\|_\infty \tag{72}$$

$$= \gamma\|P_0^T(v_n - v^*_{\mathcal{U}^s_p})\|_\infty \tag{73}$$

$$\leq \gamma\|v_n - v^*_{\mathcal{U}^s_p}\|_\infty \tag{74}$$

$$\leq \gamma^{n+1}\|v_0 - v^*_{\mathcal{U}^s_p}\|_\infty. \tag{75}$$

$$\tag{76}$$

This concludes the proof. ■

Appendix C. Main

C.1 sa-rectangular

Now, we see if the above stochastic approximation satisfies the assumption 1.

1. The function h is clearly Lipschitz as $\kappa_q(Q)$ and $\max_a Q(\cdot, a)$ is Lipschitz in Q.
2. We can take the step size η_n to satisfy to step size condition in assumption 1.
3. We have the same noise as in Borkar and Meyn (2000) (see section 3.2, Borkar and Meyn (2000)), nonetheless we show the noise satisfies the required conditions. M_n is clearly uncorrelated zero mean martingale noise. And

$$\mathbf{E}\left[\ \|M(n+1)\|^2 \ \middle|\ \mathcal{F}_n\ \right] = \gamma^2 \sum_{s,a} \mathbf{E}\left[\ |\max_{a\in\mathcal{A}} Q_n(s',a) - \sum_{s'} P_0(s'|s,a) \max_{a\in\mathcal{A}} Q_n(s',a)|^2 \ \middle|\ \mathcal{F}_n\ \right], \tag{77}$$

$$= \gamma^2 \sum_{s,a} \sum_{s'} P_0(s'|s,a) \ \Big|\ \max_{a'\in\mathcal{A}} Q_n(s',a') - \sum_{s'} P_0(s'|s,a) \max_{a'\in\mathcal{A}} Q_n(s',a')\ \Big|^2, \tag{78}$$

$$\leq \gamma^2 \sum_{s,a} \sum_{s'} P_0(s'|s,a) \ \Big|\ \max_{a'\in\mathcal{A}} Q_n(s',a')\ \Big|^2, \tag{79}$$

$$\leq \gamma^2 \sum_{s,a} \sum_{s'} \ \Big|\ \max_{a'\in\mathcal{A}} Q_n(s',a')\ \Big|^2, \tag{80}$$

$$\leq \gamma^2 \sum_{s,a} \sum_{s'} \sum_{a'} \ \Big|\ Q_n(s',a')\ \Big|^2, \tag{81}$$

$$= \gamma^2 SA \|Q_n\|^2. \tag{82}$$

   Observe that bounds are very loose, but good enough for the purpose.

4. The limiting ODE is

$$h_\infty(Q)(s,a) := \lim_{r\to\infty} \frac{h(rQ)(s,a)}{r} \tag{83}$$

$$= -\gamma\beta_{s,a} \lim_{r\to\infty} \frac{\kappa_q(rQ)}{r} + \lim_{r\to\infty} \frac{\gamma \sum_{s'} P_0(s'|s,a) \max_{a\in\mathcal{A}} rQ(s',a) - rQ(s,a)}{r} \tag{84}$$

$$= \underbrace{-\gamma\beta_{s,a}\kappa_q(Q) + \gamma \sum_{s'} P_0(s'|s,a) \max_{a\in\mathcal{A}} Q(s',a)}_{:=F_\infty(Q)(s,a)} - Q(s,a). \tag{85}$$

$$\tag{86}$$

   The equation follows from the fact that q-variance function κ_q is scalar linear (i.e. $\kappa_q(rQ) = r\kappa_q(Q)$). We have additional $-\kappa_q(Q)$ term in the ODE above as compared to the limiting ODE of non-robust ODE (see section 3.2 of Borkar and Meyn (2000)). The map F is γ-contraction map w.r.t max norm $\|\cdot\|_\infty$. Why? Recall $\mathcal{T}^*_{\mathcal{U}^{sa}_p}$ is γ-contraction map w.r.t. max norm $\|\cdot\|_\infty$ where $\mathcal{U}^{sa}_p = (R_0 + \mathcal{R}) \times (P_0 + \mathcal{P})$. Now let nominal reward function $R_0 = \mathbf{0}$ be zero function and reward noise set $\mathcal{R} = \{\mathbf{0}\}$ be singleton zero set. In conclusion, $\mathcal{U}^{sa}_p = (\mathbf{0}) \times (P_0 + \mathcal{P})$, and then verify that

