APPENDIX

In the subsequent sections, we delve into the experimental specifics and provide the technical proofs that were not included in the primary content.

In Section B, we commence by showcasing an additional experiment on the American call option. This aligns with the convergence and sample complexity discussions from the main content. We then elucidate the intricacies of Liu's algorithm to facilitate a transparent comparison with our methodology. Lastly, we discuss the algorithmic intricacies of our DDRQ algorithm and provide details on the experiments that were previously omitted.

In Section C, to prove Theorem 3.3, we begin by extending the two-timescale stochastic approximation framework to a three-timescale one. Following this, we adapt it to our algorithm, ensuring all requisite conditions are met.

A NOTATIONS

We fix some notations that will be used in the appendix. For a positive integer n, $[n]$ denotes the set $\{1, 2, \cdots, n\}$. $|A|$ denotes the cardinality of the set A. We adopt the standard asymptotic notations: for two non-negative sequences a_n and b_n, $a_n = O(b_n)$ iff $\limsup_{n\to\infty} a_n/b_n < \infty$. Δ_d is the simplex on a d dimensional space, i.e., $\Delta_d = \{x : \sum_{i=1}^d x_i = 1, x_i \geq 0, \forall i \in [d]\}$. For any vector $x \in \mathbb{R}^d$ and any semi-positive matrix $A \in \mathbb{R}^{d\times d}$ with $A \succeq 0$, we denote $\|x\|_A \coloneqq \sqrt{x^\top Ax}$. $\|\cdot\|$ is Euclidean norm.

B ADDITIONAL EXPERIMENTS DETAILS

B.1 EXPERIMENT ON THE AMERICAN PUT OPTION PROBLEM

In this section, we present additional experimental results from a simulated American put option problem (Cox et al., 1979) that has been previously studied in robust RL literature (Zhou et al., 2021; Tamar et al., 2014). The problem involves holding a put option in multiple stages, whose payoff depends on the price of a financial asset that follows a Bernoulli distribution. Specifically, the next price s_{h+1} at stage $h+1$ follows,

$$s_{h+1} = \begin{cases} c_u s_h, & \text{w.p. } p_0, \\ c_d s_h, & \text{w.p. } 1-p_0, \end{cases} \tag{12}$$

where the c_u and c_d are the price up and down factors and p_0 is the probability that the price goes up. The initial price s_0 is uniformly sampled from $[\kappa-\epsilon, \kappa+\epsilon]$, where $\kappa = 100$ is the strike price and $\epsilon = 5$ in our simulation. The agent can take an action to exercise the option ($a_h = 1$) or not exercise ($a_h = 0$) at the time step h. If exercising the option, the agent receives a reward $\max(0, \kappa - s_h)$ and the state transits into an exit state. Otherwise, the price will fluctuate based on the above model and no reward will be assigned. Moreover we introduce a discount structure in this problem, i.e., the 1 reward in the stage $h+1$ worths γ in stage h as our algorithm is designed for discounted RL setting. In our experiments, we set $H = 5$, $c_u = 1.02$, $c_d = 0.98$ and $\gamma = 0.95$. We limit the price in $[80, 140]$ and discretize with the precision of 1 decimal place. Thus the state space size $|\mathcal{S}| = 602$.

We first demonstrate the robustness gain of our DR Q-learning algorithm by comparing with the non-robust Q-learning algorithm, and investigate the effect of different robustness levels by varying ρ. Each agent is trained for 10^7 steps with an ϵ-greedy exploration policy of $\epsilon = 0.2$ and evaluated in perturbed environments. We use the same learning rates for the three timescales in our DR Q-learning algorithm as in the Cliffwalking environment: $\zeta_1(t) = 1/(1+(1-\gamma)t^{0.6})$, $\zeta_2(t) = 1/(1+0.1*(1-\gamma)t^{0.8})$, and $\zeta_3(t) = 1/(1+0.01*(1-\gamma)t)$. For the non-robust Q-learning we set the same learning rate as in our Q-update, i.e., $\zeta_3(t)$. We perturb the transition probability to the price up and down status $p = \{0.3, 0.4, 0.5, 0.6, 0.7\}$, and evaluate each agent for 5000 episodes. Figure 6 reports the average return and one standard deviation level. The non-robust Q-learning performs best when the price tends to decrease and the market gets more benefitial ($p = \{0.3, 0.4, 0.5\}$), which benefits the return of holding an American put option. However, when the prices tend to increase and the market is riskier ($p = \{0.6, 0.7\}$), our DR Q-learning algorithm significantly outperforms

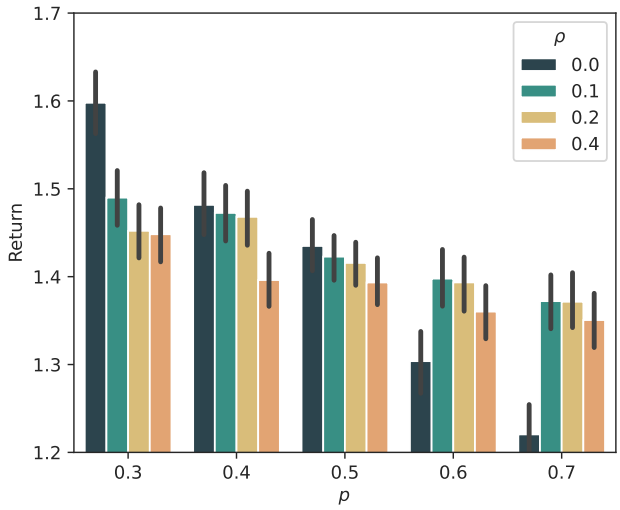

Figure 6: Averaged return in the American call option problem. $\rho = 0.0$ is the non-robust Q-learning.

the non-robust counterpart, demonstrating the robustness gain of our algorithm against worst-case scenarios.

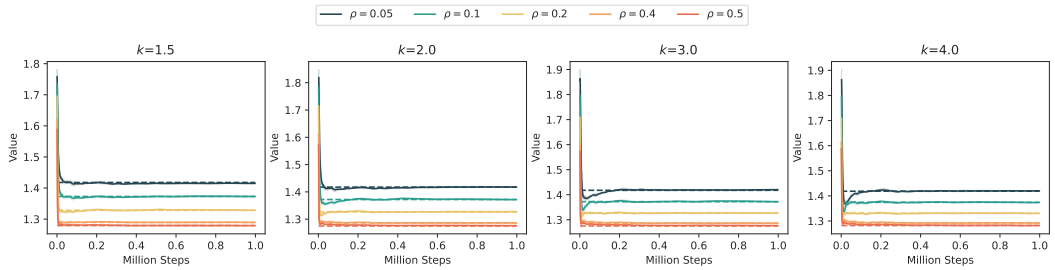

Figure 7: Convergence curve of DR Q-learning algorithm to the true DR value under different ρ's and k's. Each curve is averaged over 10 random seeds and shaded by their standard deviation. The dashed line is the optimal robust value with corresponding k and ρ.

