Regularized Q-learning through Robust Averaging

Peter Schmitt-Förster [1] **Tobias Sutter** [1]

Abstract

We propose a new Q-learning variant, called *2RA Q-learning*, that addresses some weaknesses of existing Q-learning methods in a principled manner. One such weakness is an underlying estimation bias which cannot be controlled and often results in poor performance. We propose a distributionally robust estimator for the maximum expected value term, which allows us to precisely control the level of estimation bias introduced. The distributionally robust estimator admits a closed-form solution such that the proposed algorithm has a computational cost per iteration comparable to Watkins' Q-learning. For the tabular case, we show that 2RA Q-learning converges to the optimal policy and analyze its asymptotic mean-squared error. Lastly, we conduct numerical experiments for various settings, which corroborate our theoretical findings and indicate that 2RA Q-learning often performs better than existing methods.

1. Introduction

The optimal policy of a Markov Decision Process (MDP) is characterized via the dynamic programming equations introduced by Bellman (1957). While these dynamic programming equations critically depend on the underlying model, model-free reinforcement learning (RL) aims to learn these equations by interacting with the environment without any knowledge of the underlying model (Bertsekas & Tsitsiklis, 1996; Sutton & Barto, 2018; Meyn, 2022). There are two fundamentally different notions of interacting with the unknown environment. The first one is referred to as the synchronous setting, which assumes sample access to a generative model (or simulator), where the estimates are updated simultaneously across all state-action pairs in every iteration step. The second concept concerns an asynchronous setting, where one only has access to a single trajectory of data generated under some fixed policy. A learning algorithm then updates its estimate of a single state-action pair using one state transition from the sampled trajectory in every step. In this paper, we focus on the asynchronous setting, which is the considerably more challenging task than the synchronous setting due to the Markovian nature of its sampling process.

One of the most popular RL algorithms is Q-learning (Watkins, 1989; Watkins & Dayan, 1992), which iteratively learns the value function and hence the corresponding optimal policy of an MDP with unknown transition kernel. When designing an RL algorithm, there are various desirable properties such an algorithm should have, including (i) convergence to an optimal policy, (ii) efficient computation, and (iii) "good" performance of the learned policy after finitely many iterations. Watkins' Q-learning is known to converge to the optimal policy under relatively mild conditions (Tsitsiklis, 1994) and a finite-time analysis is also available (Even-Dar & Mansour, 2001; Beck & Srikant, 2012; Qu & Wierman, 2020). Moreover, its simple update rule requires only one single maximization over the action space per step. The simplicity of Watkins' Q-learning, however, comes at the cost of introducing an overestimation bias (Thrun & Schwartz, 1993; van Hasselt, 2010), which can severely impede the quality of the learned policy (Szita & Lörincz, 2008; Strehl et al., 2009; Thrun & Schwartz, 1993; van Hasselt, 2010). It has been experimentally demonstrated that both, overestimation and underestimation bias, may improve learning performance, depending on the environment at hand (see Sutton & Barto (2018, Chapter 6.7) and Lan et al. (2020) for a detailed explanation). Therefore, deriving a Q-learning method equipped with the possibility to precisely control the (over- and under-)estimation bias is desirable.

Related Work. In the last decade, several Q-learning variants have been proposed to improve the weakness of Watkins' Q-learning while aiming to admit the desirable properties (i)-(iii). We discuss approaches that are most relevant to our work. Double Q-learning (van Hasselt, 2010) mitigates the overestimation bias of Watkins' Q-learning by introducing a double estimator for the maximum expected value term. While Double Q-learning is known to converge

[1]Department of Computer and Information Science, University of Konstanz, Germany. Correspondence to: Peter Schmitt-Förster <peter.schmitt-foerster@uni-konstanz.de>.

Proceedings of the 41^{st} International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

to the optimal policy and has a similar computational cost per iteration to Q-learning, it, unfortunately, introduces an underestimation bias which, depending on the environment considered, can be equally undesirable as the overestimation bias of Watkins' Q-learning (Lan et al., 2020).

Maxmin Q-learning (Lan et al., 2020) works with N state-action estimates, where N denotes a parameter, and chooses the smallest estimate to select the maximum action. Maxmin Q-learning allows to control the estimation bias via the parameter N. However, it generally requires a large value of N to remove the overestimation bias. Conceptually, Maxmin Q-learning is related to our approach, where we select a maximum action based on the average current Q-function and then consider a worst-case ball around this average value. Similarly, REDQ (Chen et al., 2021) updates N Q-functions based on a max-action, min-function step over a sampled subset of size M out of the N Q-functions and allows to control the over- and underestimation bias. It further incorporates multiple randomized update steps in each iteration which results in good sample efficiency. REDQ performs well in complicated non-tabular settings (including continuous state/action spaces). In the tabular setting, Maxmin Q-learning and REDQ are equipped with an asymptotic convergence proof. Averaged-DQN (Anschel et al., 2017) is a simple extension to the Deep Q-learning (DQN) algorithm (Mnih et al., 2015), based on averaging previously learned Q-value estimates, which leads to a more stable training procedure and typically results in an improved performance by reducing the approximation error variance in the target values. Averaged-DQN and other regularized variants of DQN, such as Munchausen reinforcement learning (Vieillard et al., 2020), use a deep learning architecture and are not equipped with any theoretical guarantees about convergence or quality of the learned policy (Mehta & Meyn, 2020).

In general, averaging in Q-learning is a well-known variance reduction method. A specific form of variance-reduced Q-learning is presented in Wainwright (2019), where it is shown that the presented algorithm has minimax optimal sample complexity. Regularized Q-learning (Lim et al., 2022) studies a modified Q-learning algorithm that converges when linear function approximation is used. It is shown that simply adding an appropriate regularization term ensures convergence of the algorithm in settings where the vanilla variant does not converge due to the linear function approximation used. A slightly different objective, when modifying Q-learning schemes, is to robustify them against environment shifts, i.e., settings where the environment, in which the policy is trained, is different from the environment in which the policy will be deployed. A popular approach is to consider a distributionally robust Q-learning model, where the resulting Q-function converges to the optimal Q-value that is robust against small shifts in the environment, see Liu et al. (2022) for KL-based ambiguity sets and for distributionally robust formulations using the Wasserstein distance Neufeld & Sester (2022). The recent paper from Liu et al. (2022) presents a distributionally robust Q-learning methodology, where the resulting Q-function converges to the optimal Q-value that is robust against small shifts in the environment. Mehta & Meyn (2020), Meyn (2022), Lu et al. (2022), and Lu & Meyn (2023) have introduced a new class of Q-learning algorithms called convex Q-learning which exploit a convex reformulation of the Bellman equation via the well-known linear programming approach to dynamic programming (Hernández-Lerma & Lasserre, 1996; 1999).

Contribution. In this paper, we introduce a new Q-learning variant called *Regularized Q-learning through Robust Averaging* (2RA), which combines regularization and averaging. The proposed method has two parameters, $\rho > 0$ quantifying the level of robustness/regularization introduced and $N \in \mathbb{N}$, which describes the number of state-action estimates used to form the empirical average. Centered around this new Q-learning variant, our main contributions can be summarized as follows:

- We present a tractable formulation of the proposed 2RA Q-learning where the computational cost per iteration is comparable to Watkins' Q-learning.
- For any choice of N and for any positive sequence of regularization parameters $\{\rho_n\}_{n\in\mathbb{N}}$ such that $\lim_{n\to\infty} \rho_n = 0$, we prove that the proposed 2RA Q-learning asymptotically converges to the true Q-function, see Theorem 3.1.
- We show how the choice of the two parameters ρ and N allow us to control the level of estimation bias in 2RA Q-learning, see Theorem 3.2, and show that as $N \to \infty$ our proposed estimation scheme becomes unbiased.
- We prove that under certain technical assumptions, the asymptotic mean-squared error of 2RA Q-learning is equal to the asymptotic mean-squared error of Watkins' Q-learning, provided that we choose the learning rate of our method N-times larger than that of Watkins' Q-learning, see Theorem 3.3. This theoretical insight allows practitioners to start with an initial guess of the learning rate for the proposed method that is N-times larger than that of standard Q-learning.
- We demonstrate that the theoretical properties of 2RA can be numerically reproduced in synthetic MDP settings. In more practical experiments from the OpenAI gym suite (Brockman et al., 2016) we show that, even when implementations require deviations from out theoretically required assumptions, 2RA Q-learning has good performance and mostly outperforms other Q-learning variants.

Detailed proofs are relegated to the Appendix A.

2. Problem Setting

Consider an MDP given by a six-tuple $(\mathcal{S}, \mathcal{A}, P, r, \gamma, s_0)$ comprising a finite state space $\mathcal{S} = \{1, \dots, S\}$, a finite action space $\mathcal{A} = \{1, \dots, A\}$, a transition kernel $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$, a reward-per-stage function $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, a discount factor $\gamma \in (0, 1)$, and a deterministic initial state $s_0 \in \mathcal{S}$. Note that $(\mathcal{S}, \mathcal{A}, P, r, \gamma, s_0)$ describes a controlled discrete-time stochastic system, where the state and the action applied at time t are denoted as random variables S_t and A_t, respectively. If the system is in state $s_t \in \mathcal{S}$ at time t and action $a_t \in \mathcal{A}$ is applied, then an immediate reward of $r(s_t, a_t)$ is incurred, and the system moves to state s_{t+1} at time $t + 1$ with probability $P(s_{t+1}|s_t, a_t)$. Thus, $P(\cdot|s_t, a_t)$ represents the distribution of S_{t+1} conditional on $S_t = s_t$ and $A_t = a_t$. It is often convenient to represent the transition kernel as a matrix $P \in \mathbb{R}^{SA \times S}$. Actions are usually chosen according to a policy that prescribes a random action at time t depending on the state history up to time t and the action history up to time $t - 1$. Throughout the paper, we restrict attention to stationary Markov policies, which are described by a stochastic kernel $\pi : \mathcal{S} \to \Delta(\mathcal{A})$, that is, $\pi(a_t|s_t)$ denotes the probability of choosing action a_t while being in state s_t. We denote by Π the space of all stationary Markov policies. Given a stationary policy π and an initial condition s_0, it is well-known that there exists a unique probability measure $\mathbb{P}^\pi_{s_0}$ defined on the canonical sample space $\Omega = (\mathcal{S} \times \mathcal{A})^\infty$ equipped with its power set σ-field $\mathcal{F} = 2^\Omega$, such that for all $t \in \mathbb{N}$ we have (see Hernández-Lerma & Lasserre (1996, Section 2.2) for further details)

$$\begin{aligned}
&\mathbb{P}^\pi_{s_0}(S_0 = s_0) = 1,\\
&\mathbb{P}^\pi_{s_0}(S_{t+1} = s_{t+1}|S_t = s_t, A_t = a_t) = P(s_{t+1}|s_t, a_t)\\
&\qquad \forall s_t, s_{t+1} \in \mathcal{S},\ a_t \in \mathcal{A},\\
&\mathbb{P}^\pi_{s_0}(A_t = a_t|S_t = s_t) = \pi(a_t|s_t) \quad \forall s_t \in \mathcal{S},\ a_t \in \mathcal{A}.
\end{aligned}$$

To keep the notation simple, in the following, we denote $\mathbb{P}^\pi_{s_0}$ by $\mathbb{P}$ and the corresponding expectation operator by $\mathbb{E}$. Then, the ultimate goal is to find an optimal policy $\pi^\star$ which leads to the largest expected infinite-horizon reward, i.e.,

$$\pi^\star \in \arg\max_{\pi \in \Pi} \sum_{t=0}^{\infty} \gamma^t \mathbb{E}[r(S_t, A_t)]. \tag{1}$$

An optimal (deterministic) policy can be alternatively obtained from the optimal Q-function as $\pi^\star(\cdot|s) = \delta_{a^\star}(\cdot)$, where $a^\star \in \arg\max_{a \in \mathcal{A}} Q^\star(s, a)$ and the optimal Q-function satisfies the Bellman equation (Bertsekas & Tsitsiklis, 1996), i.e., $\forall (s, a) \in \mathcal{S} \times \mathcal{A}$

$$Q^\star(s, a) = r(s, a) + \gamma \sum_{s' \in \mathcal{S}} P(s'|s, a) \max_{a' \in \mathcal{A}} Q^\star(s', a'). \tag{2}$$

Solving for the Q-function via (2) requires the knowledge of the underlying transition kernel P and reward function r, objects which in reinforcement learning problems generally are not known.

In this work, we focus on the so-called *asynchronous* RL setting, where the Q-function is learned from a single trajectory of data which we assume to be generated from a fixed behavioral policy leading to state-action pairs $\{(S_1, A_1), \dots, (S_n, A_n), \dots\}$. The standard asynchronous Q-learning algorithm (Weng et al., 2020) can then be expressed as

$$\begin{aligned}
Q_{n+1}(S_n, A_n) = Q_n(S_n, A_n) + \alpha_n^{\mathsf{QL}}\big(r(S_n, A_n)\\
+ \gamma \max_{a' \in \mathcal{A}} Q_n(S_{n+1}, a') - Q_n(S_n, A_n)\big),
\end{aligned} \tag{3}$$

where $\alpha_n^{\mathsf{QL}} \in (0, 1]$ is the learning rate. It has been shown (Tsitsiklis, 1994; Szepesvári, 1997; Qu & Wierman, 2020; Lee & He, 2020) that if each state is updated infinitely often and each action is tried an infinite number of times in each state, convergence to the optimal Q-function can be obtained. That is, for a learning rate satisfying $\sum_{n=0}^\infty \alpha_n^{\mathsf{QL}} = \infty$ and $\sum_{n=0}^\infty (\alpha_n^{\mathsf{QL}})^2 < \infty$, Q-learning converges $\mathbb{P}$-almost surely to an optimal solution of the Bellman equation (2), i.e., $\lim_{n\to\infty} Q_n = Q^\star$ $\mathbb{P}$-almost surely. To simplify notation, we introduce the state-action variables $X_n = (S_n, A_n)$ and define $\mathcal{X} = \mathcal{S} \times \mathcal{A}$. As the state space in practice is often large, the Q-functions are commonly approximated via fewer basis functions. When interpreting the Q-function as a vector on $\mathbb{R}^{|\mathcal{X}|}$, we use

$$\begin{aligned}
&Q^\star \approx \Phi^\top \theta^\star, \quad \theta^\star \in \mathbb{R}^d,\\
&\Phi = \big(\phi(s_1, a_1), \dots, \phi(s_{|\mathcal{S}|}, a_{|\mathcal{A}|})\big) \in \mathbb{R}^{d \times |\mathcal{X}|},
\end{aligned} \tag{4}$$