$$\mathcal{T}^*_{\mathcal{U}^{sa}_p} Q = F(Q).$$

   It is clear that the fixed point of $\mathcal{T}^*_{\mathcal{U}^{sa}_p}$ is $\mathbf{0}$. Now, we have the similar setting as in Borkar and Meyn (2000) (see section Borkar and Meyn (2000)). Same as Borkar and Meyn (2000), we conclude that the global asymptotic stability of unique equilibrium point of h_∞ , is a special case of the results of Borkar and Soumyanatha (1997).

5. From above discussion, we also conclude same as Borkar and Meyn (2000) (see section Borkar and Meyn (2000)), the global asymptotic stability of unique equilibrium point of h, follows from Borkar and Soumyanatha (1997).

C.2 s-rectangular

Now let us consider the stochastic version of proposition 3, as follows

$$Q_{n+1}(s,a) := Q_n(s,a) + \eta_n^1 \left[R_0(s,a) + \gamma v_n(s') - Q_n(s,a) \right], \qquad s' \sim P_0(\cdot|s,a), \tag{87}$$

$$v_{n+1}(s) := v_n(s) + \eta_n^2 \left[\psi_p(Q_n, v_n)(s) - v_n(s) \right] \tag{88}$$

Let v_0, Q_0 be any initial value function and Q-value respectively. This can be re-framed as

$$Q_{n+1} = Q_n + \eta_n^1 \left[h^1(Q) + M_{n+1} \right], \qquad v_{n+1} = v_n + \eta_n^2 \left[h^2(Q_n, v_n) \right]. \tag{89}$$

where

$$h^1(Q,v)(s,a) := R_0(s,a) + \gamma \sum_{s'} P_0(s'|s,a) v(s') - Q(s,a), \qquad h^2(Q,v) := \psi_p(Q,v) - v$$

and

$$M_{n+1}(s,a) := \gamma v_n(s') - \gamma \sum_{s'} P_0(s'|s,a) v_n(s'), \qquad s' \sim P_0(\cdot|s,a).$$

We want to point out that this is a two time scale algorithm, Q-update being the fast time timescale and value function update being the slow one. Now we show that the above stochastic algorithm satisfies algorithm 3.

1. The function $\psi_p(Q,v)$ is Lipschitz in Q and v (see proposition 9). Consequently the function $h^2(Q,v)$ is also Lipschitz in Q and v. And Lipschitzcity of h^1 is direct.

2. We can take the step size η_n^1, η_n^2 to satisfy to step size condition in assumption 3.

3. M_n is clearly uncorrelated zero mean martingale noise. And

$$\mathbf{E}\left[\|M(n+1)\|^2 \,\middle|\, \mathcal{F}_n \right] = \gamma^2 \sum_{s,a} \mathbf{E}\left[|v_n(s) - \sum_{s'} P_0(s'|s,a) v_n(s')|^2 \,\middle|\, \mathcal{F}_n \right], \tag{90}$$

$$= \gamma^2 \sum_{s,a} \sum_{s'} P_0(s'|s,a) \left| v_n(s') - \sum_{s'} P_0(s'|s,a) v_n(s') \right|^2, \tag{91}$$

$$\leq \gamma^2 \sum_{s,a} \sum_{s'} P_0(s'|s,a) \left| v_n(s') \right|^2, \tag{92}$$

$$\leq \gamma^2 \sum_{s,a} \sum_{s'} \left| v_n(s') \right|^2, \tag{93}$$

$$= \gamma^2 SA \left\| v_n \right\|^2, \tag{94}$$

$$\tag{95}$$

   Observe that bounds are very loose, but good enough for the purpose.