We present the learning curve of our DR Q-learning algorithm with different ρ in Figure 7. Our algorithm can accurately learn the DR value under different ρ's and k's within 0.1 million steps. We compare the sample efficiency of our algorithm with the DR Q-learning algorithm in Liu et al. (2022) (referred to as *Liu's*) and the model-based algorithm in Panaganti & Kalathil (2022) (referred to as *Model*). We set a smaller learning rate for Liu's as $\zeta(t) = 1/(1 + (1 - \gamma)t)$. The reason is setting the same learning rate $\zeta_3(t)$ for their algorithm would render a much slower convergence performance, which is not fair for comparisons. We use the recommended choice $\varepsilon = 0.5$ for the sampling procedure in Liu algorithm. Both DR Q-learning and Liu are trained for $5 * 10^7$ steps per run, while the model-based algorithm is trained for 10^6 steps per run to ensure sufficient samples for convergence. As shown in Figure 8, the model-based approach is the most sample-efficient, converging accurately to the optimal robust value with less than 10^4 samples. Our DR Q-learning algorithm is slightly less efficient, using 10^5 samples to converge. Liu algorithm is significantly less efficient, using 10^7 samples to converge. Note that the model-based approach we compared here is to first obtain samples for each state-action pairs, and then conduct the learning procedure to learn the optimal robust value. In particular, we need to specify the number of samples for each state-action pair n. Then the total number of samples used is the sum of all these number, i.e., $S \times A \times n$, whose computation manner is different from that in the model-free algorithms we used where each update requires one or a batch of new samples.

To ensure self-containment, we provide the pseudocode for our implemented Liu algorithm (Algorithm 3) and the model-based algorithm (Algorithm 2) below. These algorithms were not originally designed to solve the ambiguity set constructed by the Cressie-Read family of f-divergences.

B.2 Liu's Algorithm Descriptions

In this subsection, we provide the pseudo-code for the Liu algorithm, represented in Algorithm 2. Our intention is to emphasize the differences in algorithmic design between their approach and ours.

Their algorithm, in particular, relies extensively on multi-level Monte Carlo, requiring the sampling of a batch of samples for each state-action pair. Once they estimate the Doubly Robust (DR) value for a specific state-action pair, the samples are promptly discarded and subsequently resampled from a simulator. To summarize, their algorithm exhibits significant distinctions from ours in terms of algorithmic design.

Algorithm 2 Distributionally Robust Deep Q-learning with Cressie-Read family of f-divergences

Input: Discount Factor γ, Radius of robustness ρ, Cressie-Read family parameter k, Q-network target update rate τ_Q and η-network target update rate τ_η, mini-batch size N, maximum number of iterations T, start training timestep T_{tr}, training network update frequency F_{tr} and target network update frequency F_{up}.
Init: Two state-action neural networks Q_{θ_1} and Q_{θ_2}, two dual neural network η_{θ_1} and η_{θ_2}, $C = (1 + k * (k-1) * \rho)^{1/k}$.
for for $t = 1, \cdots, T$ **do**
  Observe a state s_t and execute an action a_t using ϵ-greedy policy.
  if $t \geq T_{tr}$ and $t\%F_{tr}$ **then**
    Sample a minibatch B with N samples from the replay buffer.
    Compute next-state target value for Q network

$$Q_i = r_t - \gamma C * (\eta_{\theta_1}(s_i, a_i) - \max_{a \in \mathcal{A}} Q_{\theta_1}(s_i, a_i))_+^{k_*}, \quad \forall i \in B$$

    and for η network

$$Q'_i = r_t - \gamma C * (\eta_{\theta_2}(s_i, a_i) - \max_{a \in \mathcal{A}} Q_{\theta_2}(s_i, a_i))_+^{k_*}, \quad \forall i \in B.$$

    Update $\theta_1 = \arg\min_\theta \sum_i (Q_i - Q_\theta(s_i, a_i))^2$.
    Update $\theta_3 = \arg\max_\theta \sum_i Q'_i(\theta)$.
  end if
  if $t \geq T_{tr}$ and $t\%F_{up}$ **then**
    Update target network $\theta_2 = (1 - \tau_Q)\theta_2 + \tau_Q\theta_2$, $\theta_4 = (1 - \tau_\eta)\theta_4 + \tau_\eta\theta_3$.
  end if
end for
$t = t + 1$

B.3 Practical Experiments

In this section, we provide a comprehensive description of our Deep Distributionally Robust Q-learning (DDRQ) algorithm, as illustrated in Algorithm 1, along with its experimental setup in the context of CaroPole and LunarLander.

Our practical algorithm, denoted as Algorithm 4, is a variant of Algorithm 1. Specifically, we adopt the Deep Q-Network (DQN) architecture (Mnih et al., 2015) and employ two sets of neural networks as functional approximators. One set, Q_{θ_1} and Q_{θ_2}, serves as approximators for the Q function, while the other set, η_{θ_3} and η_{θ_4}, approximates the distributionally robust dual variable η. To enhance training stability, we introduce a target network, Q_{θ_2}, for the fast Q network Q_{θ_1} and η_{θ_4} for the fast dual variable η network η_{θ_3}.

Due to the approximation error introduced by neural networks and to further improve sample efficiency, our practical DDRQ algorithm adopts a two-timescale update approach. In this approach,

Algorithm 3 Distributionally Robust Q-learning with Cressie-Read family of f-divergences **with Simulator**

Input: Exploration rate ϵ, Learning rates $\{\zeta_i(n)\}_{i\in[3]}$, Ambiguity set radius $\rho > 0$, parameter $\varepsilon \in (0, 0.5)$
Init: $\widehat{Q}(s,a) = 0, \forall (s,a) \in \mathcal{S} \times \mathcal{A}$
while Not Converge **do**
  for every $(s,a) \in \mathcal{S} \times \mathcal{A}$ **do**
    Sample $N \in \mathbb{N}$ from $P(N = n) = p_n = \varepsilon(1-\varepsilon)^n$.
    Draw 2^{N+1} samples $\{(r_i, s_i')\}_{i\in[2^{N+1}]}$ from the simulator
    Compute $\Delta^r_{N,\rho}$ via

$$\Delta^r_{N,\rho} = \sup_{\eta\in\mathbb{R}} \widehat{\sigma}^r_k([2^{N+1}], \eta) - \frac{1}{2}\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^r_k([2^N], \eta) - \frac{1}{2}\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^r_k([2^N :], \eta),$$

    where

$$\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^r_k(I, \eta) = \sup_{\eta\in\mathbb{R}}\{-c_k(\rho)[\sum_{i\in I}(\eta - r_i)_+^{k_*}/n]^{\frac{1}{k_*}} + \eta\},$$

    and $[2^N] = \{1, 2, 3, \cdots, 2^N\}$ and $[2^N :] = \{2^N, 2^N + 1, \cdots, 2^{N+1}\}$.
    Compute $\Delta^q_{N,\rho}(\widehat{Q}_t)$ via

$$\Delta^q_{N,\rho}(\widehat{Q}_t) = \sup_{\eta\in\mathbb{R}} \widehat{\sigma}^q_k(\widehat{Q}_t, [2^{N+1}], \eta) - \frac{1}{2}\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^q_k(\widehat{Q}_t, [2^N], \eta) - \frac{1}{2}\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^q_k(\widehat{Q}_t, [2^N :], \eta),$$

    where

$$\sup_{\eta\in\mathbb{R}} \widehat{\sigma}^q_k(\widehat{Q}_t, I, \eta) = \sup_{\eta\in\mathbb{R}}\{-c_k(\rho)[\sum_{i\in I}(\eta - \max_{a'\in\mathcal{A}} \widehat{Q}_t(s_i', a'))_+^{k_*}/n]^{\frac{1}{k_*}} + \eta\}.$$