where with slight abuse of notation we denote by $\phi(s, a) \in \mathbb{R}^d$ the given feature vectors associated with the pairs $s = s_i$ and $a = a_i$ for $i \in \{1, \dots |\mathcal{X}|\}$. Clearly, by choosing $d = |\mathcal{X}|$ and the canonical feature vectors $\phi(s, a) = \sum_{i=1}^{|\mathcal{X}|} e_i \cdot 1_{\{(s,a)=(s_i,a_i)\}}$ for $i = 1, \dots, |\mathcal{X}|$, the approximation (4) is exact, which is referred to as the tabular setting. For our convergence results that only hold in the tabular setting, we will use this representation. In this linear function approximation formulation, the standard asynchronous Q-learning (3) can be expressed as so-called $Q(0)$-learning (Melo et al., 2008; Meyn, 2022), which is given as

$$\begin{aligned}
\theta_{n+1} = \theta_n + \alpha_n^{\mathsf{QL}}\big(b(X_n) - A_1(X_n)\theta_n\\
+ \mathcal{E}(X_n, S_{n+1}, \theta_n)\big),
\end{aligned} \tag{5}$$

where $b(X_n) = \phi(X_n) r(X_n)$, $A_1(X_n) = \phi(X_n)\phi(X_n)^\top$, $\mathcal{E}(X_n, S_{n+1}, \theta_n) = \gamma \phi(X_n) \max_{a' \in \mathcal{A}} \phi(S_{n+1}, a')^\top \theta_n$. It is well-known (van Hasselt, 2010), that the term $\mathcal{E}$ in the Q-learning (5) introduces an overestimation bias when compared to (2), since

$$\max_{a' \in \mathcal{A}} \mathbb{E}[\phi(s, a')^\top \theta_n] \le \mathbb{E}[\max_{a' \in \mathcal{A}} \phi(s, a')^\top \theta_n] \ \ \forall s \in \mathcal{S}, \tag{6}$$

where the expectation is with respect to the random variable θ_n defined according to (5) and the inequality is due to Jensen. A common method to mitigate this overestimation bias is to modify Q-learning to the so-called Double Q-learning (van Hasselt, 2010), which using the linear function approximation, can be expressed in the form (5), where

$$\theta_n = \begin{pmatrix} \theta_n^A \\ \theta_n^B \end{pmatrix}, \quad b(X_n) = \begin{pmatrix} \beta_n \phi(X_n) r(X_n) \\ (1-\beta_n)\phi(X_n) r(X_n) \end{pmatrix},$$

$$A_1(X_n) = \begin{pmatrix} \beta_n \phi(X_n)\phi(X_n)^\top & 0 \\ 0 & (1-\beta_n)\phi(X_n)\phi(X_n)^\top \end{pmatrix},$$

$$\mathcal{E}(X_n, S_{n+1}, \theta_n) = \begin{pmatrix} \beta_n \gamma \phi(X_n)\phi(S_{n+1}, \pi_{\theta_n^A}(S_{n+1}))^\top \theta_n^B \\ (1-\beta_n)\gamma \phi(X_n)\phi(S_{n+1}, \pi_{\theta_n^B}(S_{n+1}))^\top \theta_n^A \end{pmatrix},$$

β_n are i.i.d. Bernoulli random variables with equal probability, and $\pi_\theta(s) = \arg\max_{a\in\mathcal{A}} \phi(s,a)^\top \theta$. While Double Q-learning avoids an overestimation bias, it introduces an underestimation bias (van Hasselt, 2010, Lemma 1), as each component of $\mathcal{E}$ satisfies[1]

$$\mathbb{E}[\phi(s, \pi_{\theta_n^A}(s))^\top \theta_n^B] \le \max_{a'\in\mathcal{A}} \mathbb{E}[\phi(s,a')^\top \theta_n^A] \quad \forall s \in \mathcal{S}.$$

It can be directly seen by the Jensen inequality (6) that in the special case of $\theta^A = \theta^B$ the inequality above becomes an equality. An inherent difficulty with the overestimation bias of standard Q-learning (resp. the underestimation bias of Double Q-learning) is that it cannot be controlled, i.e., the level of under-(resp. over-)estimation bias depending on the problem considered can be significant. In the following, we present a Q-learning method where the level of estimation bias can be precisely adjusted via a hyperparameter.

3. Regularization through Robust Averaging

We propose the 2RA Q-learning method defined by the update rule

$$\theta_{n+1}^{(i)} = \theta_n^{(i)} + \alpha_n \beta_n^{(i)} (b(X_n) - A_1(X_n)\theta_n^{(i)} + \mathcal{E}_\rho(X_n, S_{n+1}, \widehat{\theta}_{N,n})), \quad \text{for } i = 1, \dots, N, \tag{7}$$

where β_n is a generalized i.i.d. Bernoulli random variable on $\{1, \dots, N\}$ with equal probability for each component i, i.e., $\mathbb{P}(\beta_n = e_i) = 1/N$ for all $i = 1, \dots, N$, where e_i is the i$^{\text{th}}$ unit vector on $\mathbb{R}^N$ and α_n is the learning rate. 2RA Q-learning (7) is based on the estimator defined for all $x \in \mathcal{X}, s' \in \mathcal{S}, \theta \in \mathbb{R}^d$ as

$$\mathcal{E}_\rho(x, s', \theta) = \gamma \phi(x) \max_{a'\in\mathcal{A}} \min_{\theta' \in \mathcal{B}_\rho(\theta)} \phi(s', a')^\top \theta', \tag{8}$$

where $\rho \ge 0$ is a given parameter and the ambiguity (or uncertainty) set $\mathcal{B}_\rho(\theta)$ is assumed to be of the form $\mathcal{B}_\rho(\theta) = \{\theta' \in \mathbb{R}^d : \|\theta - \theta'\|_2^2 \le \rho\}$ with its center θ being the empirical average

$$\widehat{\theta}_{N,n} = \frac{1}{N} \sum_{i=1}^N \theta_n^{(i)}. \tag{9}$$

The intuition behind the proposed 2RA Q-learning (7) is to mitigate the overestimation bias of Watkins' Q-learning by approximating the term $\max_{a'\in\mathcal{A}} \mathbb{E}_{\mathbb{P}}[\phi(s,a')^\top \theta_n]$, where we consider θ_n as a $\mathbb{R}^d$-valued random variable distributed according to $\mathbb{P}$, via the distributionally robust model

$$\max_{a'\in\mathcal{A}} \min_{\mathbb{Q}\in\mathbb{B}_\rho(\widehat{\mathbb{P}}_{N,n})} \mathbb{E}_{\mathbb{Q}}[\phi(s,a')^\top \theta_n], \tag{10}$$

where $\mathbb{B}_\rho(\widehat{\mathbb{P}}_{N,n})$ is a set of probability measures centered around the empirical distribution $\widehat{\mathbb{P}}_{N,n} = \frac{1}{N}\sum_{i=1}^N \delta_{\theta_n^{(i)}}$. When considering the ambiguity set $\mathbb{B}_\rho(\widehat{\mathbb{P}}_{N,n})$ as the ball of all distributions that have a fixed diagonal covariance and a 2-Wasserstein distance to $\widehat{\mathbb{P}}_{N,n}$ of at most $\sqrt{\rho}$, then by (Nguyen et al., 2021, Theorem 2) the distributionally robust model (10) directly corresponds to our estimator (8). Running the 2RA Q-learning (7) requires an evaluation of the estimator $\mathcal{E}_\rho(X_n, S_{n+1}, \widehat{\theta}_{N,n})$ given by the optimization problem (8), which admits a closed form expression.

Lemma 3.1 (Estimator computation)**.** *The estimator defined in* (8) *is equivalently expressed as*

$$\mathcal{E}_\rho(x, s', \theta) = \gamma \phi(x) \max_{a'\in\mathcal{A}} \left\{ \phi(s', a')^\top \theta - \sqrt{\rho} \|\phi(s', a')\|_2 \right\}.$$

Proof. According to Bertsimas & Tsitsiklis (1997, Lemma 9.2), for any $c, \theta' \in \mathbb{R}^d$ and positive definite matrix $H \in \mathbb{R}^{d\times d}$ the optimization problem

$$\min_{\theta\in\mathbb{R}^d} \{c^\top \theta \ : \ (\theta' - \theta)^\top H^{-1} (\theta' - \theta) \le \rho\}$$

admits a closed-form solution

$$\theta^\star = \theta' - \sqrt{\frac{\rho}{c^\top H c}} H c.$$

Therefore, by setting $c = \phi(s', a')$ and H to be the identity matrix, an optimizer in (8) is

$$\theta^\star = \theta' - \frac{\sqrt{\rho}}{\|\phi(s', a')\|_2} \phi(s', a'),$$

which completes the proof. □

In the tabular setting 2RA Q-learning (7) even for $N = 2$ is different from Double Q-learning (van Hasselt, 2010). However, in the special case where $\rho = 0$ and $N = 1$, our method collapses to Watkins' Q-learning (5). In our proposed 2RA Q-learning, we've made a modification to the term $\mathcal{E}(X_n, S_{n+1}, \theta_n)$ from Wattkins Q-learning (5).

[1] Analogously we have $\mathbb{E}[\phi(s, \pi_{\theta_n^B}(s))^\top \theta_n^A] \le \max_{a'\in\mathcal{A}} \mathbb{E}[\phi(s,a')^\top \theta_n^B]$ for all $s \in \mathcal{S}$.

It is now replaced with a regularized version, denoted as $\mathcal{E}_\rho$ based on Lemma 3.1. This adjustment is combined with the averaging property using $\widehat{\theta}_{N,n}$. The regularization term $\sqrt{\rho}\|\phi(s',a')\|_2$ can be interpreted as negative UCB bonus term that discourages exploration, which has been considered for linear MDPs in (Jin et al., 2019), see also (Qian et al., 2019).

In the remainder of this section, we theoretically investigate 2RA Q-learning (7) and, in particular, study how to choose the two regularization parameters ρ, N, and the learning rate α_n. In Section 3.1, we show what properties the regularization term ρ should satisfy such that 2RA Q-learning (7) asymptotically converges to the optimal Q-function. This convergence is independent of the number N of Q-function estimates. Section 3.2 shows how the two terms ρ and N can be exploited to control the estimation bias of the presented scheme. Finally, Section 3.3 studies the convergence rate via the notion of the asymptotic mean squared error and, in particular, shows how to choose the learning rate α_n as compared to Watkins' Q-learning.

3.1. Asymptotic Convergence

The 2RA Q-learning (7) for the tabular setting actually converges almost surely to the optimal Q-function satisfying the Bellman equation (2), provided the radius ρ is chosen appropriately.

Theorem 3.1 (Asymptotic convergence)**.** *Consider the tabular setting where $d = |\mathcal{X}|$, Φ is the canonical basis and let $\{\rho_n\}_{n\in\mathbb{N}}$ be a sequence of non-negative numbers such that $\lim_{n\to\infty}\rho_n = 0$. Moreover, assume that*

(i) The learning rates satisfy $\alpha_n(s,a) \in (0,1]$, $\sum_{n=0}^{\infty}\alpha_n(s,a) = \infty$, $\sum_{n=0}^{\infty}\alpha_n^2(s,a) < \infty$ and $\alpha_n(s,a) = 0$ unless $(s,a) = (S_n, A_n)$.

(ii) The reward r is bounded.

Then, for any $N \in \mathbb{N}$, 2RA Q-learning (7) converges to the optimal Q-function $Q^\star$, i.e., $\lim_{n\to\infty}\Phi^\top\widehat{\theta}_{N,n} = Q^\star$ $\mathbb{P}$-almost surely.

Note that $\Phi^\top\widehat{\theta}_{N,n}$ is our learned 2RA Q-function under the canonical basis describing the tabular setting.

3.2. Estimation Bias

We now focus on the estimation bias of 2RA Q-learning induced by the term $\mathcal{E}_\rho$ in (8). While Watkins' Q-learning suffers from the mentioned overestimation bias, the proposed 2RA Q-learning (7) allows us to control the estimation bias via the parameters ρ and N. We show that for $\rho > 0$ with high probability, 2RA Q-learning generates an underestimation bias, somewhat similar to Double Q-learning. However, in contrast to Double Q-learning, we can control the level of underestimation via the parameter ρ. Moreover, the second parameter N, describing the number of action-value estimates, further allows us to control the estimation bias.

Theorem 3.2 (Estimation bias)**.**

(i) Consider any $N, n \in \mathbb{N}$, $\rho \geq 0$ and $i \in \{1,\dots,N\}$. If $\mathbb{E}[\theta_n^{(i)}] \in \mathcal{B}_\rho(\widehat{\theta}_{N,n})$, then the robust estimator defined in (8) provides an underestimation where the level of underestimation is controlled by ρ, i.e., for all $x \in \mathcal{X}, s' \in \mathcal{S}$

$$0 \leq \gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] - \mathbb{E}[\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n})]$$
$$\leq \sqrt{\rho}\gamma\phi(x)\max_{a'\in\mathcal{A}}\|\phi(s',a')\|_2.$$

(ii) Let $\theta^{(i)}$ be initialized with the same value for $i = 1,\dots,N$. For any $\rho \geq 0$ and $n \in \mathbb{N}$, the robust estimator (8) satisfies for all $x \in \mathcal{X}, s' \in \mathcal{S}$

$$\lim_{N\to\infty}\mathbb{E}[\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n})]$$
$$= \gamma\phi(x)\max_{a'\in\mathcal{A}}\left\{\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] - \sqrt{\rho}\|\phi(s',a')\|_2\right\}.$$

Assertion (ii) directly implies that for $\rho = 0$, the robust estimator (8) is unbiased in the limit as $N \to \infty$. We can alternatively show this via Theorem 3.2(i). By following the proof of Theorem 3.2, we can show that for any $\rho > 0$

$$\lim_{N\to\infty}\mathbb{P}\left(\mathbb{E}[\theta_n^{(i)}] \in \mathcal{B}_\rho(\widehat{\theta}_{N,n})\right) = 1, \tag{11}$$

i.e., for any given ρ if N is chosen large enough, we can expect the assumption of Assertion (i) of Theorem 3.2 to hold. To derive (11) note that in the proof of Theorem 3.2, we show (see (23) and apply Hoelder's inequality) that $\lim_{N\to\infty}\mathbb{E}[\|\widehat{\theta}_{N,n} - \mathbb{E}[\theta_n^{(i)}]\|] = 0$. Markov's inequality states that for any $\rho > 0$

$$\mathbb{P}\left(\|\widehat{\theta}_{N,n} - \mathbb{E}[\theta_n^{(i)}]\| > \sqrt{\rho}\right) \leq \frac{1}{\sqrt{\rho}}\mathbb{E}\left[\|\widehat{\theta}_{N,n} - \mathbb{E}[\theta_n^{(i)}]\|\right],$$

which directly implies (11).