4. The ODE

$$\dot{x}(t) = h^1(x(t), y) = R_0 + \gamma P_0^T y - x(t)$$

   has the following solution,

$$x(t) = R_0 + \gamma P_0^T y + \left(R_0 + \gamma P_0^T y - x(0) \right) e^{-t}.$$

   It clear that the above ODE has a globally asymptotically stable equilibrium $\lambda(y) := R_0 + \gamma P_0^T y$ (uniformly in y), and λ is a Lipschitz map.

5. The ODE

$$\dot{y}(t) = h^2(\lambda(y(t)), x(t)) = \psi_p(\lambda(y(t)), y(t)) - y(t),$$

   has a globally asymptotically stable equilibrium $v^*_{\mathcal{U}_p^s}$. Why? The map $F(v) := \psi_p(R_0 + \gamma P_0^T v, v)$ is a contraction (see theorem 3 Kumar et al. (2022)), and then the rest follows from Borkar and Soumyanatha (1997).

6. The function h_r^1, defined as

$$h_r^1(x,y) := \frac{h^1(rx,ry)}{r} \tag{96}$$
$$= \frac{R_0 + \gamma P_0^T ry - rx}{r} \tag{97}$$
$$= \frac{R_0}{r} + \gamma P_0^T y - x \tag{98}$$

converges uniformly to h_∞^1, defined as $h_\infty^1(x,y) = \gamma P_0^T y - x$. The limiting ODE

$$\dot{x}(t) = h_\infty^1(x(t), y) = \gamma P_0^T y - x(t),$$

has a globally asymptotically stable equilibrium $\lambda_\infty(y) = \gamma P_0^T y$. The function λ_∞ is linear hence Lipschitz. Further, verify that $\lambda_\infty(0) = 0$. The ODE

$$\dot{x}(t) = h_\infty^1(x(t), 0) = -x(t),$$

has the origin as its unique globally asymptotically stable equilibrium.

7. The function h_r^2 is defined as

$$h_r^2(y) := \frac{h^2(r\lambda_\infty(y), ry)}{r} \tag{99}$$
$$= \frac{\psi_p(r\lambda_\infty(y), ry) - ry}{r} \tag{100}$$
$$= \frac{\psi_p(\gamma r P_0^T y, ry)}{r} - y \tag{101}$$
$$\tag{102}$$

converges to h_∞^2, defined as

$$h_\infty^2(y) := \psi_p^\infty(\gamma P_0^T y, y) - y,$$

where $\psi_p^\infty(Q,v)(s)$ is the solution to

$$(\gamma \beta_s \kappa_q(v))^p = \sum_a \left(Q(s,a) - x \right)^p \mathbf{1}\left(Q(s,a) \geq x \right). \tag{103}$$

The limiting ODE

$$\dot{y}(t) = h_\infty^2(y(t)) = \psi_p^\infty(\gamma P_0^T y(t), y(t)) - y(t),$$

has origin as its globally asymptotically stable equilibrium. Why? The map $\psi_p^\infty(\gamma P_0^T y, y)$ is still contraction that guarantees the globally asymptotically stable equilibrium point. Furhte, the map $\psi_p^\infty(\gamma P_0^T y, y)$ corresponds to the case where nominal reward is zero, and the uncertainty in reward is zero, that implies the globally asymptotically stable equilibrium point has to be origin.

Appendix D. Sample Based Algorithms

In this section, we assume that we don't have the access to the nominal transition kernel but only to its samples. Algorithm 6, algorithm 5, algorithm 3, algorithm 4 is sample based algorithm for (sa)-rectangular L_1 robust MDPs, (sa)-rectangular L_2 robust MDPs, (s)-rectangular L_1 robust MDPs, (s)-rectangular L_2 robust MDPs respectively.

Observe that in algorithm 3, we need to calculate initial value variance $(v_{max} - v_{min})$ which can be difficult to compute for large state spaces. This can be remedy in number of ways: a) initializing Q-values to 0 b) initializing Q-values randomly in a bounded range, say $[-1, 1]$, with nice enough distribution then $v_{max} \approx 1$ and $v_{min} \approx -1$ for large state spaces (for small state spaces, it can be calculated directly) c) starting with any arbitrary values, may also work that we leave for future works.

