Should You Trust DQN?

Aditya Gopalan
Electrical Communication Engineering
Indian Institute of Science (IISc)
Bengaluru 560012, India
aditya@iisc.ac.in

Gugan Thoppe
Computer Science and Automation
Indian Institute of Science (IISc)
Bengaluru 560012, India
gthoppe@iisc.ac.in

Abstract

For a Reinforcement Learning (RL) algorithm to be practically useful, the policy it estimates in the limit must be superior to the initial guess, at least on average. In this work, we show that the widely used Deep Q-Network (DQN) fails to meet even this basic criterion, even when it gets to see all possible states and actions infinitely often (a condition that ensures tabular Q-learning's convergence to the optimal Q-value). Our work's key highlights are as follows. First, we numerically show that DQN generally has a non-trivial probability of producing a policy worse than the initial one. Second, we give a theoretical explanation for this behavior in the context of linear DQN, wherein we replace the neural network with a linear function approximation but retain DQN's other key ideas, such as experience replay, target network, and ϵ-greedy exploration. Our main result is that the tail behaviors of linear DQN are governed by invariant sets of a deterministic Differential Inclusion (DI), a set-valued generalization of a differential equation. Notably, we show that these invariant sets need not align with locally optimal policies, thus explaining DQN's pathological behaviors, such as convergence to sub-optimal policies and policy oscillation. We also provide a scenario where the limiting policy is always the worst. Our work addresses a longstanding gap in understanding the behaviors of Q-learning with function approximation and ϵ-greedy exploration.

1 Introduction

Deep Q-Network (DQN) [Mnih et al., 2015] is popular in Reinforcement Learning (RL) due to its groundbreaking success in mastering complex tasks, such as playing a video game. Notably, DQN has achieved human-level performance on a variety of Atari 2600 games, demonstrating its potential to learn and make decisions in environments with high-dimensional sensory inputs. This success has been attributed to four factors: i) *neural network* to reasonably approximate Q^*, the optimal Q-value function, for large state and action spaces, ii) *ϵ-greedy policy* to balance exploration and exploitation of optimal actions at different states, iii) *experience replay* to decouple the algorithm's sub-module that interacts with the environment from the one that updates Q^*'s estimate, and iv) a *target network* to stabilize training. In recent times, though, the DQN algorithm has also been reported to show several pathological behaviors (beyond the classical instability [Baird, 1995]) such as policy oscillation, i.e., alternating between two or more policies without end, and convergence to sub-optimal policies (including the worst) [Gordon, 1996, 2000, De Farias and Van Roy, 2000, Bertsekas, 2011, Young and Sutton, 2020]. In fact, Patterson et al. [2023] claim the following:

> "... we observed (rare) catastrophic failure events for DQN across nearly every tested domain ... In Lunar Lander, some agents would simply fly off into oblivion, obtaining incredible amounts of negative reward until the episode was mercifully terminated ... In Cliff World, DQN would get stuck in a corner perpetually in every single episode ... some agents would learn to jump into the cliff immediately to obtain massive negative rewards."

Workshop on Aligning Reinforcement Learning Experimentalists and Theorists (ARLET 2024).

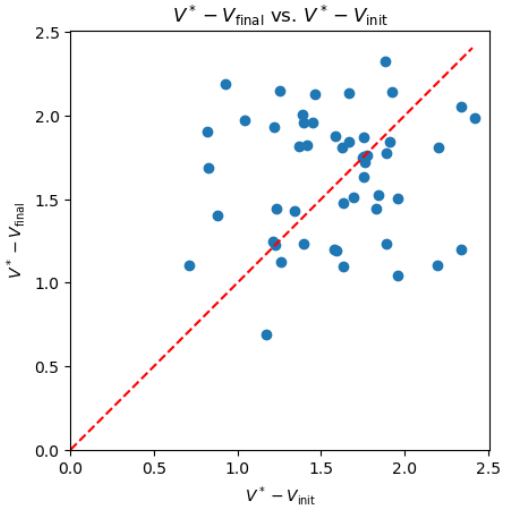


Figure 1: Scatterplot of initial (x) vs. final value-function suboptimality (y) for DQN run on randomly generated MDPs. To get this plot, we first generated a population of 10-state, 10-action MDPs (100 in all) by drawing each reward $r(s, a)$ and transition probability $\mathbb{P}(s'|s, a)$ independently and uniformly from the interval $[0, 1]$ (and then normalizing them). We then plotted each blue dot by a) picking a random MDP from this population, b) initializing DQN with a random Q-value network, c) finding the difference between V^* and the value of the greedy policy of the initial Q-network (to get the x-coordinate of the dot), d) running DQN for a fixed budget of iterations, and e) finding the value of the greedy policy of the final Q-network (to get the y-coordinate of the dot). The red dashed line is the diagonal $y = x$. We see that over 50% of the runs lead to a policy worse than the initial, represented by the dots above the diagonal.

These conflicting narratives lead us to the following three questions about DQN's behavior: 1) Does DQN ensure a monotonic improvement in the optimal policy estimate? 2) If not, does it at least ensure convergence to a locally optimal policy? 3) At the very least, is there any improvement over the initial policy? These questions have remained unresolved, even for basic Q-learning with linear function approximation and ϵ-greedy exploration. In fact, Problem 1 in 'Open Theoretical Questions in Reinforcement Learning' [Sutton, 1999] addresses the need to explain the peculiar behaviors observed in the closely related linear[1] SARSA algorithm with ϵ-greedy exploration:

> "... The parameters of the linear function can be shown to have no fixed point in the expected value. Yet neither do they diverge; they seem to 'chatter' in the neighborhood of a good policy [Bertsekas and Tsitsiklis, 1996]. This kind of solution can be completely satisfactory in practice, but can it be characterized theoretically? What can be assured about the quality of the chattering solution? New mathematical tools seem necessary."

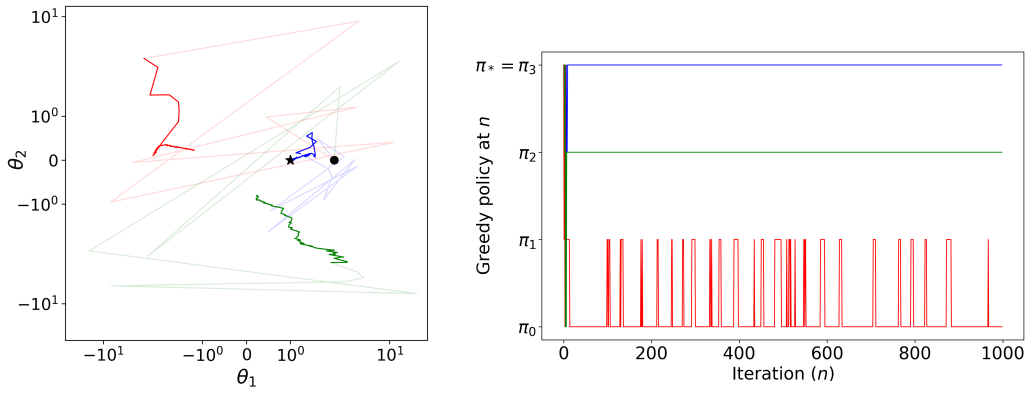
Similarly, for linear Q-learning, the following questions have been asked [Lu et al., 2021]: "Does (it) have a (fixed-point) solution? Does the solution (correspond) to a good policy?"

In this work, we provide both empirical and theoretical evidence to show that the answer to all the three questions above is an *emphatic no* in general. As our first evidence, we present Figure 1. It shows the change in the value function of the policies learned by DQN over single runs in randomly generated MDPs. As can be seen, over 50% of the runs result in DQN learning a policy that is worse than the initial guess; on $\approx 20\%$ on the runs, it is in fact significantly worse. While studies evaluating DQN's performance in specific MDPs such as Mujoco environments and Atari games are extensive, we believe ours is the first over a population of randomly generated MDPs. One may consider DQN to be a complex algorithm and, hence, attribute our observed performance simply to a poor tuning of hyperparameters such as the experience replay length, the target network refresh rate, and the stepsize schedule. The rest of our work shows that these behaviors are consequences of more fundamental issues with DQN's update and sampling rules themselves.