Corollary 3.1 (Vanishing estimation bias)**.** *Under the assumptions of Theorem 3.1 and a regularization sequence $\{\rho_n\}_{n\in\mathbb{N}} \subset \mathbb{R}_+$ such that $\lim_{n\to\infty}\rho_n = 0$, for any $N \in \mathbb{N}$ and $i \in \{1,\dots,N\}$*

$$\lim_{n\to\infty}\mathbb{E}[\mathcal{E}_{\rho_n}(x,s',\widehat{\theta}_{N,n})]$$
$$= \lim_{n\to\infty}\gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] \quad \forall x \in \mathcal{X}, s' \in \mathcal{S}.$$

Theorem 3.2 allows us to interpret the choice of regularization $\{\rho_n\}_{n\in\mathbb{N}}$ in a non-asymptotic manner.

Remark 3.1 (Selection of parameter ρ_n)**.** *We have shown in Theorem 3.1 that convergence of 2RA Q-learning (7) requires a sequence $\{\rho_n\}_{n\in\mathbb{N}}$ such that $\lim_{n\to\infty}\rho_n = 0$. Theorem 3.2 provides insights into how the specific decay of ρ_n determines the resulting performance of 2RA Q-learning. More precisely, Theorem 3.2 describes the inherent trade-off in the selection of ρ_n: choosing larger values of ρ_n increases the probability that $\mathbb{E}[\theta_n^{(i)}] \in \mathcal{B}_{\rho_n}(\widehat{\theta}_{N,n})$ which guarantees an underestimation bias. On the other hand, choosing smaller values of ρ_n decrease the level of underestimation bias but potentially introduce an overestimation bias as the probability that $\mathbb{E}[\theta_n^{(i)}] \notin \mathcal{B}_{\rho_n}(\widehat{\theta}_{N,n})$ increases. We further comment on the choice of regularization ρ_n in the numerical experiments, Section 4.*

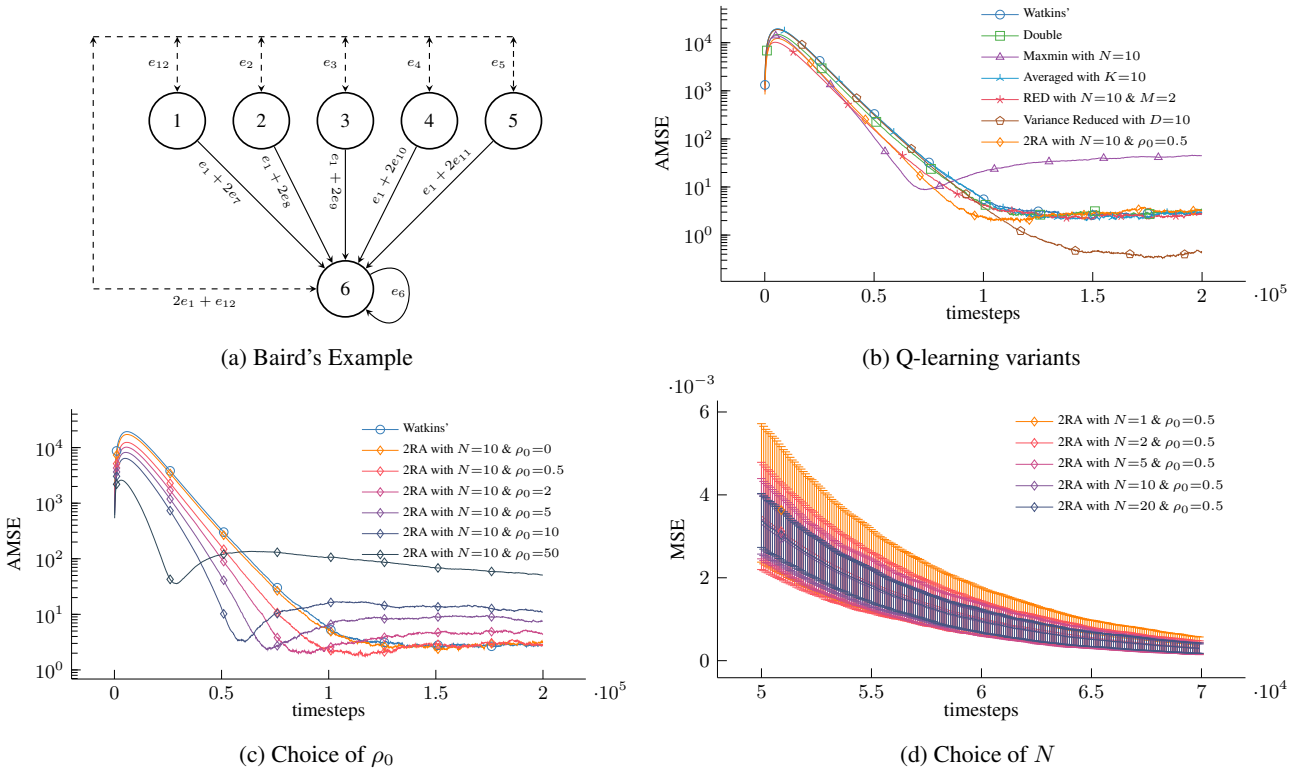
Remark 3.2 (Selection of parameter N)**.** *The convergence of 2RA Q-learning holds for any choice of N; see Theorem 3.1. Moreover, Theorem 3.2 states that increasing N decreases the estimation bias. Choosing the parameter N too large, however, when using a learning rate that is N-times the learning rate of Watkins' Q-learning (according to Theorem 3.3) can lead to numerical instability. Therefore, in practice, a trade-off must be made when selecting N.*

3.3. Asymptotic Mean-Squared Error

We have shown in Theorem 3.1 that 2RA Q-learning (7) asymptotically converges to the optimal Q-function. This section investigates the convergence rate via the so-called asymptotic mean-squared error. Throughout this section, we consider a tabular setting and assume without loss of generality that the optimal Q-function is such that $\theta^\star = 0$. If $\theta^\star \neq 0$, the results can hold by subtracting $\theta^\star$ from the estimators of the Q-learning, see Devraj & Meyn (2017). Given the 2RA Q-learning and the corresponding estimator $\widehat{\theta}_{N,n}$ as introduced in (9), we define its asymptotic mean-squared error (AMSE) as the limit of a scaled covariance

$$\mathsf{AMSE}(\widehat{\theta}_N) = \lim_{n\to\infty} n\mathbb{E}[\widehat{\theta}_{N,n}^\top \widehat{\theta}_{N,n}] = \lim_{n\to\infty} n\mathbb{E}[\|\widehat{\theta}_{N,n}\|_2^2].$$

Our analysis also discusses the choice of the learning rate in 2RA Q-learning compared to the learning rate of Watkins'



(a) Baird's Example

(b) Q-learning variants

(c) Choice of ρ_0

(d) Choice of N

Figure 1: Baird's Example. All Methods use an initial learning rate of $\alpha_0 = 0.01$, $w_\alpha = 10^5$, and $\gamma = 0.8$. All 2RA agents additionally use $w_\rho = 10^3$. The reward function has values random-uniformly sampled from $[-0.05, 0.05]$. All results are average over 100 consecutive experiments. (a) Baird's example environment with the feature vectors for each state-action pair. (b) Comparison of the AMSE of Watkins Q-learning, Double Q-learning, Maxmin Q-learning with $N = 10$, where the 2RA Q-learning uses initial $\rho_0 = 0.5$ and $N = 10$. (c) Comparison of the AMSE of 2RA Q-learning with $N = 10$ but different initial values ρ_0. (d) Experiment showing the MSE in terms of mean and standard deviation for different values of N with $\rho_0 = 0.5$.

Q-learning.

Theorem 3.3 (AMSE for 2RA Q-learning)**.** *Consider a setting where the regularization sequence $\{\rho_n\}_{n\in\mathbb{N}} \subset \mathbb{R}_+$ is such that $\lim_{n\to\infty} \rho_n = 0$ and $N \in \mathbb{N}$. Let $\alpha_n^{\mathsf{QL}} = g/n$ be the learning rate of Watkins' Q-learning (5) and consider the 2RA Q-learning (7) with learning rate $\alpha_n = N \cdot g/n$, where g is a positive constant[2]. Then, there exists some $g_0 > 0$ such that for any $g > g_0$*

$$\mathsf{AMSE}(\widehat{\theta}_N) = \mathsf{AMSE}(\theta^{\mathsf{QL}}),$$

where $\{\theta_n^{\mathsf{QL}}\}_{n\in\mathbb{N}}$ is a sequence generated by Watkins' Q-learning algorithm.

Remark 3.3 (Assumption on learning rate)**.** *As pointed out by Weng et al. (2020) the condition $g > g_0$ in the tabular setting reduces to $g > \frac{1}{\mu_{\min}(1-\gamma)}$, where $\mu_{\min}$ denotes the minimum entry of the stationary distribution of the state.*

4. Numerical Results

We numerically[3] compare our presented 2RA Q-learning (7) with Watkins' Q-learning (3), Hasselt's Double Q-learning (van Hasselt, 2010), and with the Maxmin Q-learning (Lan et al., 2020). First, we look at Baird's Example (Baird, 1995), then we consider arbitrary, randomly generated MDP environments with fixed rewards, and last, the CartPole example (Barto et al., 1983; Brockman et al., 2016). In all experiments[4], we choose a step size $\alpha_n = \frac{N\alpha_0 w_\alpha}{n+w_\alpha}$, where N is the number of state-action estimates used in the respective learning method, $w_\alpha > 0$ is a weight parameter, and α_0 is the initial step size. The decay rate for the regularization parameter ρ_n, as required for convergence (see Theorem 3.1), is chosen to be either $\rho_n = \frac{\rho_0 w_\rho}{n+w_\rho}$ or $\rho_n = \frac{\rho_0 w_\rho}{n^2+w_\rho}$ with exact parameters given for each experiment and a more detailed evaluation at the end of this section.

Baird's Example. We consider the setting described in Weng et al. (2020). The environment (Figure 1a) has six states and two actions. Under action one, the transition probability to any of the six states is $1/6$, and action two results in a deterministic transition to state six. These transition dynamics are independent of the state at which an action is chosen. Therefore the trajectories on which the methods are updated can be obtained from a random uniform behavioral policy that allows every state to be visited. The features vectors $\phi(s,a)$ are constructed as shown in Figure 1a where each $e_i \in \mathbb{R}^{12}$ is the i^{th} unit vector. In this setting, it is known that the optimal policy is unique (Weng et al., 2020), and our theoretical results apply. All $\theta^{(i)}$ are initialized, as in Weng et al. (2020), uniformly at random with values in $[0, 2]$. Figures 1b and 1c have a log-scaled y-axis to emphasize the smaller differences between models as they converge. The first important observation is that all learning methods converge to the same AMSE, which is in line with Theorem 3.3. An exception is Maxmin Q-learning, for which, however, no theoretical statement regarding the expected behavior of its AMSE is made. Higher values for ρ increase the convergence speed in the early learning phase, as shown in Figure 1c. However, if ρ is too large or its relative decay too slow, learning eventually slows down (or even temporarily worsens) as large values of ρ make the update steps too big. For the proposed choice of ρ, our 2RA-method outperforms the other learning methods in the first part of the learning process without getting significantly slower in the long run. Only the AMSE of Variance Reduced Q-Learning outperforms all other methods which appears to be caused by the specific instance of Baird's experiment (compare results of the Random Environment). Our next experiment shows that 2RA and Maxmin Q-Learning are sensitive towards different environments which prefer over-, under-, or no estimation bias at all. Figure 1d uses a non-log-scaled y-axis to ensure the size of the standard errors is comparable. It can be observed how increasing the parameter N reduces the standard error of the learning across multiple experiments. However, it is also apparent that the marginal utility of each additional theta decreases fast.

Random Environment. This experiment visualizes how different learning methods, with a fixed set of hyperparameters, behave under changes in the environment's transition dynamics. For $|\mathcal{A}| = 3$ and $|\mathcal{S}| = 10$, we consider a random environment that is described by a transition probability matrix, which, for each pair $s \in \mathcal{S}$ and $a \in \mathcal{A}$, is drawn from a Dirichlet distribution with uniform parameter 0.1. Analogously, we draw a distribution of the initial states s_0. Similar to Baird's example, these MDPs are ergodic and random uniform behavioral policies can be used to generate trajectories based on which updates are performed. We further consider a quadratic reward function $r(s,a) = -qs^2 - pa^2$ for all possible environments, where $p, q \in \mathbb{R}_+$ are such that $p < q$. Therefore, different environments have the same reward function but different transition dynamics. For our experiments, we chose $q = 0.1$ and $p = 0.01$. Each environment of Figure 2 is randomly drawn, in sequence, from the same random seed. The resulting dynamics vary significantly between different environments as only one constant selection of hyperparameters is used, but 2RA Q-learning consistently outperforms the other methods in the early stages of learning.

With the exception of Maxmin, the other benchmarks perfom similarly to Watkins' Q-Learning. A further obser-

[2]We assume that our starting index for n is large enough such that $Ng/n < 1$.

[3]Here: github.com/2RAQ/code

[4]Except for REDQ, since the learning rate would be too high for the multiple updates per step.

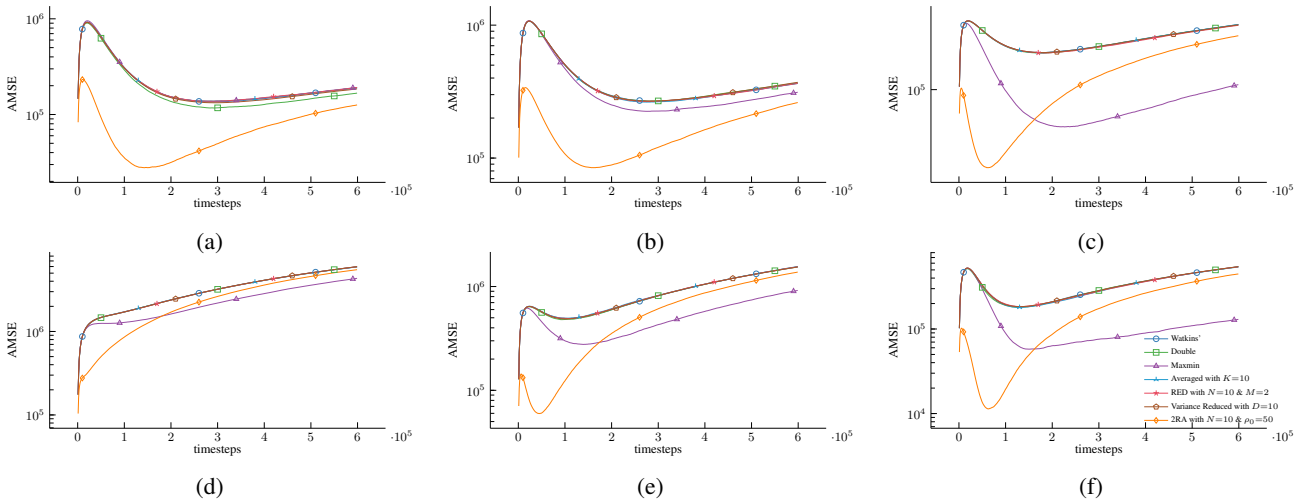


Figure 2: Random Environment. All methods use an initial learning rate of $\alpha_0 = 0.01$, $w_\alpha = 10^5$, $\gamma = 0.9$, and all $\theta^{(i)}$ initialized as zero. Maxmin as well as 2RA Q-learning have $N = 10$ and 2RA agents additionally use $\rho_0 = 50$ and $w_\rho = 10^4$. The plots show the first six randomly drawn environments and all results are average over 100 consecutive experiments. A broader plot of the first 20 random environments is provided in Figure 4 in Appendix C.

vation is that the better 2RA Q-learning performs in an environment, the better Double Q-learning performs. This indicates that these are environments where an underestimation bias is beneficial (Lan et al., 2020), with the strength of the effect varying with the drawn transition dynamics.