    Set $R_\rho(s,a) = r_1 + \frac{\Delta^r_{N,\rho}}{p_N}$.
    Update Q via

$$\widehat{Q}_{t+1}(s,a) = (1-\zeta_t)\widehat{Q}_t(s,a) + \zeta_t \widehat{\mathcal{T}}_\rho(\widehat{Q}_t)(s,a),$$

    where

$$\widehat{\mathcal{T}}_\rho(\widehat{Q}_t)(s,a) = r_1 + \Delta^r_{N,\rho} + \gamma(\max_{a'\in\mathcal{A}} \widehat{Q}_t(s_1, a') + \frac{\Delta^q_{N,\rho}(\widehat{Q}_t)}{p_N}).$$

  end for
  $t = t + 1$
end while

our Q network aims to minimize the Bellman error, while the dual variable η network strives to maximize the DR Q value defined in Equation 5. It's important to note that the two-timescale update approach could introduce bias in the convergence of the dual variable, and thus the dual variable η may not the optimal dual variable for the primal problem. Given the primal-dual structure of this DR problem, this could render an even lower target value for the Q network to learn. This approach can be understood as a robust update strategy for our original DRRL problem, share some spirits to the optimization techniques used in other algorithms like Variational Autoencoders (VAE)(Kingma & Welling, 2013), Proximal Policy Optimization (PPO)(Schulman et al., 2017), and Maximum a Posteriori Policy Optimization (MPO) (Abdolmaleki et al., 2018).

Most of the hyperparameters are set the same for both LunarLander and CartPole. We choose Cressie-Read family parameter $k = 2$, which is indeed the χ^2 ambiguity set and we set ambiguity set radius as $\rho = 0.3$. For RFQI we also use the same ρ for fair comparison. Our replay buffer size is set $1e6$ and the batch size for training is set 4096. Our fast Q and η network are update every 10 steps ($F_{tr} = 10$) and the target networks are updated every 500 steps ($F_{up} = 500$). The learning rate for Q network is 2.5×10^{-4} and for η network is 2.5×10^{-4}. The Q network and the η network both

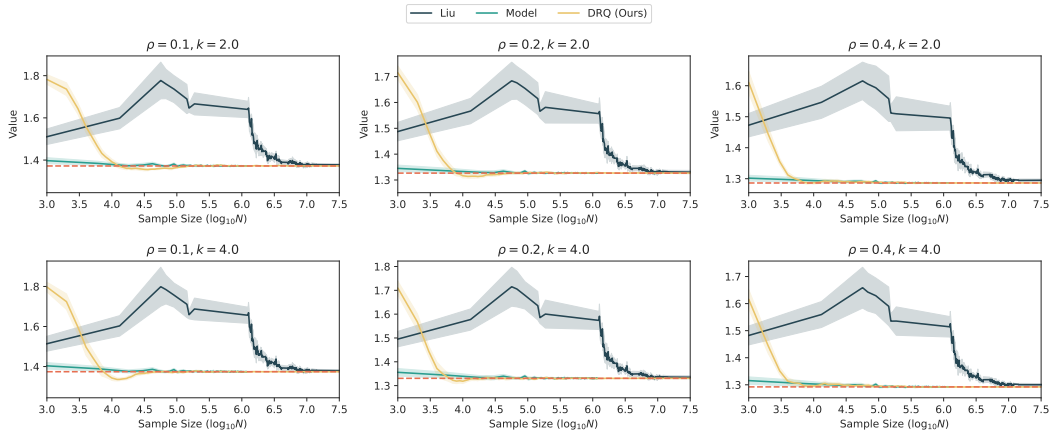

Figure 8: Sample complexity comparisons in American option environment with other DRRL algorithms. The dashed line is the optimal robust value with corresponding k and ρ. The x-axis is in log10 scale. Each curve is averaged over 10 random seeds and shaded by their one standard deviation. The dashed line is the optimal robust value with corresponding k and ρ.

Algorithm 4 Distributionally Robust DQN with Cressie-Read family of f-divergences (DDRQ)

1: **Input:** Exploration rate ϵ, Learning rates $\{\zeta_i(n)\}_{i\in[3]}$
2: **Init:** $Q(s,a) = 0, \forall (s,a) \in \mathcal{S} \times \mathcal{A}$
3: **for** $n = 1, 2, \cdots$ **do**
4: Observe the state s_n, execute the action $a_n = \arg\max_{a\in\mathcal{A}} Q_n(s_n, a)$ using ϵ-greedy policy
5: Observe the reward r_n and next state s'_n
6: Update $Z_{n+1,1}(s_n,a_n) = (1-\zeta_1(n))Z_{n,1}(s_n,a_n)+\zeta_1(n)(\eta_n(s_n,a_n)-\max_a Q_n(s'_n,a))_+^{k_*}$, and $Z_{n+1,2}(s_n,a_n) = (1-\zeta_1(n))Z_{n,2}(s_n,a_n)+\zeta_1(n)(\eta_n(s_n,a_n)-\max_a Q_n(s'_n,a))_+^{k_*-1}$.
7: Update $\eta_{n+1}(s_n,a_n) = \eta_n(s_n,a_n) + \zeta_2(n)(-c_k(\rho)Z_{n+1,1}^{\frac{1}{k_*}-1}(s_n,a_n)\cdot Z_{n+1,2}(s_n,a_n)+1)$.
8: Update $Q_{n+1}(s_n,a_n) = (1-\zeta_3(n))Q_n(s_n,a_n) + \zeta_3(n)(r_n - \gamma(c_k(\rho)Z_{n+1,1}^{\frac{1}{k_*}}(s_n,a_n) - \eta_n(s_n,a_n)))$.
9: **end for**

employ a dual-layer structure, with each layer consisting of 120 dimensions. For exploration scheme, we choose epsilon-greedy exploration with linearly decay epsilon with ending ϵ_{End}. The remain parameters tuned for each environments are referred in Table 1.