For an initial glimpse of these fundamental issues, look at Figure 2, which shows three runs of a 'linear-DQN' variant (see Section B for the implementation details). As in a standard DQN [Mnih et al., 2015], this variant also employs ϵ-greedy exploration, experience replay, and a target network. However, it uses a linear function instead of a neural-network-induced nonlinear function for approximating Q^*. The reduction in the approximation power is offset by including[2] Q^* in this linear function class. The starting conditions for all three runs are the same, ensuring that the initial policy is close to π_*. In this idealized setting, one would expect linear DQN to always find π_*. Surprisingly, we see three different behaviors: i) convergence to a sub-optimal policy (green), ii) oscillation between two sub-optimal policies (red, tail end) and iii) convergence to π_* (blue). This example already shows

[1] Linear Q-learning (resp. linear SARSA) is Q-learning (resp. SARSA) with linear function approximation.

[2] This is ensured by setting one column of the state-action feature matrix to the optimal value function.



(a) (b)

Figure 2: Trajectories of three runs of DQN on a 2-state 2-action MDP with a linear 2-dimensional Q-value approximation which *perfectly represents* Q^* (see Section B for implementation details). Figure 2a shows these trajectories in the parameter space (the faded part is the initial behavior). The black star at $(1, 0)$ is Q^*'s parameters. All trajectories start at the same place (the black dot), chosen so that the initial behavior is the ϵ-greedy version of π_*. Figure 2b shows the greedy policies associated with the different trajectories.

how *unreliable* DQN is. It also uncovers serious issues beyond instability (divergence to infinity), which a practitioner cannot avoid by just throwing in more data and computation time.

Existing attempts to study Q-learning or SARSA with function approximation are mainly based on the Ordinary Differential Equation (ODE) method. However, these are of limited utility for explaining the above pheneomena. To see why, note that the ODE method applies to algorithms of the form

$$\theta_{n+1} = \theta_n + \alpha_n[f(\theta_n) + \rho_n + M_{n+1}], \qquad n \geq 0, \tag{1}$$

where $f : \mathbb{R}^d \mapsto \mathbb{R}^d$ is some driving function, α_n is a decaying stepsize, ρ_n is some bias term, and M_{n+1} is the noise. When f is 'nice' overall, e.g., globally Lipschitz continuous, the ODE method can be used to show that the limiting dynamics of (1) is governed by the ODE $\dot{\theta}(t) = f(\theta(t))$ [Benaïm, 1999, Borkar, 2022]. This niceness holds in *policy evaluation*. For Q-learning or SARSA, though, f is quite complex: even with linear function approximation, the update rule is nonlinear and involves sampling from distributions that change with θ_n. So far, the ODE method has only been made to work for such methods by viewing them as general nonlinear schemes and using restrictive assumptions on the sampling distribution: fixed behavior policy [Carvalho et al., 2020], near-optimal behavior policy [Melo et al., 2008, Chen et al., 2022], smooth soft-max behavior policy [Zou et al., 2019], etc. With *ϵ-greedy exploration*, the situation is worse since f then is *discontinuous* and *no analysis exists for it*. As we discuss in Section 4, this discontinuity can, in fact, introduce new limiting behaviors, e.g., sliding mode, which cannot be explained by continuous ODEs.

Key contributions: The main highlights of our work can be summarized as follows.

1. *Novel analysis framework*: We introduce a new framework (see Section 3) utilizing Differential Inclusion (DI) theory [Aubin and Cellina, 2012] to analyze Q-learning and SARSA. Its key steps are i) breaking down the parameter space into regions where the algorithm's dynamics are *simple*, ii) identifying a DI that *stitches* the local dynamics together, and iii) using this DI to explain the algorithm's overall (possibly complex) behavior. Note that a DI is an extension of an ODE that enables the above stitching by allowing for *multiple* update directions at every point.
2. *Explanation of linear Q-learning and SARSA(0) behaviors*: Our main result (Theorem 6) states that the DIs uncovered by our framework govern *all* asymptotic behaviors of linear Q-learning and SARSA(0) employing ϵ-greedy exploration, (idealized) experience replay, and a target network. We thereby answer the question posed by Sutton [1999] and also show our framework's prowess in explaining the behaviors of linear-DQN-type methods, such as those in Figure 2.
3. *Discovery of traps that impede learning*: Our work shows that the limiting DI in general could have several kinds of attractors, and some of these could correspond to sub-optimal policies (see Section 4). In the latter case, these attractors act as traps that prevent the algorithm from learning a better policy. Surprisingly, we note that these attractors do *not* often align with locally optimal

policies. We also show that the policy-oscillation phenomenon is due to a new '*sliding-mode*' attractor. We remark that our DI analysis also applies to the tabular setting, but here there are no local traps because of the guaranteed existence of a global Lyapunov function.

Related work: Several works report various pathological behaviors for approximate value-function-based methods. In planning, Bertsekas and Tsitsiklis [1996] argues that approximate policy iteration may generally be prone to policy oscillations, chattering, and convergence to poor solutions. Similarly, De Farias and Van Roy [2000] shows how approximate value iteration may oscillate forever and not possess any fixed points. Within RL, Gordon [1996, 2000] and Zhang et al. [2023] discuss the chattering phenomenon in linear SARSA(0), but they formally establish only convergence to a bounded region. More recently, Young and Sutton [2020], Schaul et al. [2022], and Patterson et al. [2023] empirically discuss the above pathological behaviors in approximate value-function-based RL methods with greedification. Our work is the first to rigorously explain all these phenomena in RL.

Within the Q-learning literature, a prominent stream uses the ODE method to analyze the linear [Melo et al., 2008, Carvalho et al., 2020, Chen et al., 2022] and nonlinear (neural) function approximation [Fan et al., 2020, Xu and Gu, 2020] variants. However, these works *hold the behavior policy fixed* and impose other conditions such as this policy being close to the optimal policy. These assumptions ensure that the resulting nonlinear ODE has a Lyapunov function and thus convergence guarantees. Another such notable work is [Lee and He, 2020], which uses the switched system theory for analysis. None of these analyses carry over to the ϵ-greedy exploration case because the behavior policy and the resultant dynamics discontinuously change.

There are also analyses that apply ODE methods to study SARSA(0) with changing policies [Melo et al., 2008, Zou et al., 2019]. However, these apply only when the policy improvement operator is Lipschitz continuous with a sufficiently small Lipschitz constant, which ensures the limiting ODE is 'very smooth.' This restrictive condition holds, e.g., for softmax-type policies with a sufficiently small inverse-temperature parameter. Hence, these analyses reveal very little about the behavior under *discontinuous* ϵ-greedy exploration (the case when the inverse temperature parameter is ∞).

A few variants of Q-learning have already been analyzed using DI-based approaches [Maei et al., 2010, Bhatnagar and Lakshmanan, 2016, Avrachenkov et al., 2021]. However, they use DIs for other reasons: the use of sub-gradients, or an intrinsic problem having multiple solutions. This is fundamentally different from our need, which stems from the discontinuity of ϵ-greedy exploration. Finally, [Wunder et al., 2010] and [Banchio and Mantegazza, 2022] use DIs to shed light on the dynamics of (tabular) Q-learning in stateless, multi-agent repeated games.

2 Preliminaries

This section has two distinct parts: this first gives a brief background on Q-learning and SARSA with linear function approximation and ϵ-greedy exploration; the second, a concise introduction to DIs.