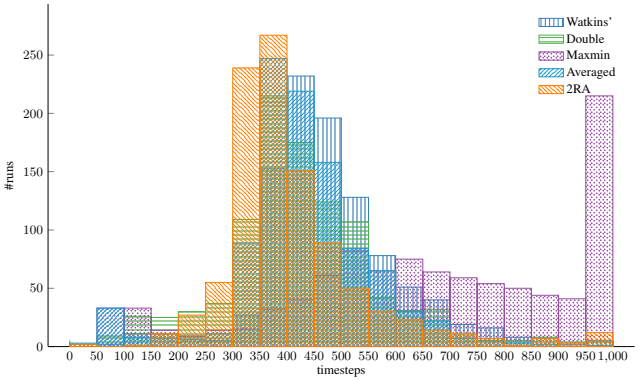
CartPole. The well-known CartPole environment (Brockman et al., 2016) serves as a more practical application while still using the same linear function approximation model from the previous experiments in combination with a discretized CartPole state space. For each timestep, in which the agent can keep the pole within an allowed deviation from an upright angle and the cart's starting position along the horizontal axis, it receives a reward of $+1$. An episode ends if either one of the thresholds for allowed deviation is broken. Since a random uniform behavioral policy would not enable visits to all regions of the state space, the latest updated policy combined with ϵ-greedy exploration is used to generate the next timestep which is then used to update the model; Updates are applied after each timestep. We compare the different learning algorithms based on how many training episodes are required to solve the CartPole task. The task is considered to be solved if, during the evaluation, the average reward over 100 episodes with a maximum allowed stepcount of 210 reaches or exceeds 195. Across 1000 experiments, the number of episodes until the task is solved (hit times) is collected for each learning method. Methods that, on average, solve the environment with fewer training episodes are ranked higher in the performance comparison. As CartPole benefits from a high learning rate, an initial $\alpha_0 = 0.4$ is chosen and decayed per episode e, as compared to the decay per timestep of the previous experiments, such that $\alpha_e = \alpha_0 \frac{w_\alpha}{e+w_\alpha}$.

Comparing the hit time distributions of the different algorithms shows that the 2RA mean performance is better than Double Q-learning, which outperforms both Watkins' and Maxmin Q-learning by a significant margin. CartPole appears to benefit from the underestimation bias introduced by Double and 2RA Q-learning. This is consistent with the previous experiments where the good performance of 2RA Q-learning correlates with good performance of Double Q-learning. Since this experiment has deterministically initialized θ_0 as well as deterministic state transitions, REDQ and Variance Reduced Q-Learning are not comparable in this setting.

In Appendix C.2, we provide an additional example, where we test 2RA Q-learning when used with neural network Q-function approximation, applied to the LunarLander environment (Brockman et al., 2016). Also there, 2RA Q-learning shows good performance, despite the fact, that our theoretical results do not apply.

5. Discussion and Conclusion

In this work, we proposed 2RA Q-learning and showed that it enables control of the estimation bias via the parameters N and ρ while maintaining the same asymptotic convergence guarantees as Double and Watkins' Q-learning. In practice, the control of the estimation bias enables faster convergence to a good-performing policy in finitely many steps which is caused by the intrinsic property of environments to favor an over-, an under-, or no estimation bias at all. Therefore, determining the optimal bias adjustment is highly dependent



(a) Distribution of hit times

Algorithm	Mean hit time
Watkins' Q-Learning	457.35 ± 128.18
Double Q-Learning	401.89 ± 144.43
Maxmin Q-Learning	645.02 ± 270.01
Averaged Q-Learning	404.09 ± 124.19
2RA Q-Learning	386.19 ± 133.47

(b) Mean and std of hit times

Figure 3: Cartpole, 1000 experiments. All methods use an initial learning rate of $\alpha_0 = 0.4$, $w_\alpha = 100$, $\gamma = 0.999$ and all $\theta^{(i)}$ initialized as zero. Maxmin, as well as 2RA Q-learning, have $N = 8$. 2RA further uses $\rho_0 = 150$ and $w_\rho = 10^4$. All algorithms are evaluated after every 50 episodes and recorded if the average evaluation reward reaches or exceeds 195. (a) Shows the distributions of each algorithm's hit times and (b) lists the respective mean hit times and corresponding standard deviations.

on the specific environment and rigorous analysis of environments' bias preferences is not yet available. To account for this, 2RA Q-learning provides two additional tuning parameters that can be used to fine-tune learning for these environment preferences. This level of control, combined with computational costs comparable to existing methods, makes 2RA Q-learning a valuable addition to the RL tool belt. The conducted numerical experiments for various settings corroborate our theoretical findings and highlight that 2RA Q-learning generally performs well and mostly outperforms other Q-learning variants.

Acknowledgements

This work was supported by the DFG in the Cluster of Excellence EXC 2117 "Centre for the Advanced Study of Collective Behaviour" (Project-ID 390829875).

Impact Statement

This paper presents work whose goal is to advance the field of Machine Learning. There are many potential societal consequences of our work, none which we feel must be specifically highlighted here.

More specifically, this work is theoretical and not intended to be directly applicable to a real-world application. Yet, there is a clear real-world motivation. It is widely believed that to move any learning algorithm (including Q-learning) out of the labs into our lives, we need better and more statistical guarantees and clear theoretical understanding. At the moment, great progress is made into this area, but the vast majority of tools is tailor-made to the application, which means that practitioners need to understand all of these different tools to safely apply them. Our work aims to assist practitioners by presenting a general framework for controlling the estimation bias of Q-learning.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G. S., Davis, A., Dean, J., Devin, M., Ghemawat, S., Goodfellow, I., Harp, A., Irving, G., Isard, M., Jia, Y., Jozefowicz, R., Kaiser, L., Kudlur, M., Levenberg, J., Mané, D., Monga, R., Moore, S., Murray, D., Olah, C., Schuster, M., Shlens, J., Steiner, B., Sutskever, I., Talwar, K., Tucker, P., Vanhoucke, V., Vasudevan, V., Viégas, F., Vinyals, O., Warden, P., Wattenberg, M., Wicke, M., Yu, Y., and Zheng, X. TensorFlow: Large-scale machine learning on heterogeneous systems, 2015.

Anschel, O., Baram, N., and Shimkin, N. Averaged-DQN: Variance reduction and stabilization for deep reinforcement learning. In *International Conference on Machine Learning*, 2017.

Baird, L. Residual algorithms: Reinforcement learning with function approximation. In *International Conference on Machine Learning*, 1995.

Barto, A. G., Sutton, R. S., and Anderson, C. W. Neuronlike adaptive elements that can solve difficult learning control problems. *Transactions on Systems, Man, and Cybernetics*, 1983.

Beck, C. and Srikant, R. Error bounds for constant step-size Q-learning. *Systems & Control Letters*, 61(12), 2012.

Bellman, R. *Dynamic Programming*. Princeton University Press, 1957.

Bertsekas, D. P. and Tsitsiklis, J. *Neuro-Dynamic Programming*. Athena Scientific, 1996.

Bertsimas, D. and Tsitsiklis, J. N. *Introduction to Linear Optimization*. Athena Scientific, 1997.

Billingsley, P. *Convergence of probability measures*. John Wiley & Sons Inc., 1999.

Bohrnstedt, G. W. and Goldberger, A. S. On the exact covariance of products of random variables. *Journal of the American Statistical Association*, 64(328), 1969.

Brockman, G., Cheung, V., Pettersson, L., Schneider, J., Schulman, J., Tang, J., and Zaremba, W. Openai gym, 2016.

Brockwell, P. J. *Time series: Theory and methods*. Springer-Verlag, 1991.

Chen, S., Devraj, A. M., Busic, A., and Meyn, S. P. Explicit Mean-Square Error Bounds for Monte-Carlo and Linear Stochastic Approximation. In *International Conference on Artificial Intelligence and Statistics*, 2020.

Chen, X., Wang, C., Zhou, Z., and Ross, K. W. Randomized ensembled Double Q-Learning: Learning fast without a model. In *International Conference on Learning Representations*, 2021.

Devraj, A. M. and Meyn, S. P. Fastest convergence for Q-learning. *ArXiv preprint*, abs/1707.03770, 2017.

Durrett, R. *Probability: Theory and Examples*. Cambridge University Press, 2010.

Even-Dar, E. and Mansour, Y. Learning Rates for Q-Learning. In *Annual Conference on Computational Learning Theory*, 2001.

Gosavi, A. Boundedness of iterates in q-learning. *Systems & Control Letters*, 55(4):347–349, 2006. ISSN 0167-6911.

He, K., Zhang, X., Ren, S., and Sun, J. Delving Deep into Rectifiers: Surpassing Human-Level Performance on ImageNet Classification. In *International Conference on Computer Vision*, 2015.

Hernández-Lerma, O. and Lasserre, J. *Further topics on discrete-time Markov control processes*. Springer, 1999.

Hernández-Lerma, O. and Lasserre, J. B. *Discrete-time Markov control processes: Basic optimality criteria*. Springer, 1996.

Huber, P. J. Robust Estimation of a Location Parameter. *The Annals of Mathematical Statistics*, 35(1), 1964.

Jaakkola, T., Jordan, M. I., and Singh, S. P. On the convergence of stochastic iterative dynamic programming algorithms. *Neural Computation*, 6(6), 1994.

Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. Provably efficient reinforcement learning with linear function approximation. *arXiv preprint, arXiv:1907.05388*, 2019.

Kingma, D. P. and Ba, J. Adam: A Method for Stochastic Optimization. In *International Conference on Learning Representations*, 2015.

Lan, Q., Pan, Y., Fyshe, A., and White, M. Maxmin Q-learning: Controlling the Estimation Bias of Q-learning. In *International Conference on Learning Representations*, 2020.

Lee, D. and He, N. A Unified Switching System Perspective and Convergence Analysis of Q-Learning Algorithms. In *Annual Conference on Neural Information Processing*, 2020.

Lim, H.-D., Kim, D. W., and Lee, D. Regularized Q-learning. *arXiv preprint*, 2202.05404, 2022.

Liu, Z., Bai, Q., Blanchet, J., Dong, P., Xu, W., Zhou, Z., and Zhou, Z. Distributionally robust Q-learning. In *International Conference on Machine Learning*, 2022.

Lu, F. and Meyn, S. Convex q learning in a stochastic environment: Extended version. *ArXiv preprint*, abs/2309.05105, 2023.

Lu, F., Mehta, P., Meyn, S., and Neu, G. Sufficient exploration for convex Q-learning. *arXiv preprint*, 2210.09409, 2022.

Mehta, P. G. and Meyn, S. P. Convex Q-learning, part 1: Deterministic optimal control. *arXiv preprint*, 2008.03559, 2020.

Melo, F. S., Meyn, S. P., and Ribeiro, M. I. An analysis of reinforcement learning with function approximation. In *International Conference on Machine Learning*, 2008.

Meyn, S. *Control Systems and Reinforcement Learning*. Cambridge University Press, 2022.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., and Hassabis, D. Human-level control through deep reinforcement learning. *Nature*, 518(7540), 2015.

Neufeld, A. and Sester, J. Robust Q-learning algorithm for markov decision processes under wasserstein uncertainty. *arXiv preprint*, 2210.00898, 2022.

Nguyen, V. A., Abadeh, S. S., Filipović, D., and Kuhn, D. Mean-Covariance Robust Risk Measurement. *ArXiv preprint*, abs/2112.09959, 2021.

Qian, J., Fruit, R., Pirotta, M., and Lazaric, A. Exploration bonus for regret minimization in discrete and continuous average reward mdps. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

Qu, G. and Wierman, A. Finite-time analysis of asynchronous stochastic approximation and Q-learning. In *Conference on Learning Theory*, 2020.

Singh, S. P., Jaakkola, T. S., Littman, M. L., and Szepesvári, C. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine Learning*, 38(3), 2000.

Strehl, A. L., Li, L., and Littman, M. L. Reinforcement learning in finite MDPs: PAC analysis. *Journal of Machine Learning Research*, 10(84), 2009.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT Press, 2018.

Szepesvári, C. The asymptotic convergence-rate of Q-learning. In *Advances in Neural Information Processing Systems*, 1997.

Szita, I. and Lörincz, A. The many faces of optimism: a unifying approach. In *International Conference on Machine Learning*, 2008.

Thrun, S. and Schwartz, A. Issues in using function approximation for reinforcement learning. In *Connectionist Models Summer School*, 1993.

Tsitsiklis, J. N. Asynchronous stochastic approximation and Q-learning. *Machine Learning*, 16, 1994.

van Hasselt, H. Double Q-learning. In *Annual Conference on Neural Information Processing Systems*, 2010.

Vieillard, N., Pietquin, O., and Geist, M. Munchausen Reinforcement Learning. In *nnual Conference on Neural Information Processing Systems*, 2020.

Wainwright, M. J. Variance-reduced Q-learning is minimax optimal. *arXiv preprint*, 1906.04697, 2019.

Watkins, C. Learning from delayed rewards. *PhD thesis, King's College, Cambridge*, 1989.

Watkins, C. J. C. H. and Dayan, P. Q-learning. *Machine Learning*, 8(3), 1992.

Weng, W., Gupta, H., He, N., Ying, L., and Srikant, R. The Mean-Squared Error of Double Q-Learning. In *Annual Conference on Neural Information Processing Systems*, 2020.

A. Proofs

A.1. Asymptotic Convergence

The proof of Theorem 3.1 is based on the following technical stochastic approximation result. We denote by $\|\cdot\|_w$ a weighted maximum norm with weight $w = (w_1, \dots, w_d)$, $w_i > 0$. If $x \in \mathbb{R}^d$, then $\|x\|_w = \max_i \frac{|x_i|}{w_i}$.