C Multiple Timescale Convergence

C.1 Three Timescales Convergence Analysis

In this subsection, we outline the roadmap for establishing the a.s. convergence of the Algorithm 1. For ease of presentation, our analysis is given for the synchronous case, where every entry of the Q function is updated at each timestep. Extension to the asynchronous case, where only one state-action pair entry is updated at each timestep, follows Tsitsiklis (1994). Our approach is to generalize the

Environment	Maximum Training Step T	ϵ_{End}	τ_Q	τ_η
CartPole	$1e8$	0.05	1	0.05
LunarLander	$3e7$	0.2	0.5	0.1

Table 1: Different Hyperparamers between CartPole and LunarLander

classic machinery of two-timescale stochastic approximation (Borkar, 2009) to a three-timescale framework, and use it to analyze our proposed algorithm. We rewrite the Algorithm 1 as

$$Z_{n+1} = Z_n + \zeta_1(n)[f(Z_n, \eta_n, Q_n) + M_n^Z], \tag{13}$$

$$\eta_{n+1} = \eta_n + \zeta_2(n)[g(Z_n, \eta_n, Q_n) + \epsilon_n^\eta], \tag{14}$$

$$Q_{n+1} = Q_n + \zeta_3(n)[h(Z_n, \eta_n, Q_n) + \epsilon_n^Q]. \tag{15}$$

Here, we use $Z_n = (Z_{n,1}, Z_{n,2})$ to represent the $Z_{n,1}$ and $Z_{n,2}$ jointly. To echo with our algorithm, $f = (f_1, f_2)$ and $M_n^Z = (M_{n,1}^Z, M_{n,2}^Z)$ are defined as,

$$f_1(Z_n, \eta_n, Q_n)(s,a) = \mathbb{E}_{s'}[(\eta_n(s,a) - \max_{a'} Q_n(s',a'))_+^{k_*} - Z_{n,1}(s,a)],$$

$$f_2(Z_n, \eta_n, Q_n)(s,a) = \mathbb{E}_{s'_n}[(\eta_n(s,a) - \max_{a'} Q_n(s',a'))_+^{k_*-1} - Z_{n,2}(s,a)],$$

$$M_{n,1}^Z(s,a) = (\eta_n(s,a) - \max_{a'} Q_n(s',a'))_+^{k_*} - Z_{n,1}(s,a) - f_1(Z_n, \eta_n, Q_n)(s,a),$$

$$M_{n,2}^Z(s,a) = (\eta_n(s,a) - \max_{a'} Q_n(s',a'))_+^{k_*-1} - Z_{n,2}(s,a) - f_2(Z_n, \eta_n, Q_n)(s,a).$$

In the update of η_n (Equation 27), g and ϵ_n^η are defined as

$$g(Z_n, \eta_n, Q_n)(s,a) = -c_k(\rho)\mathbb{E}[(\eta_n(s,a) - \max_{a' \in \mathcal{A}} Q_n(s',a'))_+^{k_*}]^{\frac{1}{k_*}-1} \cdot \mathbb{E}[(\eta_n(s,a) - \max_{a' \in \mathcal{A}} Q_n(s',a'))_+^{k_*-1}] + 1,$$

$$\epsilon_n^\eta(s,a) = -c_k(\rho) Z_{n,1}^{\frac{1}{k_*}-1}(s,a) \cdot Z_{n,2}(s,a) + 1 - g(Z_n, \eta_n, Q_n)(s,a).$$

Finally in the update of Q_n (Equation 15), h and ϵ_n^Q are defined as

$$h(Z_n, \eta_n, Q_n)(s,a) = r(s,a) - \gamma(c_k(\rho)(\mathbb{E}_P[(\eta_n(s,a) - \max_{a' \in \mathcal{A}} Q_n(s',a'))_+^{k_*}])^{\frac{1}{k_*}} - \eta_n(s,a)),$$

$$\epsilon_n^Q(s,a) = r(s,a) - \gamma(c_k(\rho) Z_{n,1}^{\frac{1}{k_*}}(s,a) - \eta_n(s,a)) - h(Z_n, \eta_n, Q_n)(s,a).$$

The algorithm 1 approximates the dynamic described by the system of f, g and h through samples along a single trajectory, with the resulting approximation error manifesting as martingale noise M_n^Z conditioned on some filtration $\mathcal{F}_n$ and the error terms ϵ_n^η and ϵ_n^Q.

To analyze the dynamic of algorithm 1, we first obtain the continuous dynamic of f, g, and h using ordinary differential equations (ODEs) analysis. The second step is to analyze the stochastic nature of the noise term M_n^Z and the error terms ϵ_n^η and ϵ_n^Q, to ensure that they are negligible compared to the main trend of f, g, and h, which is achieved by the following stepsizes,

Assumption C.1. The stepsizes $\zeta_i(n), i = 1, 2, 3$ satisfy

$$\sum_n \zeta_i(n) = \infty, \quad \sum_n \zeta_i^2(n) < \infty, \quad \zeta_1(n) = o(\zeta_2(n)), \quad \zeta_2(n) = o(\zeta_3(n)).$$

These stepsize schedules satisfy the standard conditions for stochastic approximation algorithms, ensuring that **(1).** the key quantities in gradient estimator Z_n update on the fastest timescale, **(2).** the dual variable for the DR problem, η_n, update on the intermediate timescale; and **(3).** the Q table updates on the slowest timescale. Examples of such stepsize are $\zeta_1(n) = \frac{1}{1+n^{0.6}}, \zeta_2(n) = \frac{1}{1+n^{0.8}}$ and $\zeta_3(n) = \frac{1}{1+n}$. Notably, the first two conditions in Assumption C.1 ensure the martingale noise is negligible. The different stepsizes for the three loops specificed by the third and fourth conditions ensures that $Z_{n,1}$ and $Z_{n,2}$ are sufficiently estimated with respect to the η_n and Q_n, and these outer two loops are free from bias or noise in the stochastic approximation sense.

Under Assumption C.1, when analyzing the behavior of the Z_n, the η_n and the Q_n can be viewed as quasi-static. To study the behavior of the fastest loop, we analyze the following ODEs:

$$\dot{Z}(t) = f(Z(t), \beta(t), Q(t)), \quad \dot{\eta}(t) = 0, \quad \dot{Q}(t) = 0, \tag{16}$$

and prove that ODEs (16) a.s. converge to $\lambda_1^{''}(\eta, Q)$ for proper η and Q and some mapping $\lambda_1^{''}$. Similarly, Q_n can be viewed as fixed when analyzing the behavior of η_n, and the corresponding ODEs to understand its behavior are

$$\dot{\eta}(t) = g(\lambda_1''(\eta(t), Q(t)), \eta(t), Q(t)), \quad \dot{Q}(t) = 0. \tag{17}$$

By exploiting the dual form of the distributionally robust optimization problem, we can prove these ODEs converge to the set $\{\lambda_1'(Q), \lambda_2'(Q), Q | Q \in V\}$ for some mapping λ_1' and λ_2' with V is the set containing all the mapping from $\mathcal{S}$ to $\mathbb{R}$. Lastly, we examine the slowest timescale ODE given by

$$\dot{Q}(t) = h(\lambda_1'(Q(t)), \lambda_2'(Q(t)), Q(t)), \tag{18}$$

and employ our analysis to establish the almost sure convergence of Algorithm 1 to the globally optimal pair $(Z_1^\star, Z_2^\star, \eta^\star, Q^\star)$.