2.1 Linear Q-learning and SARSA with ϵ-greedy policy: setup and update rules

For a set U, let $\Delta(U)$ denote the set of probability measures on it. Our setup is that of an MDP $(\mathcal{S}, \mathcal{A}, \gamma, \mathbb{P}, r)$, where $\mathcal{S}$ is a finite state space, $\mathcal{A}$ is a finite action space equipped with a total order, $\gamma \in [0, 1)$ is the discount factor, and $\mathbb{P} : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ and $r : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to \mathbb{R}$ are functions such that $\mathbb{P}(s, a)(s') \equiv \mathbb{P}(s'|s, a)$ specifies the probability of moving from a state s to s' under some action a, while $r(s, a, s')$ is the one-step reward obtained in this transition. Let $Q^* \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$ be the optimal Q-value function associated with this MDP, and $\Phi \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}| \times d}$ the given feature matrix. The standard goal in RL then is to find a $\theta_* \in \mathbb{R}^d$ such that $Q^* \approx \Phi\theta_*$.

Two algorithms to find such a θ_* are linear Q-learning and linear SARSA(0) with ϵ-greedy exploration. Various forms of these algorithms have been discussed in the literature, ranging from the plain vanilla type to more sophisticated ones with a replay buffer and a target network. Up to some idealization, all these variants can be expressed via a single template update rule, which we now describe.

Let $\epsilon \in [0, 1)$ be the greedy-exploration parameter and $\epsilon' \in [0, 1]$ the action-sampling parameter at the succeeding state. Further, let $\ell \geq 0$ and $\mu \equiv (\mu_0, \ldots, \mu_\ell)$ be the replay-buffer length and an associated buffer-sampling distribution. Also, let $\Delta \in (0, 1]$ be the rate at which the target-network estimate is updated. Finally, let $\theta_0^-, \theta_0, \ldots, \theta_{-\ell} \in \mathbb{R}^d$ be some initial estimates of θ_*. Then, for

$n \geq 0$, an unified update rule for linear Q-learning and SARSA(0) is

$$\theta_{n+1} = \theta_n + \alpha_n \delta_n \phi(s_n, a_n), \tag{2}$$

where

$$\delta_n = r(s_n, a_n, s'_n) + \gamma \phi^T(s'_n, a'_n)\theta_n^- - \phi^T(s_n, a_n)\theta_n. \tag{3}$$

In this update rule, $\alpha_n \in \mathbb{R}_{\geq 0}$ is the stepsize, $\theta_n \in \mathbb{R}^d$ is the current estimate of θ_*, while $\theta_n^- \in \mathbb{R}^d$, $n \geq 1$, is the output of the target network. Further, $(\theta_n^-)_{n\geq 0}$ is updated[3] using

$$\theta_{n+1}^- = \theta_n^- + \tau_n(\theta_n - \theta_n^-)\zeta_{n+1}, \tag{4}$$

where (ζ_n) is a sequence of IID Bernoulli random variables with mean Δ, and (τ_n) is another stepsize sequence. Next, $\phi^T(s, a)$, with ^{T} being transpose, denotes the (s, a)-th row of Φ, while δ_n is the one-step Temporal-Difference (TD) error. The next paragraph explicitly describes how the state-action pairs (s_n, a_n) and (s'_n, a'_n) are sampled.

Let $\pi_n^\epsilon : \mathcal{S} \to \Delta(\mathcal{A})$ be the ϵ-greedy policy at time $n \geq -\ell$, i.e., the policy that samples the greedy action w.r.t. $\Phi\theta_n$ (the current Q^* estimate) with probability $1-\epsilon$ and a random action with probability ϵ. In mathematical notations,

$$\pi_n^\epsilon(a|s) = \begin{cases} 1 - \epsilon + \dfrac{\epsilon}{|\mathcal{A}|}, & a = \arg\max_{a'} \phi^T(s, a')\theta_n, \\ \dfrac{\epsilon}{|\mathcal{A}|}, & \text{otherwise.} \end{cases} \tag{5}$$

In the above definition, we presume that $\arg\max$ breaks ties using the total order on $\mathcal{A}$. Similarly, define $\pi_n^{\epsilon'}$ with respect to θ_n^-. Since the state and action spaces are finite, the number of ϵ-greedy policies is finite. We suppose throughout that these policies satisfy the following condition.

$\mathcal{B}_1$. The Markov chain induced by each ϵ-greedy policy is ergodic or, equivalently, aperiodic and irreducible (and hence has a unique stationary distribution).

For $n \geq -\ell$, let d_n^ϵ be the stationary distribution associated with the Markov chain induced by π_n^ϵ. Then, for each $n \geq 0$, (s_n, a_n) and (s'_n, a'_n) are sampled[4] as follows. First, an index $k \in \{0, \ldots, \ell\}$ is sampled from μ; then, s_n is sampled from d_{n-k}^ϵ and a_n from $\pi_{n-k}^\epsilon(\cdot|s_n)$; finally, s'_n is sampled from $\mathbb{P}(\cdot|s_n, a_n)$ and a'_n from $\pi_n^{\epsilon'}(\cdot|s'_n)$. These five samples are drawn with independent randomness.

Remark 1. *Note that* (2) *with $\epsilon' = 0$ (resp. $\epsilon' = \epsilon$) is linear Q-learning (resp. SARSA(0)) with ϵ-greedy exploration. Specifically, the* $\max$ *operator with which Q-learning is usually written is implicitly specified via the manner in which action a'_n is sampled (from the ϵ'-greedy policy).*

Remark 2. *The sampling choice for s_n leads to different variants of Q-learning and SARSA. In standard DQN, s_n is randomly sampled from a replay buffer that holds a sufficiently long but finite record of all the state-action pairs observed recently. Our way of sampling s_n from the stationary distributions associated with $\theta_n, \ldots, \theta_{n-\ell}$ serves as an idealized version of this strategy.*

2.2 Primer on Differential Inclusions

A DI is a relation of the form $\dot{\theta}(t) \in h(\theta(t))$ where $h(\theta)$ is a non-empty *subset* of $\mathbb{R}^d$ for each $\theta \in \mathbb{R}^d$. It reduces to an ODE if $h(\theta)$ is a singleton for all θ. Its solution is any (absolutely continuous) function $t \mapsto \theta(t)$ that satisfies the given DI relation and an initial condition like $\theta(0) = \theta_0$. Unlike ODEs though, the solutions of a DI for an initial condition need not be unique.

The need for DIs can be seen from the following example (cf. [Cortes, 2008, (11)]). Consider (1) with $d = 1$, $f(\theta) = -1$ (resp. +1) for $\theta > 0$ (resp. $\theta \leq 0$), and no bias or noise (i.e., $\rho_n, M_{n+1} \equiv 0$). Due to decaying stepsizes, the iterates should converge to 0. However, this behavior cannot be studied via the ODE $\dot{\theta}(t) = f(\theta(t))$ for which the origin is not even an equilibrium point. In fact, this ODE has no solution at 0: there exists no $t \mapsto \theta(t)$ map with $\theta(0) = 0$ and $\dot{\theta}(t) = f(\theta(t))$; the natural choices: $\theta(t) = -t$, $\theta(t) = +t$, or $\theta(t) \equiv 0$ do not work.

[3] In practice, the target network is updated after every $1/\Delta$-many steps for some $\Delta \in (0, 1)$. We idealize this by presuming that the target-network estimate is updated with probability Δ in every step.

[4] In Section 4, we show that our analysis of this idealized algorithm explains all the behaviors seen in Fig. 2.

The dynamics of the above algorithm, though, can be studied using the DI $\dot{\theta}(t) \in h(\theta(t))$, where $h(\theta) = \{+1\}$ (resp. $\{-1\}$) when $\theta < 0$ (resp. $\theta > 0$), and the interval $[-1, +1]$ for $\theta = 0$. Since $h(0)$ contains the origin, the latter is indeed an unique attractor for this DI. In particular, since $h(0)$ is the convex closure of the set {-1, + 1}, it can be shown that h is *Marchaud*, i.e., *Lipschitz continuous in a set-valued sense* (see $\mathcal{C}_1$ in Theorem 11 for details). Like Lipschitz continuity guarantees the existence of solutions for an ODE (for any initial point), the Marchaud property does so for a DI.