Lemma A.1. *(Singh et al., 2000, Lemma 1) Consider a stochastic process $\{(\alpha_n, \Delta_n, F_n)\}_{n\geq 0}$, where $\alpha_n, \Delta_n, F_n : \mathcal{X} \to \mathbb{R}$ satisfy the equations*

$$\Delta_{n+1}(x) = (1 - \alpha_n(x))\Delta_n(x) + \alpha_n(x)F_n(x), \quad x \in \mathcal{X},\ n = 0, 1, \dots \tag{12}$$

Let P_n be a sequence of increasing σ-fields such that α_0 and Δ_0 are P_0-measurable and α_n, Δ_n and F_{n-1} are P_n-measurable for $n = 1, 2, \dots$. Assume the following hold

(i) the set $\mathcal{X}$ is finite;

(ii) $\alpha_n(x) \in (0, 1]$, $\sum_{n=1}^{\infty} \alpha_n(x) = \infty$, $\sum_{n=1}^{\infty} \alpha_n^2(x) < \infty$ almost surely;

(iii) $\|\mathbb{E}[F_n(\cdot)|P_n]\|_w \leq \kappa\|\Delta_n\|_w + c_n$, where $\kappa \in [0, 1)$ and c_n converges to zero almost surely;

(iv) $\mathsf{Var}(F_n(x)|P_n) \leq K(1 + \|\Delta_n\|_w)^2$, where K is some constant.

Then, Δ_n converges to zero almost surely as $n \to \infty$.

Proof of Theorem 3.1. The proof builds up on the convergence results of SARSA (Singh et al., 2000), Double Q-learning (van Hasselt, 2010) and uses Lemma A.1 as a key ingredient. For the convenience of notation, we carry out the proof in the Q-function notation. That is, in the tabular setting, where no function approximation is applied, by invoking Lemma 3.1, the proposed 2RA Q-learning (7) is expressed as

$$\begin{aligned} Q_{n+1}^{(i)}(S_n, A_n) = Q_n^{(i)}(S_n, A_n) \\ + \alpha_n\beta_n^{(i)}\big(r(S_n, A_n) + \gamma(\max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1}, a') - \sqrt{\rho_n}) - Q_n^{(i)}(S_n, A_n)\big), \end{aligned} \tag{13}$$

where $\widehat{Q}_{N,n}(s, a) = \frac{1}{N}\sum_{i=1}^{N} Q_n^{(i)}(s, a)$ for $s \in \mathcal{S}$, $a \in \mathcal{A}$. In the following, we fix an arbitrary index $i \in \{1, \dots, N\}$ and with regard to Lemma A.1, we define P_n as the σ-field generated by $\{S_n, A_n, \alpha_n, \dots, S_0, A_0, \alpha_0, Q_0^{(1)}, \dots, Q_0^{(N)}\}$, $\mathcal{X} = \mathcal{S} \times \mathcal{A}$, $\Delta_n = Q_n^{(i)} - Q^\star$ and $F_n(S_n, A_n) = r(S_n, A_n) + \gamma(\max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1}, a') - \sqrt{\rho_n}) - Q^\star(S_n, A_n)$. Then, 2RA Q-learning (7) can be expressed as an instance of (12). To apply Lemma A.1, we need to ensure its assumptions are satisfied. Assumption (i) and (ii) clearly hold.

To show Assumption (iii), we note that the term F_n can be alternatively expressed as

$$F_n(S_n, A_n) = F_n^{Q^{(i)}}(S_n, A_n) + \gamma\left(\max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1}, a') - \max_{a'\in\mathcal{A}} Q_n^{(i)}(S_{n+1}, a') - \sqrt{\rho_n}\right), \tag{14}$$

with

$$F_n^{Q^{(i)}}(S_n, A_n) = r(S_n, A_n) + \gamma \max_{a'\in\mathcal{A}} Q_n^{(i)}(S_{n+1}, a') - Q^\star(S_n, A_n), \tag{15}$$

where the term $F_n^{Q^{(i)}}(S_n, A_n)$ corresponds to Watkins' Q-learning for $Q_n^{(i)}$. Therefore, it is well-known (Jaakkola et al., 1994, Theorem 2) that $\|\mathbb{E}[F_n^{Q^{(i)}}(\cdot)|P_n]\|_w \leq \gamma\|\Delta_n\|$, which via (14) implies that $\|\mathbb{E}[F_n(\cdot)|P_n]\|_w \leq \gamma\|\Delta_n\| + c_n$, where

$$c_n = \gamma\mathbb{E}\left[\max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1}, a') - \max_{a'\in\mathcal{A}} Q_n^{(i)}(S_{n+1}, a') - \sqrt{\rho_n} \mid P_n\right]. \tag{16}$$

It remains to show that c_n converges to zero almost surely. Recall that by assumption $\lim_{n\to\infty} \rho_n = 0$. We define

$$\delta_n^{(i)}(s, a) = Q_n^{(i)}(s, a) - \widehat{Q}_{N,n}(s, a)$$

and will show that $\lim_{n\to\infty} \|\delta_n^{(i)}(\cdot,\cdot)\| = 0$ almost surely. The reverse triangle inequality applied to the ∞-norm implies that $\mathbb{P}$-almost surely

$$\lim_{n\to\infty} \left(\max_{a'\in\mathcal{A}} Q_n^{(i)}(S_{n+1}, a') - \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1}, a') \right) = 0$$

and hence that c_n converges to zero almost surely.

We distinguish two cases. First, we consider an update on component i, then by (13)

$$\begin{aligned}
\delta_{n+1}^{(i)}(s,a) &= Q_{n+1}^{(i)}(s,a) - \widehat{Q}_{N,n+1}(s,a) \\
&= Q_n^{(i)}(s,a) + \alpha_n \left(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(i)}(s,a) \right) - \widehat{Q}_{N,n}(s,a) \\
&\quad - \frac{1}{N} \left(Q_{n+1}^{(i)}(s,a) - Q_n^{(i)}(s,a) \right) \\
&= \delta_n^{(i)}(s,a) + \left(\alpha_n - \frac{\alpha_n}{N} \right) \left(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(i)}(s,a) \right),
\end{aligned}$$

where the second equality follows from the decomposition $\frac{1}{N}\sum_{j=1}^N Q_{n+1}^{(j)}(s,a) = \frac{1}{N}(\sum_{j\neq i} Q_n^{(j)} + Q_{n+1}^{(i)})$. The third equality then uses our proposed Q-learning update formula (13) given as $Q_{n+1}^{(i)}(s,a) = Q_n^{(i)}(s,a) + \alpha_n(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(i)}(s,a))$. On the other hand, if the update is performed on component $j \neq i$, then

$$\begin{aligned}
\delta_{n+1}^{(i)}(s,a) &= Q_{n+1}^{(i)}(s,a) - \widehat{Q}_{N,n+1}(s,a) \\
&= Q_n^{(i)}(s,a) - \widehat{Q}_{N,n}(s,a) - \frac{1}{N}(Q_{n+1}^{(j)}(s,a) - Q_n^{(j)}(s,a)) \\
&= \delta_n^{(i)}(s,a) - \frac{\alpha_n}{N}(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(j)}(s,a)).
\end{aligned}$$

Hence, in total, we get

$$\begin{aligned}
&\mathbb{E}[\delta_{n+1}^{(i)}(s,a)|P_n] \\
&= \frac{1}{N}\mathbb{E}\left[\delta_n^{(i)}(s,a) + \frac{N-1}{N}\alpha_n(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(i)}(s,a))|P_n \right] \\
&\quad + \frac{1}{N}\sum_{j\neq i} \mathbb{E}\left[\delta_n^{(i)}(s,a) - \frac{\alpha_n}{N}(r(s,a) + \gamma \max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(s',a') - \gamma\sqrt{\rho_n} - Q_n^{(j)}(s,a))|P_n \right] \\
&= \mathbb{E}\left[(1 - \frac{\alpha_n}{N})\delta_n^{(i)}(s,a)|P_n \right] \\
&= (1 - \frac{\alpha_n}{N})\mathbb{E}\left[\delta_n^{(i)}(s,a)|P_{n-1} \right],
\end{aligned}$$

where the second equality follows from the observation that $\sum_{j\neq i} Q_n^{(j)} = N\widehat{Q}_{N,n} - Q_n^{(i)}$. Recall that $\delta_n^{(i)}(s,a) = \mathbb{E}[\delta_n^{(i)}(s,a)|P_{n-1}]$ for any n. Hence, we have derived the update rule

$$\delta_{n+1}^{(i)}(s,a) = (1 - \frac{\alpha_n}{N})\delta_n^{(i)}(s,a), \tag{17}$$

which directly implies that $\lim_{n\to\infty} \delta_n^{(i)}(s,a) = 0$ almost surely for all $(s,a) \in \mathcal{S}\times\mathcal{A}$. Since $\mathcal{S}$ and $\mathcal{A}$ are finite sets this implies that $\lim_{n\to\infty} \|\delta_n^{(i)}\| = 0$ almost surely as desired. Hence, $\lim_{n\to\infty} c_n = 0$ almost surely, which ensures Assumption (iii).

We finally show that Assumption (iv) holds. Again we use the decomposition (14). Since the reward r is assumed to be bounded, it is known (Singh et al., 2000) that

$$\mathsf{Var}(F_n^{Q^{(i)}}(x)|P_n) \leq K(1 + \|\Delta_n\|_w)^2, \tag{18}$$

where again $\Delta_n = Q_n^{(i)} - Q^\star$. We next show that there exists a constant K_2 such that

$$\mathsf{Var}\left(\max_{a'\in\mathcal{A}} \widehat{Q}_{N,n}(S_{n+1},a') - \max_{a'\in\mathcal{A}} Q_n^{(i)}(S_{n+1},a')|P_n\right) \le K_2(1+\|\Delta_n\|_w)^2. \tag{19}$$

By the Cauchy-Schwarz inequality (18) and (19) imply Assumption (iv). To establish (19), we show that the Q-functions $Q_n^{(i)}$ are bounded for any n and any i. Such results are well-known for classical Q-learning, see (Gosavi, 2006). For our modified Q-learning, we can show it via a simple contradiction argument. Suppose for the sake of contradiction there is an index i such that $\lim_{n\to\infty}\|Q_n^{(i)}\| = \infty$. We first consider the setting, where there is an $s'\in\mathcal{S}$ and $a'\in\mathcal{A}$ such that $\lim_{n\to\infty} Q_n^{(i)}(s',a') = \infty$. So there exists $\bar{n}\in\mathbb{N}$ such that for all $n\ge\bar{n}$ we have $Q_n^{(j)}(s',a) \le Q_n^{(i)}(s',a')$ for all $a\in\mathcal{A}$ and for all $j=1,\dots,N$. Therefore, $\max_{\bar{a}\in\mathcal{A}}\widehat{Q}_{N,n}(s',\bar{a}) \le Q_n^{(i)}(s',a') \le \max_{\bar{a}\in\mathcal{A}} Q_n^{(i)}(s',\bar{a})$, which implies

$$\begin{aligned} Q_{n+1}^{(i)}(s,a) &= Q_n^{(i)}(s,a) + \alpha_n(r(s,a) + \gamma\max_{a'\in\mathcal{A}}\widehat{Q}_{N,n}(s',a') - Q_n^{(i)}(s,a)) &\text{(20a)}\\ &\le Q_n^{(i)}(s,a) + \alpha_n(r(s,a) + \gamma\max_{a'\in\mathcal{A}} Q_n^{(i)}(s',a') - Q_n^{(i)}(s,a)), &\text{(20b)}\end{aligned}$$

for any $s\in\mathcal{S}$ and $a\in\mathcal{A}$. When considering $S_n = s$, $A_n = a$ and $S_{n+1} = s'$, we see that the upper bound (20b) is Watkins' Q-learning which leads to a bounded Q-function, so $Q_{n+1}^{(i)}(s,a)$ needs to be bounded for all s,a which is a contradiction. The case where there is an $s'\in\mathcal{S}$ and $a'\in\mathcal{A}$ such that $\lim_{n\to\infty} Q_n^{(i)}(s',a') = -\infty$ follows analogously. Consequently, the Q-functions $Q_n^{(i)}$ for any n and any i are bounded and (19) indeed holds, which implies Assumption (iv). □

A.2. Estimation Bias

Proof of Theorem 3.2 (Estimation Bias). To prove Assertion (i), note that according to the definition of the robust estimator (8) for any $\mathbb{E}[\theta_n^{(i)}] \in \mathcal{B}_\rho(\widehat{\theta}_{N,n})$ it must hold that

$$\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n}) \le \gamma\phi(x)\max_{a'\in\mathcal{A}}\phi(s',a')^\top\mathbb{E}[\theta_n^{(i)}] = \gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] \quad \forall x\in\mathcal{X}, s'\in\mathcal{S},$$

which implies

$$\mathbb{E}[\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n})] \le \gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] \quad \forall x\in\mathcal{X}, s'\in\mathcal{S}. \tag{21}$$

A lower bound can be derived as for all $x\in\mathcal{X}$, $s'\in\mathcal{S}$

$$\begin{aligned} \mathbb{E}[\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n})] &= \gamma\phi(x)\mathbb{E}\left[\max_{a'\in\mathcal{A}}\left\{\phi(s',a')^\top\widehat{\theta}_{N,n} - \sqrt{\rho}\|\phi(s',a')\|_2\right\}\right] &\text{(22a)}\\ &\ge \gamma\phi(x)\mathbb{E}\left[\max_{a'\in\mathcal{A}}\phi(s',a')^\top\widehat{\theta}_{N,n}\right] - \sqrt{\rho}\gamma\phi(x)\max_{a'\in\mathcal{A}}\|\phi(s',a')\|_2 &\text{(22b)}\\ &\ge \gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}\left[\phi(s',a')^\top\widehat{\theta}_{N,n}\right] - \sqrt{\rho}\gamma\phi(x)\max_{a'\in\mathcal{A}}\|\phi(s',a')\|_2 &\text{(22c)}\\ &= \gamma\phi(x)\max_{a'\in\mathcal{A}}\mathbb{E}\left[\phi(s',a')^\top\theta_n^{(i)}\right] - \sqrt{\rho}\gamma\phi(x)\max_{a'\in\mathcal{A}}\|\phi(s',a')\|_2, &\text{(22d)}\end{aligned}$$

where the first equality is due to Lemma 3.1. The first inequality follows from splitting the maximization or reverse triangle inequality. The second inequality follows from a Jensen step as explained in (6). The second equality uses the fact that all $\theta_n^{(i)}$ follow the same distribution. Combining (21) and (22) implies Assertion (i).