Algorithm 3 Sample Based Regularized Q-Learning Algorithm for S Rectangular L_1 Robust MDP

Choose appropriate step sizes η_n and exploration probability ϵ_n for $n \geq 0$. Take initial Q-values Q_0 randomly, and sample initial state s_0 from initial distribution μ. Calculate initial value peak values $v_{max} = \max_{s\in\mathcal{S},a\in\mathcal{A}} Q_0(s,a), v_{min} = \min_{s\in\mathcal{S},a\in\mathcal{A}} Q_0(s,a)$

Input: α_s, β_s are uncertainty radius in reward and transition kernel respectively in state s.

while not converged **do**

Update reward regularizer.

$$\sigma = \alpha_{s_n} + \gamma\beta_{s_n}\frac{v_{max} - v_{min}}{2}$$

Sort the Q-value such that

$$Q(s_n, a_1) \geq Q(s_n, a_2), \cdots, \geq Q(s_n, a_A)$$

Update value function:

$$v_{n+1}(s_n) = \max_m \frac{\sum_{i=1}^m Q(s_n, a_i) - \sigma}{m}$$

.

Get optimal policy:

$$\pi_n(a|s_n) = \frac{\mathbf{1}(Q_n(s_n,a) \geq v_{n+1}(s_n))}{\sum_a \mathbf{1}(Q_n(s_n,a) \geq v_{n+1}(s_n))}$$

With probability $1-\epsilon_n$, play optimal policy

$$a_n \sim \pi_n(\cdot|s_n)$$

and with probability ϵ_n play exploratory action.

Get next state s_{n+1} from the environment.

Update Q-value as

$$Q_{n+1}(s_n,a_n) = Q_n(s_n,a_n) + \eta_n[R(s_n,a_n) + \gamma v_{n+1}(s_{n+1}) - Q_n(s_n,a_n)].$$

Update the value peak values

$$v_{max} = \max(v_{max}, v_{n+1}(s_n)), \qquad v_{min} = \max(v_{min}, v_{n+1}(s_n)).$$

end while

Algorithm 4 Sample Based Q-Learning Algorithm for S Recantangular L_2 Robust MDP

Choose appropriate step sizes η_t and exploration probability ϵ_t for $t \geq 0$. Take initial Q-values Q_0 randomly, and sample initial state s_0 from initial distribution q. Calculate initial value mean $\mu_0 = \sum_{s\in\mathcal{S}} \max_{a\in\mathcal{A}} Q_0(s,a)$, and calculate initial value second moment $\rho_0 = \sum_{s\in\mathcal{S}} (\max_{a\in\mathcal{A}} Q_0(s,a))^2$.

Input: α_s, β_s are uncertainty radius in reward and transition kernel respectively in state s.

while not converged **do**

Sort the actions according to their Q-values, that is

$$Q_n(s_n, a_i) \geq Q_n(s_n, a_{i+1}), \qquad \forall i.$$

Update reward regularizer.

$$\sigma = \alpha_s + \gamma\beta_s\sqrt{\rho_n - (\mu_n)^2}$$

Solve for value function. Set $v_{n+1}(s_n) = Q_n(s_n, a_1) - \sigma$, and $k = 1$

while $v_{n+1}(s_n) < Q_n(s_n, a_k)$ and $k \leq A - 1$ **do**

$$k = k + 1 \tag{104}$$

$$v_{n+1}(s_n) = \frac{1}{k}[\sum_{i=1}^{k} Q_n(s_n, a_i) - \sqrt{k\sigma^2 + (\sum_{i=1}^{k} Q_n(s_n, a_i))^2 - k\sum_{i=1}^{k}(Q_n(s_n, a_i))^2}], \tag{105}$$

end while

Get optimal policy:

$$\pi_n(a|s_n) = \frac{(Q_n(s_n,a) - v_{n+1}(s_n))\mathbf{1}(Q_n(s_n,a) > v_{n+1}(s_n))}{\sum_a (Q_n(s_n,a) - v_{n+1}(s_n))\mathbf{1}(Q_n(s_n,a) > v_{n+1}(s_n))}$$

With probability $1 - \epsilon_t$, play optimal policy

$$a_n \sim \pi_n(\cdot|s_n)$$

and with probability ϵ_n play exploratory action.