In Section 4, we show that the above picture is natural even in Q-learning with ϵ-greedy policy.

3 Key contributions: our analysis framework & its application to linear Q-learning/SARSA with ϵ-greedy policy

This section has two subsections. In the first, we provide a detailed description of our proposed analysis framework. In the second, we use this framework to give the first pathway to systematically explain all asymptotic behaviors of Q-learning and SARSA(0) with linear function approximation and ϵ-greedy exploration. This approach is summarized in our main result (Theorem 6) below. Proofs of all the results stated here are given in Section A in the appendix.

3.1 Our Analysis Framework

We propose the following approach to analyze an update rule like (1) when f is not continuous.

1. *Partition the parameter space $\mathbb{R}^d$ into regions over which f is 'simple'*: The word simple is subjective and will depend on the algorithm. For linear Q-learning and SARSA with ϵ-greedy exploration, our partition is made up of the θ's where the ϵ-greedy policy is constant. Under linear function approximation, f restricted to these regions turns out to be *linear* and continuous, but it changes *discontinuously* from one region to the other.
2. *Use 'Filippov convexification' to stitch the different f-pieces and make a DI*: Formally, the f-pieces are to be combined via the set-valued map $h : \mathbb{R}^d \to 2^{\mathbb{R}^d}$ (power set of $\mathbb{R}^d$) given by
$$h(\theta) = \bigcap_{\delta > 0} \overline{\text{co}}(f(B(\theta, \delta))). \tag{6}$$
Here, $\overline{\text{co}}$ is the convex closure. Further, $B(\theta, \delta)$ and $f(B(\theta, \delta))$ mean the open ball of radius δ at θ, and its image under f, respectively. The set $h(\theta)$ is the singleton set $\{f(\theta)\}$ if f is continuous at θ, and all convex combinations of neighboring update directions otherwise. The Filippov construction is commonly employed in control theory to deal with discontinuous dynamics [Filippov, 2013]. The DI to study the overall behavior of (1) is
$$\dot{\theta}(t) \in h(\theta(t)). \tag{7}$$
Clearly, the DI for Section 2.2's example matches the one obtained via the above construction.
3. *Establish a formal link between the DI and the algorithm's dynamics*: The idea is to show that the discrete-time iterates (θ_n) of (1) eventually track a solution of the (non-stochastic) DI in (7). To prove this claim, one typically has to show that h is Marchaud, i.e., continuous in a set-valued sense, and that the stepsizes decay sufficiently fast so that the cumulative noise and bias effect is negligible. In our work, we build upon [Borkar, 2022] to rigorously establish these claims. The asymptotics of the DI solutions can then be used to explain all limiting behaviors of the algorithm.

3.2 Application of our framework to linear Q-learning/SARSA with ϵ-greedy policy

We now use our framework to analyze the limiting behaviors of Q-learning and SARSA(0) with linear function approximation, ϵ-greedy exploration, (idealized) experience replay, and a target network.

3.2.1 Analysis Step 1 (partitioning $\mathbb{R}^d$)

We first show how (2) can be rewritten as (1). For any $\theta^-, \theta^{(0)}, \ldots, \theta^{(-\ell)} \in \mathbb{R}^d$, let

$$v(\theta^-, \theta^{(0)}, \ldots, \theta^{(-\ell)}) \;:=\; \mathbb{E}\Big[\delta_0 \phi(s_0, a_0) \Big| \theta_0^- \;=\; \theta^-, \theta_k \;=\; \theta^{(k)}, k \;=\; -\ell, \ldots, 0\Big]. \tag{8}$$

Further, for any $\theta \in \mathbb{R}^d$, let

$$f(\theta) := v(\theta, \theta, \ldots, \theta). \tag{9}$$

Finally, for $n \geq 0$, let

$$\rho_n := v(\theta_n^-, \theta_n, \ldots, \theta_{n-\ell}) - f(\theta_n) \tag{10}$$

$$M_{n+1} := \delta_n \phi(s_n, a_n) - v(\theta_n^-, \theta_n, \ldots, \theta_{n-\ell}). \tag{11}$$

Using the above definitions, it is easy to see that (2) can be expressed in the form given in (1).

Next, we describe the way we partition $\mathbb{R}^d$. For $\mathbf{a} \equiv (\mathbf{a}(s))_s \in \mathcal{A}^{\mathcal{S}}$, let $\mathcal{R}_{\mathbf{a}} := \{\theta \in \mathbb{R}^d : \forall s \in \mathcal{S}, \mathbf{a}(s) = \arg\max_a \phi^T(s,a)\theta\}$, where we break ties in $\arg\max$ using the total order. Clearly, for any $\theta \in \mathbb{R}^d$, there is a unique $\mathbf{a}$ such that $\theta \in \mathcal{R}_{\mathbf{a}}$. Thus, $\{\mathcal{R}_{\mathbf{a}} : \mathbf{a} \in \mathcal{A}^{\mathcal{S}}\}$ partitions $\mathbb{R}^d$, and this is the one we work with. For $\mathbf{a}$ where $\mathcal{R}_{\mathbf{a}} \neq \emptyset$, the greedy (hence, ϵ-greedy) policy corresponding to $\Phi\theta$ is the same for every $\theta \in \mathcal{R}_{\mathbf{a}}$, and it is $\mathbf{a}$. Hence, we refer to $\mathcal{R}_{\mathbf{a}}$ as the *greedy region* associated to $\mathbf{a}$. Finally, note that each $\mathcal{R}_{\mathbf{a}}$ is a cone, i.e., $\theta \in \mathcal{R}_{\mathbf{a}} \implies c\theta \in \mathcal{R}_{\mathbf{a}}$ for any scalar $c > 0$.

The advantage of the above partition is that f has a simple linear form in each region, which we describe in Lemma 3 below. We need a few notations for stating this result. Let $\pi_{\mathbf{a}}^{\epsilon}$ (resp. $\pi_{\mathbf{a}}^{\epsilon'}$) be the ϵ-randomization (resp. ϵ'-randomization) of the policy $\mathbf{a}$. That is, at any state s, $\pi_{\mathbf{a}}^{\epsilon}$ picks a random action with probability ϵ and $\mathbf{a}(s)$, the action prescribed by $\mathbf{a}$, with probability $1-\epsilon$. Clearly, $\pi_n^{\epsilon} = \pi_{\mathbf{a}}^{\epsilon}$ (resp. $\pi_n^{\epsilon'} = \pi_{\mathbf{a}}^{\epsilon'}$) whenever $\theta_n \in \mathcal{R}_{\mathbf{a}}$ (resp. $\theta_n^- \in \mathcal{R}_{\mathbf{a}}$). Next, let $d_{\mathbf{a}}^{\epsilon}$ denote the stationary distribution associated with the Markov chain induced by $\pi_{\mathbf{a}}^{\epsilon}$, and let

$$b_{\mathbf{a}} := \mathbb{E}[\phi(s,a) r(s,a,s')] = \Phi^T D_{\mathbf{a}}^{\epsilon} \mathbf{r} \tag{12}$$

and

$$A_{\mathbf{a}} := \mathbb{E}[\phi(s,a)\phi^T(s,a) - \gamma\phi(s,a)\phi^T(s',a')] = \Phi^T D_{\mathbf{a}}^{\epsilon}(\mathbb{I} - \gamma P_{\mathbf{a}}^{\epsilon'})\Phi. \tag{13}$$

In the above definitions, the expectation is with respect to $s \sim d_{\mathbf{a}}^{\epsilon}$, $a \sim \pi_{\mathbf{a}}^{\epsilon}(\cdot|s)$, $s' \sim \mathbb{P}(\cdot|s,a)$, and $a' \sim \pi_{\mathbf{a}}^{\epsilon'}(\cdot|s')$. Further, $D_{\mathbf{a}}^{\epsilon}$ is the diagonal matrix of size $|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}||\mathcal{A}|$ whose (s,a)-th diagonal entry is $d_{\mathbf{a}}^{\epsilon}(s)\pi_{\mathbf{a}}^{\epsilon}(a|s)$, $\mathbf{r}$ is the $|\mathcal{S}||\mathcal{A}|$-dimensional vector whose (s,a)-th coordinate is $\mathbf{r}(s,a) = \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s,a) r(s,a,s')$, while $P_{\mathbf{a}}^{\epsilon'}$ is the matrix of size $|\mathcal{S}||\mathcal{A}| \times |\mathcal{S}||\mathcal{A}|$ such that $P_{\mathbf{a}}^{\epsilon'}((s,a),(s',a')) = \mathbb{P}(s'|s,a)\pi_{\mathbf{a}}^{\epsilon'}(a'|s')$.