To prove Assertion (ii), we first claim that for any $n\in\mathbb{N}$

$$\lim_{N\to\infty}\widehat{\theta}_{N,n} = \mathbb{E}[\theta_n^{(i)}], \tag{23}$$

where the convergence is in mean square, i.e., $\lim_{N\to\infty}\mathbb{E}[\|\widehat{\theta}_{N,n} - \mathbb{E}[\theta_n^{(i)}]\|^2] = 0$. This in particular implies that $\widehat{\theta}_{N,n}$ converges to $\mathbb{E}[\theta^{(i)}]$ in distribution. Our second claim states that for any $x\in\mathcal{X}$ and $s'\in\mathcal{S}$, the function $\mathcal{E}_\rho(x,s',\theta')$ is

uniformly continuous in θ'. Equipped with these two claims,

$$\lim_{N\to\infty} \mathbb{E}[\mathcal{E}_\rho(x,s',\widehat{\theta}_{N,n})] = \mathbb{E}[\mathcal{E}_\rho(x,s',\mathbb{E}[\theta_n^{(i)}])] \tag{24a}$$

$$= \mathcal{E}_\rho(x,s',\mathbb{E}[\theta_n^{(i)}]) \tag{24b}$$

$$= \gamma\phi(x)\max_{a'\in\mathcal{A}}\left\{\mathbb{E}[\phi(s',a')^\top\theta_n^{(i)}] - \sqrt{\rho}\|\phi(s',a')\|_2\right\}, \tag{24c}$$

where the first equality holds due to the Portmanteau Theorem (Billingsley, 1999, Theorem 2.1), since $\widehat{\theta}_{N,n}$ converges to $\mathbb{E}[\theta_n^{(i)}]$ in distribution and the function $\mathcal{E}_\rho(x,s',\theta')$ is uniformly continuous in θ'. The second equality is true as the function $\mathcal{E}_\rho$ is deterministic, and the last equality is implied by Lemma 3.1.

It, therefore, remains to show the two claims. To show (23), we recall that by definition

$$\widehat{\theta}_{N,n} = \frac{1}{N}\sum_{i=1}^{N}\theta_n^{(i)}.$$

By symmetry of the 2RA Q-learning (7) the variables $\theta_n^{(i)}$ for all $i \in \{1,\dots,N\}$ have the same distribution, but are correlated. We can exploit the weak correlations via the following result.

Lemma A.2. *(Brockwell, 1991, Theorem 7.1.1) Let $\{Y^{(i)}\}$ be a stationary process with mean μ and autocovariance function $\gamma(\cdot)$ defined as $\gamma(N) = \mathsf{cov}(Y^{(i+N)}, Y^{(i)})$ for any $i \in \mathbb{N}$. Then,*

$$\lim_{N\to\infty}\mathbb{E}\left[\left(\frac{1}{N}\sum_{i=0}^{N-1}Y^{(i)} - \mu\right)^2\right] = 0 \quad \textit{if} \quad \lim_{N\to\infty}\gamma(N) = 0.$$

The 2RA Q-learning (7) is given as

$$\begin{aligned}\theta_{n+1}^{(i)} = (1-\alpha_n\beta_n^{(i)}A_1(X_n))\theta_n^{(i)} + \gamma\alpha_n\beta_n^{(i)}\phi(X_n)\max_{a'\in\mathcal{A}}\left\{\phi(S_{n+1},a')^\top\widehat{\theta}_{N,n} - \sqrt{\rho}\|\phi(S_{n+1},a')\|_2\right\}\\ + \alpha_n\beta_n^{(i)}b(X_n), \quad i = 1,\dots,N.\end{aligned} \tag{25}$$

We claim that for any $s,t \in \{1,2,\dots,N\}$ and for any $n \in \mathbb{N}_0$

$$\mathsf{cov}(\theta_n^{(s)},\theta_n^{(t)}) \le C\cdot\mathcal{O}(1/N), \tag{26}$$

where $C \in \mathbb{R}^{d\times d}$ is some constant matrix. Then, according to Lemma A.2, we obtain (23). It, therefore, remains to show (26), which we do by induction over n.

The initial condition $\theta_0^{(i)} = \theta_0$ is assumed to be some deterministic value for all i, then by using the update rule (25) and recalling that $\widehat{\theta}_{N,0} = \theta_0$

$$\begin{aligned}\mathsf{cov}(\theta_1^{(s)},\theta_1^{(t)}) &= \mathsf{cov}(v+\beta_1^{(s)}w, v+\beta_1^{(t)}w)\\ &= \mathsf{cov}(\beta_1^{(s)}w,\beta_1^{(t)}w)\\ &= \mathbb{E}[\beta_1^{(s)}\beta_1^{(t)}ww^\top] - \mathbb{E}[\beta_1^{(s)}w]\mathbb{E}[\beta_1^{(t)}w]\\ &= \mathbb{E}[\beta_1^{(s)}\beta_1^{(t)}]\mathbb{E}[ww^\top] - \mathbb{E}[\beta_1^{(s)}]\mathbb{E}[w]\mathbb{E}[\beta_1^{(s)}]\mathbb{E}[w]^\top\\ &= -\frac{1}{N^2}\mathbb{E}[w]\mathbb{E}[w]^\top\\ &\le C\cdot\mathcal{O}(1/N),\end{aligned}$$

where $v = \theta_0$, $w = \alpha_n\big(-A_1(X_0)\theta_0 + \gamma\phi(X_0)\max_{a'\in\mathcal{A}}\{\phi(S_1,a')^\top\theta_0 - \sqrt{\rho}\|\phi(S_1,a')\|_2\} + b(X_0)\big)$ and we use the fact that $\mathbb{E}[\beta_1^{(s)}\beta_1^{(t)}] = 0$ and $\mathbb{E}[\beta_1^{(s)}] = \frac{1}{N}$. Moreover, we use $C = \mathbb{E}[w]\mathbb{E}[w]^\top$. To proceed with the induction step, assume that for any $k,\ell \in \{1,2,\dots,N\}$ we have $\mathsf{cov}(\theta_n^{(k)},\theta_n^{(\ell)}) = C\cdot\mathcal{O}(1/N)$ and show that for any $k,\ell \in \{1,2,\dots,N\}$

$$\mathsf{cov}(\theta_{n+1}^{(k)},\theta_{n+1}^{(\ell)}) \le C\cdot\mathcal{O}(1/N). \tag{27}$$

Applying the update rule (25) leads to

$$\begin{aligned}
&\mathsf{cov}(\theta_{n+1}^{(k)}, \theta_{n+1}^{(\ell)}) \\
&\quad= \mathsf{cov}\Big((1-\alpha_n\beta_n^{(k)}A_1(X_n))\theta_n^{(k)} + \beta_n^{(k)}\alpha_n\Big(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\Big), \\
&\qquad\quad (1-\alpha_n\beta_n^{(\ell)}A_1(X_n))\theta_n^{(\ell)} + \beta_n^{(\ell)}\alpha_n\Big(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\Big)\Big) \\
&\quad= \mathsf{cov}\left((1-\alpha_n\beta_n^{(k)}A_1(X_n))\theta_n^{(k)}, (1-\alpha_n\beta_n^{(\ell)}A_1(X_n))\theta_n^{(\ell)}\right) \\
&\qquad+ \mathsf{cov}\Big((1-\alpha_n\beta_n^{(k)}A_1(X_n))\theta_n^{(k)}, \beta_n^{(\ell)}\alpha_n\left(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\right)\Big) \\
&\qquad+ \mathsf{cov}\Big((1-\alpha_n\beta_n^{(\ell)}A_1(X_n))\theta_n^{(\ell)}, \beta_n^{(k)}\alpha_n\left(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\right)\Big) \\
&\qquad+ \mathsf{cov}\Big(\beta_n^{(k)}\alpha_n\left(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\right), \beta_n^{(\ell)}\alpha_n\left(\gamma\phi(X_n)\mathcal{M}_n + b(X_n)\right)\Big),
\end{aligned} \tag{28}$$

where $\mathcal{M}_n = \max_{a'\in\mathcal{A}}\{\phi(S_{n+1},a')^\top\widehat{\theta}_{N,n} - \sqrt{\rho}\|\phi(S_{n+1},a')\|_2\}$. We can show that each covariance term from above is of the order $\mathcal{O}(1/N)$. For this, we recall that the properties of β_n and its independence with respect to X_n ensure that

$$\mathsf{cov}(\beta_n^{(\ell)}, \beta_n^{(k)}) = \mathcal{O}(1/N) \quad \text{and} \quad \mathsf{cov}(\beta_n^{(\ell)}, f(X_n)) = 0 \tag{29a}$$

for any bounded function $f : \mathcal{X} \to \mathbb{R}$. Moreover, for any $k \neq \ell$

$$\begin{aligned}
\mathsf{cov}(\beta_n^{(\ell)}f(X_n), \beta_n^{(k)}f(X_n)) &= \mathbb{E}[\beta_n^{(\ell)}\beta_n^{(k)}f(X_n)f(X_n)^\top] - \mathbb{E}[\beta_n^{(k)}f(X_n)]\mathbb{E}[\beta_n^{(\ell)}f(X_n)^\top] \\
&= -\frac{1}{N^2}\|\mathbb{E}[f(X_n)]\|_2^2 \leq \mathcal{O}(1/N).
\end{aligned} \tag{29b}$$

For any other bounded function $g : \mathcal{X} \to \mathbb{R}$, we get

$$\begin{aligned}
\mathsf{cov}(\beta_n^{(\ell)}f(X_n), g(X_n)) &= \mathbb{E}[\beta_n^{(\ell)}f(X_n)g(X_n)] - \mathbb{E}[\beta_n^{(\ell)}f(X_n)]\mathbb{E}[g(X_n)] \\
&= \frac{1}{N}\mathbb{E}[f(X_n)g(X_n)] - \frac{1}{N}\mathbb{E}[f(X_n)]\mathbb{E}[g(X_n)] = \mathcal{O}(1/N).
\end{aligned} \tag{29c}$$

Similarly, we obtain for any $s, t \in \{1, 2, \ldots, N\}$

$$\mathsf{cov}(\theta_n^{(k)}, \beta_n^{(\ell)}f(X_n)) = C \cdot \mathcal{O}(1/N), \tag{29d}$$

and

$$\mathsf{cov}(\theta_n^{(k)}, g(X_n)\widehat{\theta}_{N,n}) = \frac{1}{N}\sum_{\ell=1}^{N}\mathsf{cov}(\theta_n^{(k)}, g(X_n)\theta_n^{(\ell)}),$$

whereby by using the results of Bohrnstedt & Goldberger (1969), we can show that $\mathsf{cov}(\theta_n^{(k)}, g(X_n)\theta_n^{(\ell)}) = C \cdot \mathcal{O}(1/N)$. Therefore,

$$\mathsf{cov}(\theta_n^{(k)}, g(X_n)\widehat{\theta}_{N,n}) = C \cdot \mathcal{O}(1/N). \tag{29e}$$

Finally, equipped with (27) and (29) by exploiting the results of Bohrnstedt & Goldberger (1969) and continuing with (28), we show

$$\mathsf{cov}(\theta_{n+1}^{(k)}, \theta_{n+1}^{(\ell)}) = C \cdot \mathcal{O}(1/N),$$

which completes the induction step. We, therefore, have shown (26) and hence (23). Regarding our second claim, we show that for any $x \in \mathcal{X}$ and $s' \in \mathcal{S}$, the function $\mathcal{E}_\rho(x, s', \theta)$ is uniformly continuous in θ. For any fixed $x \in \mathcal{X}$ and $s' \in \mathcal{S}$ the the function $\mathcal{E}_\rho(x, s', \theta)$ can be expressed as

$$\mathcal{E}_\rho(x, s', \theta) = \gamma\phi(x)\max_{a'\in\mathcal{A}}\left\{\phi(s',a')^\top\theta - \sqrt{\rho}\|\phi(s',a')\|_2\right\} = \gamma\phi(x)\varphi(\theta, s'),$$

where $\varphi(\theta, s') = \max_{a' \in \mathcal{A}} \{\phi(s', a')^\top \theta - \sqrt{\rho}\|\phi(s', a')\|_2\}$. To show that $\mathcal{E}_\rho(x, s', \theta)$ is uniformly continuous in θ, it remains to show that φ is uniformly continuous in θ. Clearly, φ is Lipschtiz continuous in θ, as $\max\{\cdot\} - \max\{\cdot\} \leq \max\{\cdot - \cdot\}$, which is the reversed triangle inequality for the ∞-norm. This implies the desired uniform continuity and hence hence $\mathcal{E}_\rho(x, s', \theta)$ is uniformly continuous in θ. Having shown both claims completes the proof. □

Proof of Corollay 3.1. Recall that according to Theorem 3.1 for any $i \in \{1, \dots, N\}$ we have $\lim_{n\to\infty} \theta_n^{(i)} = \theta^\star$ almost surely and accordingly $\lim_{n\to\infty} \widehat{\theta}_{N,n} = \theta^\star$ almost surely. By the definition of $\mathcal{E}_{\rho_n}$ in (8)

$$\gamma\phi(x) \max_{a' \in \mathcal{A}} \mathbb{E}[\phi(s', a')^\top \theta^\star] = \mathbb{E}[\mathcal{E}_0(x, s', \theta^\star)] \quad \forall x \in \mathcal{X}, s' \in \mathcal{S}. \tag{30}$$

Therefore, for all $x \in \mathcal{X}, s' \in \mathcal{S}$

$$\begin{aligned}
\lim_{n\to\infty} \mathbb{E}\left[\mathcal{E}_{\rho_n}(x, s', \widehat{\theta}_{N,n})\right] &= \mathbb{E}\left[\lim_{n\to\infty} \mathcal{E}_{\rho_n}(x, s', \widehat{\theta}_{N,n})\right] && \text{(31a)}\\
&= \mathbb{E}\left[\lim_{n\to\infty} \gamma\phi(x) \max_{a' \in \mathcal{A}} \left\{\phi(s', a')^\top \widehat{\theta}_{N,n} - \sqrt{\rho_n}\|\phi(s', a')\|_2\right\}\right] && \text{(31b)}\\
&= \mathbb{E}\left[\gamma\phi(x) \max_{a' \in \mathcal{A}} \left\{\phi(s', a')^\top \theta^\star\right\}\right] && \text{(31c)}\\
&= \mathbb{E}\left[\mathcal{E}_0(x, s', \theta^\star)\right] && \text{(31d)}\\
&= \gamma\phi(x) \max_{a' \in \mathcal{A}} \mathbb{E}[\phi(s', a')^\top \theta^\star] && \text{(31e)}\\
&= \gamma\phi(x) \max_{a' \in \mathcal{A}} \lim_{n\to\infty} \mathbb{E}\left[\phi(s', a')^\top \theta_n^{(i)}\right] && \text{(31f)}\\
&= \lim_{n\to\infty} \gamma\phi(x) \max_{a' \in \mathcal{A}} \mathbb{E}\left[\phi(s', a')^\top \theta_n^{(i)}\right], && \text{(31g)}
\end{aligned}$$

where the equality (31a) follows from the bounded convergence theorem. The equality (31b) is due to Lemma 3.1. In (31c), we use the fact that the limit and maximum can be interchanged as the maximum is over a finite set and that $\widehat{\theta}_{N,n}$ converges to $\theta^\star$ due to Theorem 3.1. The step (31d) uses the definition of $\mathcal{E}_0$. The equality (31e) is due to (30) and (31f) uses that $\theta_n^{(i)}$ converges to $\theta^\star$ due to Theorem 3.1 together with the bounded convergence theorem. Finally, (31g) uses again the fact that the limit and maximum can be interchanged as the maximum is over a finite set. □