Lemma C.2 (Discrete Gronwall inequality)**.** *Let $\{x_n, n \geq 0\}$ (resp. $\{a_n, n \geq 0\}$) be nonnegative (resp. positive) sequences and $C, L \geq 0$ scalars such that for all n,*

$$x_{n+1} \leq C + L\left(\sum_{m=0}^{n} a_m x_m\right).$$

Then for $T_n = \sum_{m=0}^{n} a_m$,

$$x_{n+1} \leq Ce^{LT_n}.$$

Lemma C.3 (Gronwall inequality)**.** *For continuous $u(\cdot), v(\cdot) \geq 0$ and scalars $C, K, T \geq 0$*

$$u(t) \leq C + K\int_0^t u(s)v(s)ds, \quad \forall t \in [0, T],$$

implies

$$u(t) \leq Ce^{K\int_0^T v(s)ds}, \quad \forall t \in [0, T].$$

C.2 Stability Criterion

Consider the stochastic approximation scheme $z_n \in \mathbb{R}^N$ given by

$$z_{n+1} = z_n + a_n\left[g\left(z_n\right) + M_{n+1}\right],$$

with the following assumptions:

Assumption C.4. $g : \mathbb{R}^N \to \mathbb{R}^N$ is Lipschitz.

Assumption C.5. The sequence $\{a_n\} \subset \mathbb{R}$ satisfies $\sum_n a_n = \infty, \sum_n a_n^2 < \infty$.

Assumption C.6. $\{M_n\}$ is a martingale difference sequence with respect to the filtration $\mathcal{F}_n = \sigma\left(z_m, M_m, m \leq n\right)$, there exists $K > 0$ such that $E\left[\|M_{n+1}\|^2 \mid \mathcal{F}_n\right] \leq K(1 + \|z_n\|^2)$ a.s..

Assumption C.7. The functions $g_d(z) = g(dz)/d, d \geq 1$ satisfy $g_d(z) \to g_\infty(z)$ as $d \to \infty$ uniformly on compacts for some continuous function $g_\infty : \mathbb{R}^N \to \mathbb{R}^N$. In addition, the ODE

$$\dot{z}(t) = g_\infty(z(t))$$

has the origin as its globally asymptotically stable equilibrium.

We then have

Lemma C.8. *Under Assumptions C.4 to C.6, we have $\sup_n \|z_n\| < \infty$ a.s.*

See Section 2.2 and 3.2 in Borkar (2009) for the proof. As the stability proofs in Section 3.2 of Borkar (2009) are path-wise, we can apply this result to analyze multiple timescales dynamic.

C.3 Three Timescales Convergence Criterion

Consider the scheme

$$x_{n+1} = x_n + a_n\left[f\left(x_n, y_n, z_n\right) + M_{n+1}^{(1)}\right] \tag{19}$$

$$y_{n+1} = y_n + b_n\left[g\left(x_n, y_n, z_n\right) + M_{n+1}^{(2)}\right] \tag{20}$$

$$z_{n+1} = z_n + c_c\left[h\left(x_n, y_n, z_n\right) + M_{n+1}^{(3)}\right] \tag{21}$$

where $f : \mathbb{R}^{d+k+p} \to \mathbb{R}^d$, $g : \mathbb{R}^{d+k+p} \to \mathbb{R}^k$, $h : \mathbb{R}^{d+k+p} \to \mathbb{R}^p$, $\{M_n^{(i)}\}, i = 1, 2, 3$ are martingale difference sequences with respect to the σ-fields $\mathcal{F}_n = \sigma\left(x_m, y_m, M_m^{(1)}, M_m^{(2)}, M_m^{(3)}; m \leq n\right)$, and the a_n, b_n, c_n form decreasing stepsize sequences.

It is instructive to compare the stochastic update algorithms from Equations 19 to 21 with the following o.d.e.,

$$\dot{x}(t) = \frac{1}{a}f(x(t), y(t), z(t)),$$
$$\dot{y}(t) = \frac{1}{b}g(x(t), y(t), z(t)),$$
$$\dot{z}(t) = \frac{1}{c}h(x(t), y(t), z(t)),$$

in the limit that $a, b, c \to 0$ and $a = o(b)$, $c = o(b)$.

We impose the following assumptions.

Assumption C.9. f and g is L-Lipschitz map for some $0 < L < \infty$ and h is bounded.

Assumption C.10.

$$\sum_n a_n = \sum_n b_n = \sum_n c_n = \infty, \sum_n (a_n^2 + b_n^2 + c_n^2) < \infty, \text{ and } b_n = o(a_n), c_n = o(b_n).$$

Assumption C.11. For $i = 1, 2, 3$ and $n \in \mathbb{N}^+$, $\{M_n^{(i)}\}$ is a martingale differeence sequence with respect to the increasing family of σ-fields $\mathcal{F}_n$. Furthermore, there exists some $K > 0$, such that for $i = 1, 2, 3$ and $n \in \mathbb{N}^+$,

$$\mathbb{E}[\|M_{n+1}^{(i)}\|^2 | \mathcal{F}_n] \leq K(1 + \|x_n\|^2 + \|y_n\|^2 + \|z_n\|^2).$$

Assumption C.12. $\sup_n(\|x_n\| + \|y_n\| + \|z_n\|) < \infty$, a.s..

Assumption C.13. For each $y \in \mathbb{R}^k$ and $z \in \mathbb{R}^p$, $\dot{x}(t) = f(x(t), y, z)$ has a globally asymptotically stable equilibrium $\lambda_1(y, z)$, where $\lambda_1 : \mathcal{R}^{k+p} \to \mathcal{R}^d$ is a L-Lipschitz map for some $L > 0$.

Assumption C.14. For each $z \in \mathbb{R}^p$, $\dot{y}(t) = g(\lambda_1(y(t), z), y(t), z)$ has a globally asymptotically stable equilibrium $\lambda_2(z)$, where $\lambda_2 : \mathcal{R}^p \to \mathcal{R}^k$ is a L-Lipschitz map for some $L > 0$.

Assumption C.15. $\dot{z}(t) = h(\lambda_1(z(t)), \lambda_2(z(t)), z(t))$ has a globally asymptotically stable equilibrium $z^\star$.

Assumptions C.9, C.10, C.11 and C.12 are necessary for the a.s. convergence for each timescale itself. Moreover, Assumption C.12 itself requires Assumptions like C.9, C.10, C.11, with an extra assumption like Assumption C.6. Instead, we need to prove the boundedness for each timescale, thus the three timescales version is as follow

Assumption C.16. The ODE

$$\begin{aligned}
\dot{z}(t) &= f_\infty(x(t), y, z) \\
\dot{y}(t) &= g_\infty(\lambda_1(y(t), z), y(t), z) \\
\dot{z}(t) &= h_\infty(\lambda_1(z(t)), \lambda_2(z(t)), z(t))
\end{aligned}$$

all have the origin as their globally asymptotically stable equilibrium for each $y \in \mathcal{R}^k$ and $z \in \mathcal{R}^p$, where

$$f_\infty = \lim_{d\to\infty} \frac{f(dx)}{d}, \quad g_\infty = \lim_{d\to\infty} \frac{g(dx)}{d}, \text{ and } h_\infty = \lim_{d\to\infty} \frac{h(dx)}{d}.$$

We have the following results, which appears as a three timescales extension of Lemma 6.1 in Borkar (2009) and serves as a auxiliary lemma for the our a.s. convergence.