Lemma 3. *For $\theta \in \mathbb{R}^d$, the function f given in* (9) *satisfies $f(\theta) = \sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} (b_{\mathbf{a}} - A_{\mathbf{a}}\theta)\, \mathbb{1}[\theta \in \mathcal{R}_{\mathbf{a}}]$.*

Remark 4. *While f is nonlinear overall, Lemma 3 shows that it is piece-wise linear. That is, $f(\theta) = b_{\mathbf{a}} - A_{\mathbf{a}}\theta$ for $\theta \in \mathcal{R}_{\mathbf{a}}$, and this definition changes discontinuously from one greedy region to the other. For $\epsilon = \epsilon'$, $f|_{\mathcal{R}_{\mathbf{a}}}$ is the driving function that governs the behavior of TD(0) with linear function approximation for evaluating the policy $\pi_{\mathbf{a}}^{\epsilon}$ [Sutton and Barto, 2018, (9.11)]. Figure 3 shows the partitions and the nature of f over each sub-region for two different MDP settings.*

3.2.2 Analysis Step 2 (DI identification)

The DI to study the limiting dynamics of (2) is the one given in (7), where the set-valued map $h : \mathbb{R}^d \to 2^{\mathbb{R}^d}$ from (6) is defined using the function f from (9) (or, equivalently, the one in Lemma 3). Henceforth, we refer to this DI as the limiting DI of (2).

In Lemma 5 below, we give an equivalent but simpler description of this specific function h. For $\theta \in \mathbb{R}^d$, let $\text{supp}(\theta) = \left\{\mathbf{a} \in \mathcal{A}^{\mathcal{S}} : \phi^T(s, \mathbf{a}(s))\theta = \max_{a \in \mathcal{A}} \phi^T(s,a)\theta \; \forall s \in \mathcal{S}\right\}$. Clearly, $1 \leq |\text{supp}(\theta)| \leq |\mathcal{A}|^{|\mathcal{S}|}$ since $\{\mathcal{R}_{\mathbf{a}}\}$ partitions $\mathbb{R}^d$. In particular, if θ is in the interior of $\mathcal{R}_{\mathbf{a}}$ for some $\mathbf{a}$, then $\text{supp}(\theta) = \{\mathbf{a}\}$; for the one on the boundary, $|\text{supp}(\theta)| \geq 2$.

Lemma 5. *For $\theta \in \mathbb{R}^d$, we have $h(\theta) = \text{co}\left\{b_{\mathbf{a}} - A_{\mathbf{a}}\theta : \mathbf{a} \in \text{supp}(\theta)\right\}$, where* co *is the convex hull. Specifically, $h(\theta) = \{b_{\mathbf{a}} - A_{\mathbf{a}}\theta\}$ for any θ in the interior of $\mathcal{R}_{\mathbf{a}}$. Further, $f(\theta) \in h(\theta)$.*

3.2.3 Analysis Step 3 (algorithm-DI connection)

Our main result (Theorem 6) is that (2)'s limiting DI completely governs its limiting dynamics. To state this result, we need two additional assumptions. Let $\|\cdot\|$ denote the Euclidean norm.

$\mathcal{B}_2$. (α_n) satisfies $\sup_{n \geq 0} \alpha_n \leq 1$, $\sum_{n \geq 0} \alpha_n = \infty$ and $\sum_{n \geq 0} \alpha_n^2 < \infty$. Further, (τ_n) satisfies $\sup_{n \geq 0} \tau_n \leq 1$, $\sum_{n \geq 0} \tau_n = \infty$, $\sum_{n \geq 0} \tau_n^2 < \infty$, and $\lim_{n \to \infty} \frac{\alpha_n}{\tau_n} = 0$.

$\mathcal{B}_3$. Φ has full column rank.

Our result also needs a few definitions. In relation to (7), we will say a set $\Gamma \subseteq \mathbb{R}^d$ is *invariant* if, for every $\theta_0 \in \Gamma$, there is *some* solution trajectory $(\theta(t))_{t\in(-\infty,\infty)}$ of (7) with $\theta(0) = \theta_0$ that lies entirely in Γ. An invariant set Γ is additionally *internally chain transitive* if it is compact and connected in a certain way: for $x, y \in \Gamma$, $\nu > 0$, and $T > 0$, there exist $m \geq 1$ and points $x_0 = x, x_1, \ldots, x_{m-1}, x_m = y$ in Γ such that a solution trajectory of (7) initiated at x_i meets the ν-neighborhood of x_{i+1} for $0 \leq i < m$ after a time that is equal or larger than T. Such characterizations are useful to restrict the possible sets to which (2) could converge to. For example, for the DI in Section 2.2, while $\mathbb{R}$, $[0, \infty)$, $(-\infty, 0]$, and $\{0\}$ are all invariant, only $\{0\}$ is internally chain transitive.

Theorem 6 (**Main Result**). *Suppose $\mathcal{B}_1$, $\mathcal{B}_2$, and $\mathcal{B}_3$ hold. Then, (θ_n) obtained by* (2) *converges to a closed, connected, internally chain transitive set of its limiting DI a.s. on the event $\{\sup_n \|\theta_n\| < \infty\}$.*

Remark 7. *Our result states that (θ_n) either diverges to ∞ or converges to a suitable (sample-point dependent) invariant set of its limiting DI. In this way, our result captures all possible limiting behaviors of* (2) *and resolves the open question in [Sutton, 1999, Problem 1]. Notably, our result is the first to characterize the asymptotic behaviors of any value-function-based algorithm with function approximation and ϵ-greedy exploration. In Section 4, we show that the convergence to an invariant set does not guarantee the superiority of a resulting limiting greedy policy over intermediate policies.*

Remark 8. *The limiting DI for* (2) *does not depend on the hyperparameters such as experience replay length ℓ, target-network refresh rate Δ, and the stepsizes (α_n) and (τ_n). This means that adjusting these hyperparameters does not change the possible limiting sets for the sequence (θ_n).*

Remark 9. *There are two important cases of value-function-based algorithms where the iterates are already known to be almost surely stable, i.e., $\mathbb{P}\{\sup_{n\geq 0} \|\theta_n\| < \infty\} = 1$. In these cases, our claim holds on almost every sample point. The first case is that of linear SARSA(0) with ϵ-greedy exploration ($\epsilon' = \epsilon$), but without experience replay ($\ell = 0$) or a target network ($\theta_n^- = \theta_n$). Its stability has been established in [Gordon, 2000]. The second case is that of tabular Q-learning ($\Phi = \mathbb{I}$), whose stability follows using a simple inductive argument, e.g., [Gosavi, 2006].*