A.3. Asymptotic Mean-Squared Error

Proof of Theorem 3.3 (AMSE for 2RA Q-learning). Our proof is inspired by the recent treatment of Double Q-learning (Weng et al., 2020) and by Devraj & Meyn (2017) analyzing the asymptotic properties of Q-learning. The key idea is to recall that from the proof of Theorem 3.1, we know that $\theta_n^{(i)} \to \theta^\star = 0$ as $n \to \infty$ for any $i \in \{1, 2, \dots, N\}$. Hence, we can express the AMSE of $\theta^{(i)}$ alternatively as

$$\mathsf{AMSE}(\theta^{(i)}) = \lim_{n\to\infty} n\mathbb{E}[\theta_n^{(i)\top} \theta_n^{(i)}] = \operatorname{tr}\left(\lim_{n\to\infty} n\mathbb{E}[\theta_n^{(i)} \theta_n^{(i)\top}]\right) = \operatorname{tr}(V),$$

where the matrix $V = \lim_{n\to\infty} n\mathbb{E}[\theta_n^{(i)} \theta_n^{(i)\top}]$ is called the *asymptotic covariance*. It has been shown in Devraj & Meyn (2017) that the asymptotic covariance of Watkins' Q-learning (5) can be studied via the linearized counterpart, given as

$$\theta_{n+1}^{\mathsf{QL}} = \theta_n^{\mathsf{QL}} + \alpha_n^{\mathsf{QL}} \phi(X_n)(r(X_n) + \gamma\phi(S_{n+1}, \pi^\star(S_{n+1}))^\top \theta_n^{\mathsf{QL}} - \phi(X_n)^\top \theta_n^{\mathsf{QL}}),$$

where $\pi^\star$ is the optimal policy based on $\theta^\star$. Using similar arguments from Devraj & Meyn (2017) and Weng et al. (2020), we can show that the asymptotic variance of 2RA Q-learning (7), which is defined as $\lim_{n\to\infty} n\mathbb{E}[\widehat{\theta}_{N,n} \widehat{\theta}_{N,n}^\top]$, can be studied by considering the linearized recursion with $\rho_n = 0$, given as

$$\theta_{n+1}^{(i)} = \theta_n^{(i)} + \alpha_n \beta_n^{(i)} (b(X_n) - A_1(X_n)\theta_n^{(i)} + \underbrace{\gamma\phi(X_n)\phi(S_{n+1}, \pi^\star(S_{n+1}))^\top}_{=A_2(Z_n)} \widehat{\theta}_{N,n}), \quad i = 1, \dots, N, \tag{32}$$

where $Z_n = (X_n, S_{n+1})$, $b(X_n) = \phi(X_n)r(X_n)$ and $A_1(X_n) = \phi(X_n)\phi(X_n)^\top$. We formally justify this linearization argument in Lemma B.1. Using a more compact notation where $\theta_n = (\theta_n^{(1)\top}, \dots, \theta_n^{(N)\top})^\top$ and $\beta_n = (\beta_n^{(1)}, \dots, \beta_n^{(N)})^\top$ and choosing the learning rate $\alpha_n = \alpha_n^{\mathsf{QL}} \cdot N$, the update equation (32) can be expressed in a standard form as

$$\theta_{n+1} = \theta_n + \alpha_n^{\mathsf{QL}}(\mathsf{b}(X_n) + \mathsf{A}_2(Z_n)\theta_n - \mathsf{A}_1(X_n)\theta_n), \tag{33}$$

where

$$\begin{aligned}
&\mathsf{A}_1(X_n) = N \cdot \mathsf{diag}(\beta_n^{(1)} A_1(X_n), \dots, \beta_n^{(N)} A_1(X_n)),\\
&\mathsf{A}_2(Z_n) = \begin{pmatrix} \beta_n^{(1)} A_2(Z_n) & \dots & \beta_n^{(1)} A_2(Z_n) \\ \beta_n^{(2)} A_2(Z_n) & \dots & \beta_n^{(2)} A_2(Z_n) \\ \vdots & \ddots & \vdots \\ \beta_n^{(N)} A_2(Z_n) & \dots & \beta_n^{(N)} A_2(Z_n) \end{pmatrix}, \qquad \mathsf{b}(X_n) = \begin{pmatrix} N\beta_n^{(1)} \cdot b(X_n) \\ \vdots \\ N\beta_n^{(N)} \cdot b(X_n) \end{pmatrix}.
\end{aligned} \tag{34}$$

Let μ denote the invariant distribution of the Markov chain $\{X_n\}_{n\in\mathbb{N}}$ and let D be a diagonal matrix with entries $D_{ii} = \mu_i$ for all $i = 1, \dots, |\mathcal{X}|$. Then, when considering X_∞ as a random variable under the stationary distribution, we introduce

$$\begin{aligned}
&\bar{A}_1 = \mathbb{E}[A_1(X_\infty)] = \Phi D \Phi^\top, \qquad && \bar{A}_2 = \mathbb{E}[A_2(Z_\infty)] = \gamma \Phi D P S_{\pi^\star} \Phi^\top,\\
&\bar{\mathsf{A}}_1 = \mathbb{E}[\mathsf{A}_1(X_\infty)], && \bar{\mathsf{A}}_2 = \mathbb{E}[\mathsf{A}_2(Z_\infty)],
\end{aligned} \tag{35}$$

where $S_{\pi^\star}$ is the action selection matrix of the optimal policy $\pi^\star$ such that $S_{\pi^\star}(s, (s, \pi^\star(s))) = 1$ for $s \in \mathcal{S}$ and Φ is defined in (4). With these variables at hand, we define $\bar{\mathsf{A}} = \bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1$, i.e.,

$$\bar{\mathsf{A}} = \begin{pmatrix} \frac{1}{N}\bar{A}_2 - \bar{A}_1 & \frac{1}{N}\bar{A}_2 & \frac{1}{N}\bar{A}_2 & \dots & \frac{1}{N}\bar{A}_2 \\ \frac{1}{N}\bar{A}_2 & \frac{1}{N}\bar{A}_2 - \bar{A}_1 & \frac{1}{N}\bar{A}_2 & \dots & \frac{1}{N}\bar{A}_2 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \frac{1}{N}\bar{A}_2 & \frac{1}{N}\bar{A}_2 & \frac{1}{N}\bar{A}_2 & \dots & \frac{1}{N}\bar{A}_2 - \bar{A}_1 \end{pmatrix}.$$

Moreover, we introduce

$$\Sigma_{\mathsf{b}} = \mathbb{E}[\mathsf{b}(X_0)\mathsf{b}(X_0)^\top] + \sum_{n\geq 1} \mathbb{E}[\mathsf{b}(X_n)\mathsf{b}(X_0)^\top + \mathsf{b}(X_0)\mathsf{b}(X_n)^\top].$$

According to the definition of b, we get the block-diagonal structure

$$\mathbb{E}[\mathsf{b}(X_0)\mathsf{b}(X_0)^\top] = N \cdot \mathsf{diag}(\mathbb{E}[b(X_0)b(X_0)^\top], \dots, \mathbb{E}[b(X_0)b(X_0)^\top]),$$

where the expectation is in steady-state. Moreover,

$$\mathbb{E}[\mathsf{b}(X_n)\mathsf{b}(X_0)^\top] = \begin{pmatrix} \mathbb{E}[b(X_n)b(X_0)^\top] & \dots & \mathbb{E}[b(X_n)b(X_0)^\top] \\ \vdots & & \vdots \\ \mathbb{E}[b(X_n)b(X_0)^\top] & \dots & \mathbb{E}[b(X_n)b(X_0)^\top] \end{pmatrix},$$

which eventually leads to a matrix of the form

$$\Sigma_{\mathsf{b}} = \begin{pmatrix} N\mathbb{E}[b(X_0)b(X_0)^\top] + 2B_2 & 2B_2 & \dots & 2B_2 \\ 2B_2 & N\mathbb{E}[b(X_0)b(X_0)^\top] + 2B_2 & \dots & 2B_2 \\ \vdots & \vdots & \ddots & \vdots \\ 2B_2 & 2B_2 & \dots & N\mathbb{E}[b(X_0)b(X_0)^\top] + 2B_2 \end{pmatrix}, \tag{36}$$

where we introduce the variables $B_2 = \frac{1}{2}\sum_{n\geq 1} \mathbb{E}[b(X_n)b(X_0)^\top + b(X_0)b(X_n)^\top]$ and $B_1 = \mathbb{E}[b(X_0)b(X_0)^\top] + B_2$.

We define $g_0 = \inf\{g \geq 0 \,:\, g\max\{\lambda_{\max}(\bar{A}), \lambda_{\max}(\bar{\mathsf{A}})\} < -1\}$ and note that g_0 exists, since both $\bar{A}$ and $\bar{\mathsf{A}}$ are Hurwitz as the corresponding Q-learning variants converge (Theorem 3.1). As a result for any $g > g_0$ the matrix $\frac{1}{2}I + g\bar{\mathsf{A}}$ is Hurwitz and hence the Lyapunov equation

$$\Sigma_\infty \left(\frac{1}{2}I + g\bar{\mathsf{A}}^\top\right) + \left(\frac{1}{2}I + g\bar{\mathsf{A}}\right)\Sigma_\infty + g^2\Sigma_{\mathsf{b}} = 0 \tag{37}$$

has a unique solution, which describes AMSE of our proposed method (see Chen et al. (2020) and Weng et al. (2020, Theorem 1)), i.e., $\Sigma_\infty = \lim_{n\to\infty} n\mathbb{E}[\theta_n\theta_n^\top]$. Due to the symmetry of the proposed scheme, the matrix Σ_∞ will consist of diagonal elements equal to $V = \lim_{n\to\infty} n\mathbb{E}[\theta_n^{(i)}\theta_n^{(i)\top}]$ and off-diagonal entries $C = \lim_{n\to\infty} n\mathbb{E}[\theta_n^{(i)}\theta_n^{(j)\top}]$ for $i \neq j$. Therefore, summing the first row of matrices in (37) and using (36) leads to

$$V + (N-1)C + g(V + (N-1)C)(\bar{A}_2 - \bar{A}_1)^\top + g(\bar{A}_2 - \bar{A}_1)(V + (N-1)C) + g^2N(B_1 + B_2) = 0. \tag{38}$$

Due to the definition of g_0, the matrix $\frac{1}{2}I + g\bar{A}$ is Hurwitz and hence the Lyapunov equation

$$\Sigma_\infty^{\mathsf{QL}}\left(\frac{1}{2}I + g(\bar{A}_2 - \bar{A}_1)^\top\right) + \left(\frac{1}{2}I + g(\bar{A}_2 - \bar{A}_1)\right)\Sigma_\infty^{\mathsf{QL}} + g^2\left(B_1 + B_2\right) = 0 \tag{39}$$

has a unique solution, which is denoted by $\Sigma_\infty^{\mathsf{QL}}$ and describes the AMSE of Watkins' Q-learning, i.e., $\Sigma_\infty^{\mathsf{QL}} = \lim_{n\to\infty} \mathbb{E}[\theta_n^{\mathsf{QL}}\theta_n^{\mathsf{QL}\top}]$, see Weng et al. (2020).

By comparing (39) with (38) and recalling that the solutions are unique, we obtain $\Sigma_\infty^{\mathsf{QL}} = \frac{V+(N-1)C}{N}$. Finally,

$$\begin{aligned}
\mathsf{AMSE}(\widehat{\theta}_N) &= \lim_{n\to\infty} n\mathbb{E}\left[\left(\frac{1}{N}\sum_{j=1}^N \theta_n^{(j)}\right)^\top \left(\frac{1}{N}\sum_{i=1}^N \theta_n^{(i)}\right)\right] && \text{(40a)}\\
&= \frac{1}{N^2}\lim_{n\to\infty} n\mathbb{E}\left[\left(\sum_{j=1}^N \theta_n^{(j)\top}\right)\left(\sum_{i=1}^N \theta_n^{(i)}\right)\right] && \text{(40b)}\\
&= \frac{1}{N^2}\left(N\mathrm{tr}\left(V\right) + N(N-1)\mathrm{tr}\left(C\right)\right) && \text{(40c)}\\
&= \mathrm{tr}\left(\frac{V + (N-1)C}{N}\right) && \text{(40d)}\\
&= \mathrm{tr}\left(\Sigma_\infty^{\mathsf{QL}}\right) && \text{(40e)}\\
&= \mathsf{AMSE}(\theta^{\mathsf{QL}}). && \text{(40f)}
\end{aligned}$$

□

B. Linearization results

The proof of Theorem 3.3 uses a key result, stating that the asymptotic mean-squared error of the proposed 2RA Q-learning method (7) can be alternatively characterized by the linearized recursion (32). This result is a modification of the analysis for Double Q-learning derived in (Weng et al., 2020, Appendix A). To derive a formal statement, we introduce the key tool to analyze the linearization (32), which is the ODE counterpart of the 2RA Q-learning scheme (7) given as

$$\dot{\theta}^{(i)}(t) = \lim_{n\to\infty} g\mathbb{E}[\phi(X_n)\big(r(X_n) - \phi(X_n)^\top \theta^{(i)}(t)\big) + \gamma\phi(X_n)(\max_{a'} \phi(S_{n+1}, a')\widehat{\theta}_N(t))], \tag{41}$$

where $\widehat{\theta}_N(t) = \frac{1}{N}\sum_{i=1}^N \theta^{(i)}(t)$ and where we have used $\lim_{n\to\infty}\rho_n = 0$. With the help of the ODE (41) we can justify why working with the linearized system (32) allows us to quantify the AMSE for the 2RA Q-learning (7). This justification requires the following Assumption. We use the vectorized notation $\theta = ({\theta^{(1)}}^\top, \dots, {\theta^{(N)}}^\top)^\top$ and $\theta^\star = ({\theta^\star}^\top, \dots, {\theta^\star}^\top)^\top$ where $\theta^\star \in \mathbb{R}^d$ is the optimal solution to the underlying MDP.

Assumption B.1 (Linearization)**.** *We stipulate that for any $N \in \mathbb{N}$*

(i) The process $\theta^{(i)}(t)$ described by the ODE (41) has a globally asymptotically stable equilibrium $\bar{\theta}^{(i)}$ for any $i = 1, \dots, N$.

(ii) The optimal policy $\pi^\star$ of the underlying MDP is unique.

(iii) The sequence of random variables $\{n\|\theta_n - \theta^\star\|_2^2, n \geq 1\}$ is uniformly integrable.

Sufficient conditions for Assumption (i), when using linear function approximation and in the setting $N = 1$, are studied in Melo et al. (2008); Lee & He (2020). Assumption (ii) is standard in many theoretical treatments of Q-learning and Assumption (iii) for $N = 1$ has been established, see Durrett (2010, Theorem 5.5.2), Devraj & Meyn (2017).