Lemma C.17. *Under the assumptions C.9, C.10, C.11 and C.12.* $(x_n, y_n, z_n) \to \{\lambda_1'(z), \lambda_2'(z), z : z \in \mathcal{R}^p\}$ *a.s..*

Proof. Rewrite equations 20 and 21 as

$$\begin{aligned}
y_{n+1} &= y_n + a_n \left[\epsilon_{1,n} + M_{n+1}^{(2)'}\right] \\
z_{n+1} &= z_n + a_n \left[\epsilon_{2,n} + M_{n+1}^{(3)'}\right],
\end{aligned}$$

where $\epsilon_{1,n} = \frac{b_n}{a_n} g(x_n, y_n, z_n)$, $\epsilon_{2,n} = \frac{c_n}{a_n} h(x_n, y_n, z_n)$, $M_{n+1}^{(2)'} = \frac{b_n}{a_n} M_{n+1}^{(2)}$, $M_{n+1}^{(3)'} = \frac{c_n}{a_n} M_{n+1}^{(3)}$. Note that $\epsilon_{1,n}, \epsilon_{2,n} \to 0$ as $n \to \infty$. Consider them as the special case in the third extension in Section 2.2 in Borkar (2009) and then we can conclude that (x_n, y_n, z_n) converges to the internally chain transitive invariant sets of the o.d.e.,

$$\begin{aligned}
\dot{x}(t) &= h(x(t), y(t), z(t)) \\
\dot{y}(t) &= 0 \\
\dot{z}(t) &= 0,
\end{aligned}$$

which implies that $(x_n, y_n, z_n) \to \{\lambda_1'(y, z), y, z : y \in \mathcal{R}^k, z \in \mathcal{R}^p\}$.

Rewrite Equation 21 again as

$$z_{n+1} = z_n + b_n \left[\epsilon_{2,n}' + M_{n+1}^{(3)''}\right],$$

where $\epsilon_{2,n}' = \frac{c_n}{b_n} h(x_n, y_n, z_n)$ and $M_{n+1}^{(3)''} = \frac{c_n}{b_n} M_{n+1}^{(3)}$. We use the same extension again and can conclude that (x_n, y_n, z_n) converges to the internally chain transitive invariant sets of the o.d.e.,

$$\begin{aligned}
\dot{y}(t) &= g(\lambda_1'(y(t)), y(t), z(t)) \\
\dot{z}(t) &= 0.
\end{aligned}$$

Thus $(x_n, y_n, z_n) \to \{\lambda_1(y), \lambda_2(z), z : z \in \mathcal{R}^p\}$. □

Theorem C.18. *Under the assumptions C.9 to C.16,* $(x_n, y_n, z_n) \to (\lambda_1(z^*), \lambda_2(z^*), z^*)$.

Proof. Let $t(0) = 0$ and $t(n) = \sum_{i=0}^{n-1} c_i$ for $n \geq 1$. Define the piecewise linear continuous function $\tilde{z}(t), t \geq 0$ where $\tilde{z}(t(n)) = z_n$ and $\tilde{z}(t) = \frac{t(n+1)-t}{t(n+1)-t(n)} z_{n+1} + \frac{t-t(n)}{t(n+1)-t(n)} z_n$ for $t \in [t(n), t(n+1)]$ with any $n \in N$. Let $\psi_n = \sum_{i=0}^{n-1} c_i M_{i+1}^{(3)}, n \in \mathbb{N}^+$. For any $t \geq 0$, denote $[t] = \max\{s(n) : s(n) \leq t\}$. Then for $n, m \geq 0$, we have

$$\begin{aligned}
\tilde{z}(t(n+m)) &= \tilde{z}(t(n)) + \sum_{k=1}^{m-1} c_{n+k} h(x_{n+k}, y_{n+k}, z_{n+k}) + (\psi_{m+n+1} - \psi_n) \\
&= \tilde{z}(t(n)) + \int_{t(n)}^{t(n+m)} h(\lambda_1(z(s)), \lambda_2(z(s)), z(s)) ds \\
&+ \int_{t(n)}^{t(n+m)} (h(\lambda_1(z([s])), \lambda_2(z([s])), z([s])) - h(\lambda_1(z(s)), \lambda_2(z(s)), z(s))) ds \\
&+ \sum_{k=0}^{m-1} c_{n+k} (h(x_{n+k}, y_{n+k}, z_{n+k}) - h(\lambda_1(z_{n+k}), \lambda_2(z_{n+k}), z_{n+k})) \\
&+ (\psi_{n+m+1} - \psi_n). \qquad (22)
\end{aligned}$$

We further define $z^{t(n)}(t)$ as the trajectory of $\dot{z}(t) = g(\lambda_1(z(t)), \lambda_2(z(t)), z(t))$ with $z^{t(n)}(t(n)) = \tilde{z}(t(n))$.

$$z^{t(n)}(t(n+m)) = \tilde{z}(t(n)) + \int_{t(n)}^{t(n+m)} h(\lambda_1(z^{t(n)}(s)), \lambda_2(z^{t(n)}(s)), z^{t(n)}(s)) ds. \qquad (23)$$

Taking the difference between Equation 22 and the Equation 23 we have

$$\begin{aligned}
&|\tilde{z}(t(n+m)) - z^{t(n)}(t(n+m))| \\
&= \underbrace{\sum_{k=0}^{m-1} c_{n+k} (h(\lambda_1(\tilde{z}(t+k)), \lambda_2(\tilde{z}(t+k)), \tilde{z}(t+k)) - h(\lambda_1(z(t(n+k))), \lambda_2(z(t(n+k))), z(t(n+k))))}_{} \\
&+ \underbrace{|\int_{t(n)}^{t(n+m)} (h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])) - h(\lambda_1(z(s)), \lambda_2(z(s)), z(s))) ds|}_{\text{I}} \\
&+ \underbrace{|\sum_{k=1}^{m-1} c_{n+k} (h(x_{n+k}, y_{n+k}, z_{n+k}) - h(\lambda_1(z_{n+k}), \lambda_2(z_{n+k}), z_{n+k}))|}_{\text{II}} \\
&+ \underbrace{|\psi_{n+m+1} - \psi_n|}_{\text{III}}.
\end{aligned}$$

We analyze the I term. For notation simplicity we ignore the supsript $t(n)$.

$$\begin{aligned}
&|h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])) - h(\lambda_1(z(t)), \lambda_2(z(t)), z(t))| \\
&= |(h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])) - h(\lambda_1(z([t])), \lambda_2(z([t])), z(t)))| \\
&\quad + |(h(\lambda_1(z([t])), \lambda_2(z([t])), z(t)) - h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])))| \\
&= |(h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])) - h(\lambda_1(z([t])), \lambda_2(z(t)), z(t)))| \\
&\quad + |h(\lambda_1(z([t])), \lambda_2(z([t])), z(t)) - h(\lambda_1(z([t])), \lambda_2(z([t])), z(t))| \\
&\quad + |(h(\lambda_1(z([t])), \lambda_2(z([t])), z(t)) - h(\lambda_1(z([t])), \lambda_2(z([t])), z([t])))|. \qquad (24)
\end{aligned}$$