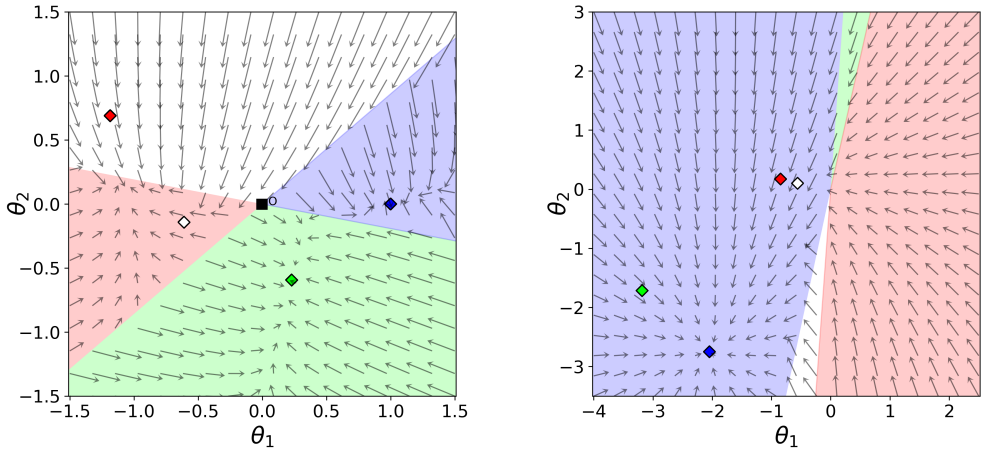
Remark 10. *Our DI-based approach can also recover the well-known result that tabular Q-learning (i.e.,* (2) *with $\epsilon' = 0$ and $\Phi = \mathbb{I}$) converges to Q^* a.s. First, Theorem 6 and Remark 9 together show that this algorithm's iterates must a.s. converge to some invariant set of its limiting DI. Separately, it can be shown that $\|\theta - Q^*\|$ serves as a global Lyapunov function and, hence, $\{Q^*\}$ is the unique globally asymptotically stable invariant set of this DI. The desired claim now follows. We note that, even here, we have greedy regions and dynamics change discontinuously from one region to the other.*

4 Numerical illustrations and discussion

We now use Theorem 6 to explain the 'problematic' behaviors of linear DQN in Figure 2. Additionally, we give an MDP example where linear DQN will always converge to the worst policy.

Explanation of Figure 2. For the linear DQN example in Figure 2, we use Step 1 (Lemma 3) of our framework to get the associated vector-field $f : \mathbb{R}^2 \to \mathbb{R}^2$ of its limiting DI. This is given in Figure 3a, along with the underlying partition. This MDP has four distinct policies: the colored cones are the corresponding greedy regions. The local dynamics is governed by the associated $b_{\mathbf{a}}$ and $A_{\mathbf{a}}$ values. Each diamond is the point $A_{\mathbf{a}}^{-1} b_{\mathbf{a}}$, the equilibrium for $\dot{\theta}(t) = b_{\mathbf{a}} - A_{\mathbf{a}}\theta(t)$, which we dub as the *'landmark'* for the dynamics in $\mathcal{R}_{\mathbf{a}}$. Note how the vector field is *discontinuous* at the boundaries.

Step 2 convexifies the vector field on the boundaries between regions. In effect, it permits solutions (of the DI) in which the velocity at a boundary point can be *any convex combination* of the two (different) velocities associated with the regions comprising the boundary. Applying this logic to Figure 3a, we get the following patterns: i.) trajectories starting from the blue region either remain there and converge to its (blue) landmark at $[1, 0]^T$ (representing Q^*) or cross over to the green region, ii.) trajectories that start within the green region converge either to its (green) landmark or cross over to the blue region, iii.) trajectories that start from the red region either cross over to the green region, or hit the red-white boundary in finite time. In the latter case, since the red and white vector fields near the boundary are always oriented towards it, the resultant solutions are forced to 'slide' along the boundary towards a point where the red and white regions' velocities oppose each other (a *sliding mode attractor*), iv.) trajectories starting from the white region either cross over to the blue region, after which i.) applies, or hit the red-white boundary and slide as before.



(a) (b)

Figure 3: The vector field $f(\theta)$ for 2 different MDP settings; the colored regions represent greedy partitions and the diamond markers their respective landmarks. (Left) The MDP setting of Figure 2. (Right) An MDP setting where Q-learning will always find the worst policy.

Step 3 or Theorem 6 guarantees that the (orginal) iterates of Fig. 2's linear DQN converge to a closed, connected, invariant and internally chain transitive set of the above DI. It can rigorously be established that the only such sets of this DI are 4 singletons: i.) the green and blue landmark points corresponding to a suboptimal and the optimal policy, respectively, ii.) a point θ_{sliding} on the red-white boundary which is not a proper landmark but satisfies $0 \in h(\theta_{\text{sliding}})$, and iii.) a similar point θ_{unstable} on the red-green boundary. This final point, though, is an unstable equilibrium point, because any neighborhood around it contains points from where the DI's solutions will escape away from it.

Figure 2a can now be explained as follows: the blue (resp. green) trajectory in Fig. 2a converges to the blue (resp. green) diamond in Fig. 3a; recall that the blue diamond is Q^*'s parameter. In contrast, the red trajectory goes to the sliding-mode attractor on the boundary between the red and white regions. In the last case, the iterates continuously 'chatter' or bounce between the red and white regions, which explains the policy oscillation we see in Fig. 2b. Suppose we define the neighborhood of a stable equilibrium as the set of cones which i.) contain the equilibrium, or ii) shares a boundary with the cone(s) containing the equilibrium. That is, the set of places that linear DQN can potentially explore before reaching this equilibrium, e.g., green landmark's neighborhood consists of the green, blue, and red cones. Then, it follows that neither the green nor the sliding-mode attractor is locally optimal! The positive probability of convergence to any attractor from any starting point now explains why linear DQN may end up at a locally sub-optimal policy or even one that is worse than the initial.

Reliable convergence but to the worst policy. Fig 3b provides the vector field for linear DQN's limiting DI in the context of another 2-state, 2-action MDP (see Section B for details). Interpreting this vector field as above, we get that linear DQN's iterates will converge a.s. to the blue diamond. The striking fact is that the greedy policy associated to this landmark is the *worst* of all the 4 deterministic policies, demonstrating a hopeless 'no-improvement' scenario. A similar observation has been made by Young and Sutton [2020] for the episodic (finite-horizon, undiscounted) MDP setting.

Discussion: On a somber note, our insights about Q-learning and SARSA under arguably the simplest possible (linear) function approximation with ϵ-greedy exploration cast doubt on their utility in more complicated, nonlinear approximation architectures. Unless the specific setting where the algorithms are applied has favorable structural properties (in terms of its limiting DI), the practitioner must anticipate unreliable behaviors. Our work also reinforces the fact that merely ensuring stability of an incremental RL algorithm's iterates is by no means sufficient to guarantee good performance—the discontinuous policy update and the sampling distribution can still induce complex behaviors.

On the positive side, our approach provides a systematic design pathway for reliable RL algorithms whose associated DIs are sound, e.g., those whose attractors lie in regions associated with high-value policies, potentially via Lyapunov techniques.

Acknowledgments and Disclosure of Funding

We express our gratitude to the anonymous reviewers for taking out their time and providing us with valuable comments. This feedback has helped in improving the quality of the paper. Gugan Thoppe's research is supported in part by DST-SERB's Core Research Grant CRG/2021/008330, the Indo-French Centre for the Promotion of Advanced Research—CEFIPRA (7102-1), the Walmart Center for Tech Excellence, the Kotak-IISc AI/ML Centre, and by the Pratiksha Trust Young Investigator Award. He would also declare his Associate Researcher position at the Robert Bosch Centre for Data Science and Artificial Intelligence (RBCDSAI) at IIT Madras, Chennai 600 036, India.

References

J-P Aubin and Arrigo Cellina. *Differential inclusions: set-valued maps and viability theory*, volume 264. Springer Science & Business Media, 2012.