Lemma B.1 (Linearization)**.** *Let $\{\theta_n\}_{n\in\mathbb{N}}$ be a sequence generated by the 2RA Q-learning (7) and let $\{\bar{\theta}_n\}_{n\in\mathbb{N}}$ be a sequence generated by its linearized counterpart (32). Under Assumption B.1, we have*

$$\lim_{n\to\infty} n\mathbb{E}[\|\theta_n - \theta^\star\|_2^2] = \lim_{n\to\infty} n\mathbb{E}[\|\bar{\theta}_n - \theta^\star\|_2^2].$$

Proof. In a first step, we show that the ODE (41) for any $i = 1, \dots, N$ has a unique globally asymptotically stable equilibrium given as $\bar{\theta}^{(i)} = \theta^\star$, where $\theta^\star$ is the the limit of Watkins' Q-learning (Weng et al., 2020, Equation (22)). Note that by Assumption B.1(i), the ODE (41) has a unique globally asymptotically stable equilibrium that we denote as $\bar{\theta}^{(i)}$ for all $i = 1, \dots, N$. By symmetry, any perturbation of it is a globally asymptotically equilibrium too. Hence, by uniqueness we must have $\bar{\theta}^{(i)} = \bar{\theta}^{(j)}$ for all $i, j \in \{1, \dots, N\}$. Since all equilibrium points are identical, we recover the equilibrium point of the ODE describing Watkins' Q-learning, i.e., $\bar{\theta}^{(i)} = \theta^\star$ for all $i = 1, \dots, N$.

Next, in order to apply (Weng et al., 2020, Theorem 3) with respect to the ODE (41) we define for any $i = 1, \dots, N$

$$w^{(i)}(\theta(t)) = \lim_{n\to\infty} g\mathbb{E}[\phi(X_n)\big(r(X_n) - \phi(X_n)^\top \theta^{(i)}(t)\big) + \gamma\phi(X_n)(\max_{a'} \phi(S_{n+1}, a')\widehat{\theta}_N(t))].$$

With the vector notation $w(\theta) = (w^{(1)}(\theta)^\top, \dots, w^{(N)}(\theta)^\top)^\top$ and by plugging in the globally asymptotically stable equilibrium from above we obtain

$$w(\theta^\star) = g\bar{\mathsf{b}} + g(\bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1)\theta^\star, \tag{42}$$

where $\bar{\mathsf{b}}$, $\bar{\mathsf{A}}_1$ and $\bar{\mathsf{A}}_2$ have been introduced in (35). Note that (42) corresponds to the ODE of the linearized 2RA Q-learning (32) at the point $\theta^\star$. We aim to show that $\nabla_\theta w(\theta^\star) = g(\bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1)$. This result follows from observing that there exists $\varepsilon > 0$ such that for any θ such that $\|\theta - \theta^\star\|_\infty \leq \varepsilon$ it holds $w^{(i)}(\theta) = g\bar{\mathsf{b}} + g(\bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1)\theta$. To see why this is the case, by following Weng et al. (2020), note that for the optimal policy $\pi^\star$ corresponding to $\theta^\star$ we can define

$$\omega = \min_{(s,a)\in\mathcal{X}: a\neq\pi^\star(s)} \{\phi(s, \pi^\star(s))^\top\theta^\star - \pi(s,a)^\top\theta^\star\} > 0,$$

where the strict positivity follows from the uniqueness of $\pi^\star$. Choose $\varepsilon = \frac{\omega}{3\|\Phi\|_1}$ and consider any $\theta^{(i)} \in \mathbb{R}^d$ such that $\|\theta^{(i)} - \theta^\star\|_\infty \leq \varepsilon$. Then, for any $s \in \mathcal{S}$ and $a \in \mathcal{A}$ with $a \neq \pi^\star(s)$, it holds

$$\phi(s, \pi^\star(s))^\top\theta^{(i)} - \phi(s,a)^\top\theta^{(i)} \geq \phi(s, \pi^\star(s))^\top\theta^\star - \phi(s,a)^\top\theta^\star - 2\|\Phi^\top(\theta^{(i)} - \theta^\star)\|_\infty \geq \omega - \frac{2\omega}{3} > 0,$$

which implies $\pi^\star = \pi_\theta$, i.e., for any θ such that $\|\theta - \theta^\star\|_\infty \leq \varepsilon$, the corresponding greedy policy $\pi_{\theta^\star}$ is optimal. Since $\|\theta^{(i)} - \theta^\star\|_\infty \leq \|\theta - \theta^\star\|_\infty$, it indeed holds for any θ such that $\|\theta - \theta^\star\|_\infty \leq \varepsilon$ that $w^{(i)}(\theta) = g\bar{\mathsf{b}} + g(\bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1)\theta$. So we have shown

$$\nabla_\theta w(\theta^\star) = g(\bar{\mathsf{A}}_2 - \bar{\mathsf{A}}_1). \tag{43}$$

In the final step, we define

$$W^{(i)}(Z_n) = \mathbb{E}[\phi(X_n)\big(r(X_n) - \phi(X_n)^\top\theta^\star\big) + \gamma\phi(X_n)(\max_{a'}\phi(S_{n+1}, a')\theta^\star)],$$

with the notation from (34) and denote its vectorized version in a compact form as $\mathbf{W}(Z_n) = (W^{(1)}(Z_n)^\top, \ldots, W^{(N)}(Z_n)^\top)^\top$. We then note that

$$\mathbf{W}(Z_n) = \mathsf{b}(Z_n) + (\mathsf{A}_2(Z_n) - \mathsf{A}_1(Z_n))\theta^\star.$$

Then, by definition of the asymptotic covariance

$$\begin{aligned} C_\theta(\theta^\star) &= \sum_{n=-\infty}^{\infty} \mathbb{E}[(g\mathbf{W}(Z_n) - w(\theta^\star)(g\mathbf{W}(Z_1) - w(\theta^\star)^\top] \\ &= g^2 \sum_{n=-\infty}^{\infty} \mathbb{E}[\mathbf{W}(Z_n)\mathbf{W}(Z_1)^\top] \\ &= b^2\Sigma_{\mathsf{b}}. \end{aligned} \tag{44}$$

From the two results (43) and (44) by invoking (Weng et al., 2020, Theorem 3) we obtain

$$\sqrt{n}(\theta_n - \theta^\star) \xrightarrow{d} \mathcal{N}(0, \Sigma), \tag{45}$$

where Σ is the unique solution to the Lyapunov equation (37). By applying the Continuous Mapping Theorem to (45) we get

$$n\|\theta_n - \theta^\star\|_2^2 \xrightarrow{d} \|X\|_2^2, \quad X \sim \mathcal{N}(0, \Sigma). \tag{46}$$

Finally, combining (46) with Assumption B.1(iii) according to (Durrett, 2010, Theorem 5.5.2) ensures that

$$\lim_{n\to\infty} n\mathbb{E}[\|\theta_n - \theta^\star\|_2^2] = \mathbb{E}[\|X\|_2^2] = \operatorname{tr}(\Sigma).$$

When considering the linearization (7) instead of the 2RA Q-learning, by following the same lines without the need to linearize, we obtain

$$\lim_{n\to\infty} n\mathbb{E}[\|\bar{\theta}_n - \theta^\star\|_2^2] = \mathbb{E}[\|X\|_2^2] = \operatorname{tr}(\Sigma),$$

where again P is the unique solution to the Lyapunov equation (37). This completes the proof. □

C. Additional Numerical Results

C.1. Additional Plots for Random Environment Experiment

We provide some additional plots of the random experiment described in Section 4.

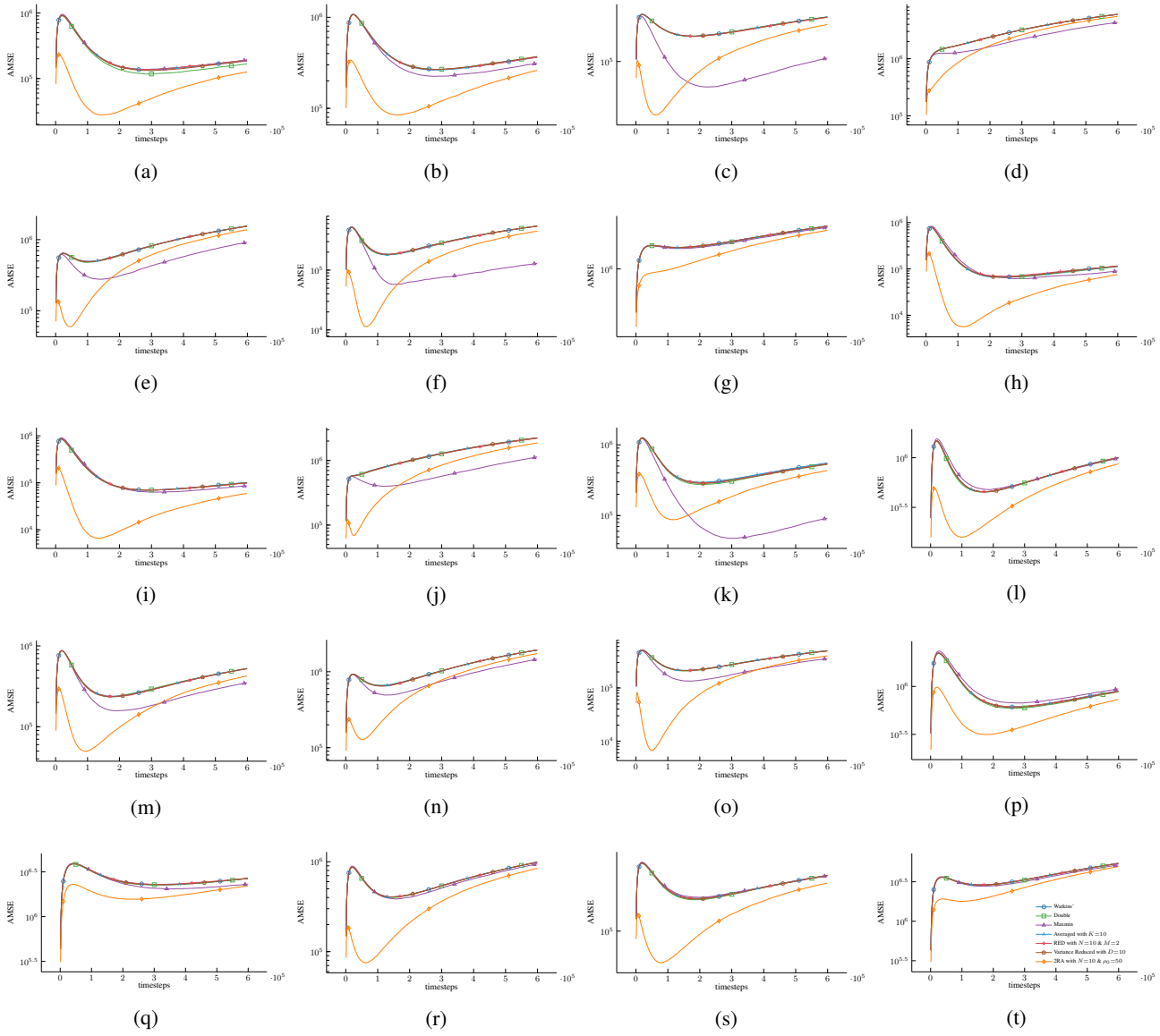
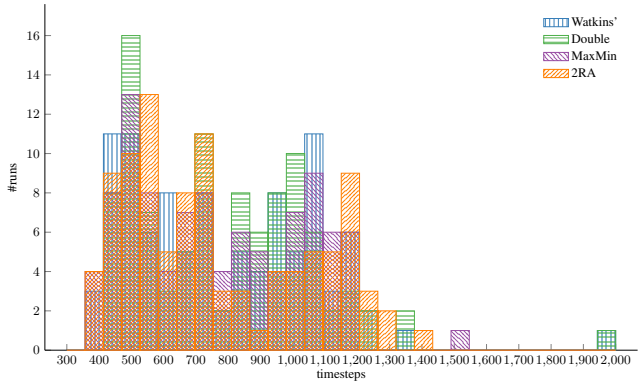


Figure 4: Random Environment. All methods use an initial learning rate of $\alpha_0 = 0.01$, $w_\alpha = 10^5$, $\gamma = 0.9$, and all $\theta^{(i)}$ initialized as zero. Maxmin as well as 2RA Q-learning have $N = 10$ and 2RA agents additionally use $\rho_0 = 50$ and $w_\rho = 10^4$. The plots show the first 20 randomly drawn environments.

C.2. Neural Network Function Approximation

As an additional experiment, to test 2RA Q-learning when used with neural network Q-function approximation implemented in Tensorflow (Abadi et al., 2015), the LunarLander environment (Brockman et al., 2016) is chosen. A lander receives a large positive reward for landing in a designated area, a large negative reward for crashing, and a small negative reward for firing a thruster. Similar to the CartPole experiment, the latest updated policy with ϵ-greedy exploration is used to generate the next timestep on which the model is updated since not all states may be reached by a random uniform policy. The environment is considered to be solved if the average reward over 100 episodes, during evaluation, reaches or exceeds 200.



(a) Distribution of hit times

Algorithm	Mean hit time
Watkins' Q-Learning	785 ± 281.38
Double Q-Learning	791 ± 268.64
Maxmin Q-Learning	770 ± 252.19
2RA Q-Learning	767 ± 276.24

(b) Mean and std of hit times

Figure 5: LunarLander, 100 experiments. All methods use a learning rate of $\alpha = 0.0002$ and a decay factor of $\gamma = 0.99$. Maxmin, as well as 2RA Q-learning, have $N = 5$. 2RA further uses $\rho_0 = 25$ and $w_\rho = 10^4$. All algorithms are evaluated every 50 episodes and recorded if the average evaluation reward reaches or exceeds 200. (a) Shows the distributions of each algorithm's hit times and (b) lists the respective mean hit times and corresponding standard deviations.

Analogue to the CartPole experiment, different algorithms are compared based on how many training episodes are required to solve the LunarLander environment where fewer average timesteps, until the environment is solved, result in a higher performance ranking for the corresponding model. Instead of the $\theta^{(i)}$ a small neural network with two hidden ReLU (He et al., 2015) layers and N sets of weights are used. Averaging operations that were performed on the θ^i in the linear function approximation scenarios are now performed on sets of weights for the neural network. All models are trained with a Huber loss (Huber, 1964) and the Adam optimizer (Kingma & Ba, 2015) using a learning rate of $\alpha_0 = 0.0002$. The different learning methods are implemented as plain and close to the theory as possible. Updates are therefore applied after every timestep and on that single timestep.

Comparing the hit times shows only little difference between all learning methods with Maxmin and 2RA Q-learning leading the field by a small margin. Future work could aim to implement and test the method with more contemporary training pipelines, such as the incorporation of experience replay etc., to analyse whether such optimizations amplify the differences in performance or just apply a uniform shift to all learning methods.