By the Lipschitzness of the h we have

$$\|h(x) - h(0)\| \leq L\|x\|,$$

which implies

$$\|h(x)\| \leq \|h(0)\| + L\|x\|.$$

$$\begin{aligned}
\|z^{t(n)}(t)\| &\leq \|\tilde{z}(s)\| + \int_s^t \|h(z^{t(n)}(s))\| ds \\
&\leq \|\tilde{z}(s)\| + \int_s^t (\|h(0)\| + L\|z^{t(n)}(s)\|) ds \\
&\leq (\|\tilde{z}(s)\| + \|h(0)\| T) + L \int_s^t \|z^{t(n)}(s)\| ds.
\end{aligned}$$

By Gronwall's inequality (Lemma C.3), we have

$$\|z^{t(n)}(t)\| \leq (C + \|h(0)\|T)e^{LT}, \quad \forall t \in [t(n), t(n+m)].$$

Thus for all $t \in [t(n), t(n+m)]$, we have

$$\|h(\lambda_1(z^{t(n)}(t)), \lambda_2(z^{t(n)}(t)), z^{t(n)}(t))\| \leq C_T \coloneqq \|h(0)\| + L(C + \|h(0)\|T)e^{LT} < \infty, a.s..$$

For any $k \in [m-1]$ and $t \in [t(n+k), t(n+k+1)]$,

$$\begin{aligned}\|z^{t(n)}(t) - z^{t(n)}(t(n+k))\| &\leq \|\int_{t(n+k)}^{t} h(\lambda_1(z^{t(n)}(s)), \lambda_2(z^{t(n)}(s)), z^{t(n)}(s))ds\| \\ &\leq C_T(t - t(n+k)) \\ &\leq C_T a(n+k),\end{aligned}$$

where the last inequality is from the construction of $\{t(n) : n \in \mathbb{N}^+\}$. Finally we can conclude

$$\begin{aligned}&\|\int_{t(n)}^{t(n+m)} (h(\lambda_1(z([s])), \lambda_2(z([s])), z(s)) - h(\lambda_1(z([s])), \lambda_2(z([s])), z([s])))ds\| \\ &\leq \int_{t(n)}^{t(n+m)} L\|z(s) - z([s])\|ds \\ &= L\sum_{k=0}^{m-1} \int_{t(n+k)}^{t(n+k-1)} \|z(s) - z(t(n+k))\|ds \\ &\leq C_T L \sum_{k=0}^{m-1} c_{n+k}^2 \\ &\leq C_T L \sum_{k=0}^{\infty} c_{n+k}^2 \to 0, a.s..\end{aligned}$$

For the III term, it converges to zero from the martingale convergence property.

Subtracting equation 22 from 23 and take norms, we have

$$\begin{aligned}&\|\tilde{z}(t(n+m)) - z^{t(n)}(t(n+m))\| \\ &\leq L\sum_{i=0}^{m-1} c_{n+i}\|\tilde{z}(t(n+i)) - z^{t(n)}(t(n+i))\| \\ &\quad + C_T L \sum_{k\geq 0} c_{n+k}^2 + \sup_{k\geq 0}\|\delta_{n,n+k}\|, a.s..\end{aligned}$$

Define $K_{T,n} = C_T L \sum_{k\geq 0} c_{n+k}^2 + \sup_{k\geq 0}\|\delta_{n,n+k}\|$. Note that $K_{T,n} \to 0$ a.s. $n \to \infty$. Let $u_i = \|\tilde{x}(t(n+i)) - x^{t(n)}(t(n+i))\|$. Thus, above inequality becomes

$$u_m \leq K_{T,n} + L\sum_{i=0}^{m-1} c_{n+i}u_i.$$

Thus the above inequality becomes

$$z(t(n+m)) \leq K_{T,n} + L\sum_{k=0}^{m-1} c_k z(t(n+k)).$$

Note that $u_0 = 0$ and $\sum_{i=0}^{m-1} b_i \leq T$, then using the discrete Gronwall lemma (Lemma C.2) we have

$$\sup_{0\leq i\leq m} u_i \leq K_{T,n}e^{LT}.$$

Following the similar logic as in Lemma 1 in Borkar (2009), we can extend the above result to the case $\|\tilde{z}(t) - z^{t(n)}(t)\| \to 0$ where $t \in [0, T]$.

Then using the proof of Theorem 2 of Chapter 2 in Borkar (2009), we get $z_n \to z^*$ a.s. and thus by Lemma C.17 the proof can be concluded. □

D CONVERGENCE OF THE DR Q-LEARNING ALGORITHM

Before we start the proof of the DR Q-learning algorithm, we first introduce the following lemma.

Lemma D.1. *Denote* $\eta^* = \arg\max_\eta \sigma_k(X,\eta) = -c_k(\rho)\mathbb{E}_P[(\eta - X)_+^{k_*}]^{\frac{1}{k_*}} + \eta$. *Given that* $X(\omega) \in [0, M]$, *then we have* $\eta* \in [0, \frac{c_k(\rho)}{c_k(\rho)-1}M]$.

Proof. Note that for $\eta = \min_\omega X(\omega)$, $-c_k(\rho)\mathbb{E}_P[(\eta - X)_+^{k_*}]^{\frac{1}{k_*}} + \eta = \min_\omega X(\omega) \geq 0$. Also we know that when $\eta \geq \frac{c_k(\rho)}{c_k(\rho)-1}M$,

$$\begin{aligned}
&- c_k(\rho)\mathbb{E}_P[(\eta - X)_+^{k_*}]^{\frac{1}{k_*}} + \eta \\
\leq &- c_k(\rho)\mathbb{E}_P[(\eta - M)_+^{k_*}]^{\frac{1}{k_*}} + \eta \\
= &- c_k(\rho)(\eta - M) + \eta \\
\leq &0.
\end{aligned}$$

Then we can conclude that $\eta^* \leq \frac{c_k(\rho)}{c_k(\rho)-1}M$. Moreover, as $X(\omega) \geq 0$, we know $\sigma_k(X, 0) = 0$, which concludes that $\eta* \in [0, \frac{c_k(\rho)}{c_k(\rho)-1}M]$. □

Note that $Q_n \in [0, \frac{1}{1-\gamma}]$ when reward is bounded by $[0, 1]$. Thus $M = \frac{1}{1-\gamma}$ in our case and then we denote $\overline{\eta} = \frac{c_k(\rho)}{c_k(\rho)-1}M$. Now we are ready to prove the convergence of the DR Q-learning algorithm. For theoretical analysis, we consider the clipping version of our DR Q-learning algorithm.