Konstantin E Avrachenkov, Vivek S Borkar, Hars P Dolhare, and Kishor Patil. Full gradient DQN reinforcement learning: A provably convergent scheme. In *Modern Trends in Controlled Stochastic Processes:*, pages 192–220. Springer, 2021.

Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pages 30–37. Elsevier, 1995.

Martino Banchio and Giacomo Mantegazza. Adaptive algorithms and collusion via coupling. *arXiv preprint arXiv*, 2202, 2022.

Michel Benaïm. Dynamics of stochastic approximation algorithms. In *Seminaire de probabilites XXXIII*, pages 1–68. Springer, 1999.

Michel Benaïm, Josef Hofbauer, and Sylvain Sorin. Stochastic approximations and differential inclusions. *SIAM Journal on Control and Optimization*, 44(1):328–348, 2005.

Dimitri P Bertsekas. Approximate policy iteration: A survey and some new methods. *Journal of Control Theory and Applications*, 9(3):310–335, 2011.

Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-dynamic programming*. Athena Scientific, 1996.

Shalabh Bhatnagar and K Lakshmanan. Multiscale Q-learning with linear function approximation. *Discrete Event Dynamic Systems*, 26(3):477–509, 2016.

Vivek S Borkar. *Stochastic approximation: a dynamical systems viewpoint*, volume 48. Hindustan Book Agency, 2022. Second Edition.

Diogo Carvalho, Francisco S Melo, and Pedro Santos. A new convergent variant of q-learning with linear function approximation. *Advances in Neural Information Processing Systems*, 33: 19412–19421, 2020.

Zaiwei Chen, Sheng Zhang, Thinh T Doan, John-Paul Clarke, and Siva Theja Maguluri. Finite-sample analysis of nonlinear stochastic approximation with applications in reinforcement learning. *Automatica*, 146:110623, 2022.

Jorge Cortes. Discontinuous dynamical systems. *IEEE Control systems magazine*, 28(3):36–73, 2008.

Daniela Pucci De Farias and Benjamin Van Roy. On the existence of fixed points for approximate value iteration and temporal-difference learning. *Journal of Optimization theory and Applications*, 105(3):589–608, 2000.

Jianqing Fan, Zhaoran Wang, Yuchen Xie, and Zhuoran Yang. A theoretical analysis of deep Q-learning. In *Learning for Dynamics and Control*, pages 486–489. PMLR, 2020.

Aleksei Fedorovich Filippov. *Differential equations with discontinuous righthand sides: control systems*, volume 18. Springer Science & Business Media, 2013.

Geoffrey J Gordon. Chattering in SARSA (lambda)-a CMU learning lab internal report. Technical report, Carnegie Mellon University, 1996.

Geoffrey J Gordon. Reinforcement learning with function approximation converges to a region. *Advances in neural information processing systems*, 13, 2000.

Abhijit Gosavi. Boundedness of iterates in q-learning. *Systems & control letters*, 55(4):347–349, 2006.

Donghwan Lee and Niao He. A unified switching system perspective and convergence analysis of Q-learning algorithms. *Advances in Neural Information Processing Systems*, 33, 2020.

Fan Lu, Prashant G Mehta, Sean P Meyn, and Gergely Neu. Convex Q-learning. In *2021 American Control Conference (ACC)*, pages 4749–4756. IEEE, 2021.

Hamid Reza Maei, Csaba Szepesvári, Shalabh Bhatnagar, and Richard S Sutton. Toward off-policy learning control with function approximation. In *ICML*, 2010.

Francisco S Melo, Sean P Meyn, and M Isabel Ribeiro. An analysis of reinforcement learning with function approximation. In *Proceedings of the 25th international conference on Machine learning*, pages 664–671, 2008.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.

Andrew Patterson, Samuel Neumann, Martha White, and Adam White. Empirical design in reinforcement learning. *arXiv preprint arXiv:2304.01315*, 2023.

Herbert Robbins and Sutton Monro. A stochastic approximation method. *The annals of mathematical statistics*, pages 400–407, 1951.

Tom Schaul, André Barreto, John Quan, and Georg Ostrovski. The phenomenon of policy churn. *arXiv preprint arXiv:2206.00730*, 2022.

Richard S Sutton. Open theoretical questions in reinforcement learning. In *European Conference on Computational Learning Theory*, pages 11–17. Springer, 1999.

Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.

Michael Wunder, Michael L Littman, and Monica Babes. Classes of multiagent q-learning dynamics with epsilon-greedy exploration. In *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 1167–1174, 2010.

Pan Xu and Quanquan Gu. A finite-time analysis of Q-learning with neural network function approximation. In *International Conference on Machine Learning*, pages 10555–10565. PMLR, 2020.

Kenny Young and Richard S Sutton. Understanding the pathologies of approximate policy evaluation when combined with greedification in reinforcement learning. *arXiv preprint arXiv:2010.15268*, 2020.

Shangtong Zhang, Remi Tachet Des Combes, and Romain Laroche. On the convergence of sarsa with linear function approximation. In *International Conference on Machine Learning*, pages 41613–41646. PMLR, 2023.

Shaofeng Zou, Tengyu Xu, and Yingbin Liang. Finite-sample analysis for SARSA with linear function approximation. *Advances in Neural Information Processing Systems*, 32, 2019.

A Proofs

The proofs for all Section 3's results are given here. The proof of our main result is in Subsection A.1, and those of Lemmas 3 and 5 are in Subsection A.2.

A.1 Proof of our Main Result (Theorem 6)

Using our Step 1 from Section 3.2, recall that the linear Q-learning/SARSA(0) update from (2) can be rewritten as (1), where f, ρ_n, and M_{n+1} are given by (9) (or, equivalently, the expression in Lemma 3), (10), and (11), respectively. Compared to a standard stochastic approximation [Robbins and Monro, 1951, Borkar, 2022, Benaïm, 1999], the analysis of (2) has two main challenges. First, the resultant driving function f is discontinuous (see Lemma 3), a consequence of the ϵ-greedy exploration. Second, the perturbation term ρ_n need not necessarily decay to 0, especially when (θ_n) continually jumps between two or more greedy regions. Recall that ρ_n arises due to the experience-replay and target-network-based sampling of s_n, a_n, s'_n, and a'_n.

Our approach to overcoming the above two challenges is as follows. We handle the discontinuity by treating (2) as a Stochastic Recursive Inclusion (SRI) [Benaïm et al., 2005, Borkar, 2022] wherein the driving function can also be discontinuous, unlike in a stochastic approximation. This enables us to study the limiting dynamics of (2) using the powerful DI viewpoint instead of the standard differential-equation-based one. Separately, we handle ρ_n by carefully decomposing it into terms that arise only due to experience replay, and those that arise due to the target network. On a sample path where the (θ_n) iterates are stable, we exploit the fact that the target network parameter θ_n^- is updated using a faster timescale to show that $\|\theta_n^- - \theta_n\|$ and, hence, the terms that depend on θ_n^- asymptotically vanish. In contrast, for the terms that arise due to experience replay, we show that a telescopic sum exists that ensures their cumulative effect is asymptotically negligible.

A key result that we build upon to handle the discontinuity of f in (2) is [Borkar, 2022, Corollary 5.1] which concerns the convergence of Stochastic Recursive Inclusions (SRIs). To help the reader, we first describe SRIs and then state the above result. Alongside, we also explain why this result is not sufficient to directly prove Theorem 6. Finally, we provide our own proof.

An SRI is a generic update like

$$\theta_{n+1} = \theta_n + \alpha_n[y_n + M_{n+1}], \qquad n \geq 0, \tag{14}$$

where y_n is some desired update vector satisfying $y_n \in h(\theta_n)$ for some set-valued map h, α_n is some stepsize, and M_{n+1} is noise. A stochastic approximation and more specifically a stochastic gradient descent are special cases of an SRI, where $h(\theta)$ is a singleton for all θ. Note that (2) has the form given in (14), but with an additional perturbation term ρ_n. In particular, in the case of (2), $y_n = f(\theta_n) \in h(\theta_n)$, where f and h are as in (9) and (6), respectively.