Get next state s_{n+1} from the environment.

Update Q-value as

$$Q_{n+1}(s_n, a_n) = Q_n(s_n, a_n) + \eta_n[R_0(s_n, a_n) + \gamma v_{n+1}(s_{n+1}) - Q_n(s_n, a_n)].$$

Update the value mean (first moment)

$$\mu_{n+1} = \mu_n + \frac{\max_a Q_{n+1}(s_n, a) - \max_a Q_n(s_n, a)}{S}$$

Update the value second moment

$$\rho_{n+1} = \rho_n + (\max_a Q_{t+1}(s_t, a))^2 - (\max_a Q_t(s_t, a))^2$$

end while

Algorithm 5 Sample Based Q-Learning Algorithm for SA Recantangular L_2 Robust MDP

Choose appropriate step sizes η_t and exploration probability ϵ_t for $t \geq 0$. Take initial Q-values Q_0 randomly, and sample initial state s_0 from initial distribution q. Calculate initial value mean $\mu_0 = \sum_{s\in\mathcal{S}} \max_{a\in\mathcal{A}} Q_0(s,a)$, and calculate initial value second moment $\rho_0 = \sum_{s\in\mathcal{S}} (\max_{a\in\mathcal{A}} Q_0(s,a))^2$.

Input: $\alpha_{s,a}, \beta_{s,a}$ are uncertainty radius in reward and transition kernel respectively in state s and action a.

while not converged **do**

With probability $1-\epsilon_t$, play a best action

$$a_t \in arg\max_{a\in\mathcal{A}} Q_t(s_t,a),$$

and with probability ϵ_t play exploratory action.

Get next state s_{t+1} from the environment.

Get effective reward

$$r_t = R_0(s_t,a_t) - \alpha_{s_t,a_t} - \gamma\beta_{s_t,a_t}\sqrt{\rho_t - (\mu_t)^2}.$$

Update Q-value as

$$Q_{t+1}(s_t,a_t) = Q_t(s_t,a_t) + \eta_t[r_t + \gamma \max_a Q_t(s_{t+1},a) - Q_t(s_t,a_t)].$$

Update the value mean (first moment)

$$\mu_{t+1} = \mu_t + \frac{\max_a Q_{t+1}(s_t,a) - \max_a Q_t(s_t,a)}{S}$$

Update the value second moment

$$\rho_{t+1} = \rho_t + (\max_a Q_{t+1}(s_t,a))^2 - (\max_a Q_t(s_t,a))^2$$

end while

Algorithm 6 Sample Based Regularized Q-Learning Algorithm for SA Recantangular L_1 Robust MDP

Choose appropriate step sizes η_n and exploration probability ϵ_n for $n \geq 0$. Take initial Q-values Q_0 randomly, and sample initial state s_0 from initial distribution μ. calculate value peak values: $v_{max} = \max_{s\in\mathcal{S}} \max_{a\in\mathcal{A}} Q_0(s,a), v_{min} = \min_{s\in\mathcal{S}} \max_{a\in\mathcal{A}} Q_0(s,a)$

while not converged **do**

With probability $1-\epsilon_n$, play a best action

$$a_n \in arg\max_{a\in\mathcal{A}} Q_n(s_n,a)$$

and with probability ϵ_n play exploratory action.

Get next state s_{n+1} from the environment and effective reward

$$r_n = R_0(s_n,a_n) - \alpha_{s_n,a_n} - \gamma\beta_{s_n,a_n}\frac{v_{max} - v_{min}}{2}.$$

Update Q-value as

$$Q_{n+1}(s_n,a_n) = Q_n(s_n,a_n) + \eta_n[r_n + \gamma \max_a Q_n(s_{n+1},a) - Q_n(s_n,a_n)].$$

Update the value peak values

$$v_{max} = \max(v_{max}, \max_a Q_{n+1}(s_n,a)), \qquad v_{min} = \max(v_{min}, \max_a Q_{n+1}(s_n,a))$$

end while