Proof of Theorem 3.3. We define the filtration generated by the historical trajectory,

$$\mathcal{F}_n = \sigma(\{(s_t, a_t, s_t', r_t)\}_{t\in[n-1]}, s_n, a_n).$$

In the following analysis, we fix for a $(s, a) \in \mathcal{S} \times \mathcal{A}$ but ignore the (s, a) dependence for notation simplicity. Following the roadmap in Section 3.4, we rewrite the algorithm as

$$Z_{n+1,1} = Z_{n,1} + \zeta_1(n)[f_1(Z_{n,1}, Z_{n,2}, \eta_n, Q_n) + M_{n+1}^{(1)}], \tag{25}$$

$$Z_{n+1,2} = Z_{n,2} + \zeta_1(n)[f_2(Z_{n,1}, Z_{n,2}, \eta_n, Q_n) + M_{n+1}^{(2)}], \tag{26}$$

$$\eta_{n+1} = \Gamma_\eta\left[\eta_n + \zeta_2(n) f_3(Z_{n,1}, Z_{n,2}, \eta_n, Q_n)\right], \tag{27}$$

$$Q_{n+1} = \Gamma_Q[Q_n + \zeta_3(n)[f_4(Z_{n,1}, Z_{n,2}, \eta_n, Q_n)]]. \tag{28}$$

Here for theoretical analysis, we add a clipping operator $\Gamma_\eta(x) = \min(\max(x, 0), \overline{\eta})$ and $\Gamma_Q(x) = \min(\max(x, 0), M)$ compared with the algorithm presented in the main text.

We first proceed by first identifying the terms in Equation 25 and 26 and studying the corresponding ODEs

$$\begin{aligned}
\dot{Q}(t) &= 0, \\
\dot{\eta}(t) &= 0, \\
\dot{Z}_1(t) &= f_1(Z_1(t), Z_2(t), \eta(t), Q(t)). \\
\dot{Z}_2(t) &= f_2(Z_1(t), Z_2(t), \eta(t), Q(t)).
\end{aligned}$$

As f_1 and f_2 is in fact irrelavant to the Z_2 and Z_1, we analyze their equilibria seperately. For notation convenience, we denote $y_n(s) = \max_{a'\in\mathcal{A}} Q_n(s, a')$.

For ODE 25 and each $\eta_n \in \mathbb{R}, Q_n \in \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, it is easy to know there exists a unique global asymtotically stable equilibrium $Z_{n,1}^\star = \lambda_1(\eta_n, y_n) = \mathbb{E}[(\eta_n - y_n)_+^{k_*}]$. Similarly, For ODE 26 and each $\eta_n \in \mathbb{R}, Q_n \in \mathcal{S}\times\mathcal{A} \to \mathbb{R}$, there exists a unique global asymtotically stable equilibrium $Z_{n,2}^\star = \lambda_2(\eta, y) = \mathbb{E}[(\eta_n - y_n)_+^{k_*-1}]$.

Second, $M_{n+1}^{(1)} = (\eta_n - y_n)_+^{y_*} - \mathbb{E}[(\eta_n - y_n)_+^{y_*}]$ and $M_{n+1}^{(2)} = (\eta_n - y_n)_+^{y_*-1} - \mathbb{E}[(\eta_n - y_n)_+^{y_*-1}]$. Note that for any $(s, a) \in \mathcal{S} \times \mathcal{A}$, $\eta_n(s, a) \leq \overline{\eta}$, $y_n(s') \leq M$ and $M \leq \overline{\eta}$. Thus $|(\eta_n(s, a) - y_n(s'))_+^{y_*}| \leq \overline{\eta}^{y_*}$, which leads to $|M_{n+1}^{(1)}(s, a)| = |(\eta_n(s, a) - y_n(s'))_+^{y_*} - \mathbb{E}[(\eta_n(s, a) - y_n(s'))_+^{y_*}]| \leq \overline{\eta}^{k_*}$.

Since $\|y_n\|_\infty \leq \|Q_n\|_\infty$ and $(x - y)_+^2 \leq x^2 + y^2$ for any x, y, we have,

$$\begin{aligned}
&\mathbb{E}[\|M_{n+1}^{(1)}\|^2|\mathcal{F}_n] \\
&= \mathbb{E}[\|(\eta_n - y_n)_+^{y_*} - \mathbb{E}[(\eta_n - y_n)_+^{y_*}]\|^2|\mathcal{F}_n] \\
&\leq K_1(1 + \|Z_{n,1}\|^2 + \|Z_{n,2}\|^2 + \|Q_n\|^2 + \|\eta_n\|^2),
\end{aligned}$$

where $K_1 = S\overline{\eta}^{2k_*}$. Similarly, we can conclude that $\mathbb{E}[\|M_{n+1}^{(2)}\|^2|\mathcal{F}_n] \leq K_2(1 + \|Z_{n,1}\|^2 + \|Z_{n,2}\|^2 + \|Q_n\|^2 + \|\eta_n\|^2)$ for some $K_2 = S\overline{\eta}^{2(k_*-1)}$.

Next we analyze the second loop.

$$\begin{aligned}\dot{Q}(t) &= 0,\\ \dot{\eta}(t) &= \Gamma_\eta[f_3(\lambda_1(\eta(t), Q(t)), \lambda_2(\eta(t), Q(t)), \eta(t), Q(t))],\end{aligned}$$

where

$$f_3(\lambda_1(\eta, Q), \lambda_2(\eta, Q), \eta, Q) = -c_k(\rho)\lambda_1(\eta, Q)^{\frac{1}{k_*}-1}\lambda_2(\eta, Q) + 1.$$

The global convergence point is $\eta^*(t) = \arg\max_{\eta\in[0,\overline{\eta}]}\{\sigma_k(Q, \eta)\} = \arg\max_{\eta\in\mathbb{R}}\{\sigma_k(Q, \eta)\}$.

Finally we arrive to the outer loop, i.e.,

$$\dot{Q}(t) = \Gamma_Q[f_4(\lambda_1(Q(t)), \lambda_2(Q(t)), \lambda_3(Q(t)), Q(t))].$$

By using the dual form of Cressie-Read Divergence (Lemma 3.1), we know that this is equivilant to

$$\dot{Q}(t) = r + \gamma \inf_{P\in\mathcal{P}} \mathbb{E}_P[\max_{a'} Q(s', a')] - Q(t),$$

for ambiguity set using Cressie-Read of f divergence.

Denote $H(t) = r + \gamma \inf_{P\in\mathcal{P}} \mathbb{E}_P[\max_{a'} Q(s', a')]$ and thus we can rewrite the above ODE as

$$\dot{Q}(t) = H(t) - Q(t).$$

Following , we consider its infity version, i.e., $H^\infty(t) = \lim_{c\to\infty} H(ct)/c$.

$$\dot{Q}(t) = \gamma \inf_{P\in\mathcal{P}} \mathbb{E}_P[\max_{a'} Q(s', a')] - Q(t).$$

This is a contraction by Theorem 3.2 in Iyengar (2005). By the proof in Section 3.2 in Borkar & Meyn (2000), we know the contraction can lead to the global unique equilibrium point in the ode. Thus we finish verifying all the conditions in Section C.3, which can lead to the desired result. □