We next state [Borkar, 2022, Corollary 5.1], which provides a sufficient set of conditions for the (θ_n) sequence generated by (14) to converge to the invariant sets of the DI $\dot{\theta}(t) \in h(\theta(t))$.

Theorem 11 (Corollary 5.1, Borkar [2022])**.** *Consider a generic SRI like* (14) *and suppose the following conditions hold.*

$\mathcal{C}_1$. ***Driving function**: h is Marchaud or continuous in a set-value sense, i.e.,*

- *(a) $h(\theta)$ is convex and compact for all $\theta \in \mathbb{R}^d$;*
- *(b) $\exists K_h > 0$ such that $\sup_{y \in h(\theta)} \|y\| \leq K_h(1 + \|\theta\|)$ for all $\theta \in \mathbb{R}^d$, and*
- *(c) h is upper semicontinuous or, equivalently, $\{(\theta, y) \in \mathbb{R}^d \times \mathbb{R}^d : y \in h(\theta)\}$ is closed.*

$\mathcal{C}_2$. ***Stepsize schedule**: (α_n) is a non-increasing sequence that satisfies the Robbins-Monro condition, i.e., $\sum_{n=0}^{\infty} \alpha_n = \infty$, but $\sum_{n=0}^{\infty} \alpha_n^2 < \infty$.*

$\mathcal{C}_3$. ***Noise behavior**: (M_n) is a square-integrable martingale-difference sequence adapted to an increasing family of σ-fields $(\mathcal{F}_n)$. Furthermore, there exists a constant $K_m \geq 0$ such that $\mathbb{E}[\|M_{n+1}\|^2 | \mathcal{F}_n] \leq K_m[1 + \|\theta_n\|^2]$ a.s. for all $n \geq 0$.*

Then, almost surely on every sample path where the iterate sequence (θ_n) is stable, i.e., $\sup_{n \geq 0} \|\theta_n\| < \infty$, we have that (θ_n) converges to a (possibly sample-path dependent) closed, connected, internally chain transitive set of the DI $\dot{\theta}(t) \in h(\theta(t))$.

Theorem 11 is not directly applicable to (2) due to the additional perturbation term, ρ_n. Instead, we prove Theorem 6 by building upon Theorem 11's proof from [Borkar, 2022]. Our strategy involves, firstly, verifying that h, (α_n), and (M_n), as defined in the context of (2), meet the criteria stipulated in Theorem 11. Thereafter, we prove that the cumulative impact of the ρ_n's is asymptotically negligible, a key and intricate part of our analysis. Finally, we show that, under the above conditions, the core arguments and thereby the conclusions of Theorem 11 are upheld, which then leads to our result.

Proof of Theorem 6. We first show that the conditions $\mathcal{C}_1$, $\mathcal{C}_2$, and $\mathcal{C}_3$ of Theorem 11 hold for (2).

Consider $\mathcal{C}_1$. Lemma 5 shows that $h(\theta)$ is convex. Since $|\text{supp}(\theta)| \leq |\mathcal{A}^{\mathcal{S}}|$, a finite number, we also have that $h(\theta)$ is closed and bounded (hence, compact). This establishes ($\mathcal{C}_1$.a). Similarly, we have

$$\sup_{y \in h(\theta)} \|y\| \leq \max \left\{ \max_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} \|b_{\mathbf{a}}\|, \max_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} \|A_{\mathbf{a}}\| \right\} (1 + \|\theta\|).$$

Separately, since we have finite state and action spaces, we have that $\exists K_\phi, K_r \geq 0$ such that, for any $s, s' \in \mathcal{S}$ and $a \in \mathcal{A}$,

$$\|\phi(s,a)\| \leq K_\phi \qquad \text{and} \qquad |r(s,a,s')| \leq K_r. \tag{15}$$

From (12), (13), and (15), it then follows that $\|b_{\mathbf{a}}\| \leq K_\phi K_r$ and $\|A_{\mathbf{a}}\| \leq (1 + \gamma\sqrt{|\mathcal{S}|})K_\phi^2$. Hence, ($\mathcal{C}_1$.b) is satisfied for $K_h := K_\phi \max\{K_r, (1 + \gamma\sqrt{|\mathcal{S}|})K_\phi\}$. It remains to establish the upper semicontinuity of h. That is, for any sequences (x_n) and (z_n) such that $x_n \to \theta$, $z_n \to y$, and $z_n \in h(x_n)\, \forall n \geq 0$, we need to show that $y \in h(\theta)$. This is a consequence of the 'Filippov convexification' and, hence, we use the form of h given in (6) for deriving it. Let $\delta > 0$ be arbitrary. Then, $\exists N_\delta \geq 0$ such that $x_n \in B(\theta, \delta)$ for all $n \geq N_\delta$. Further, for each such n, since $B(\theta, \delta)$ is open, there is also small ball around x_n that is contained in $B(\theta, \delta)$ which, in turn, implies

$$z_n \in h(x_n) \subseteq \overline{\text{co}}(f(B(\theta, \delta))). \tag{16}$$

Because the set on the extreme right is closed and y is the limit of (z_n), we have $y \in \overline{\text{co}}(f(B(\theta, \delta)))$. The choice of δ being arbitrary finally shows that $y \in h(\theta)$, as desired.

Condition $\mathcal{C}_2$ holds due to our stepsize assumption in $\mathcal{B}_2$.

Next, consider $\mathcal{C}_3$. Let $\mathcal{F}_n$ be the σ-field generated by $\theta_0^-, \theta_0, \ldots, \theta_{-\ell}, s_0, a_0, s_0', a_0', \ldots, s_{n-1}, a_{n-1}, s_{n-1}', a_{n-1}'$. The fact that (M_n) is a martingale-difference sequence adapted to $(\mathcal{F}_n)$ is a direct consequence of its definition in (11). Further, for any $n \geq 0$, it follows from (2) and (15) that

$$\begin{aligned} \|\delta_n \phi(s_n, a_n)\| &\leq |\delta_n| \, \|\phi(s_n, a_n)\| \\ &\leq K_\phi \left[K_r + \gamma K_\phi \|\theta_n^-\| + K_\phi \|\theta_n\| \right] \end{aligned} \tag{17}$$

and, hence,

$$\|M_{n+1}\| \leq 2K_\phi \left[K_r + \gamma K_\phi \|\theta_n^-\| + K_\phi \|\theta_n\| \right]. \tag{18}$$

Therefore, $\mathbb{E}[\|M_{n+1}\|^2 | \mathcal{F}_n] \leq K_m[1 + \|\theta_n^-\|^2 + \|\theta_n\|^2]$ for $K_m = 12K_\phi^2 \max\{K_r^2, K_\phi^2\}$. Note that there is an additional $\|\theta_n^-\|$ term in this expression. However, it does not pose any additional difficulties and can be handled just like the $\|\theta_n\|$ term. It remains to show that (M_n) is a square integrable sequence, i.e., $\mathbb{E}\|M_n\|^2 < \infty$ for all $n \geq 1$. Since $\|\theta_0\|^2, \|\theta_0^-\|^2 < \infty$, it follows from (2), (4), and using (17) and (18) for $n = 0$, that $\mathbb{E}\|M_1\|^2 < \infty$, $\mathbb{E}\|\theta_1\|^2 < \infty$ and $\mathbb{E}\|\theta_1^-\|^2 < \infty$. The desired result now follows by induction.

We now study the asymptotic behavior of (θ_n), updated using (2) and (4), along the lines given in the proof of [Borkar, 2022, Corollary 5.4]. Fix a sample point where $\sup_{n \geq 0} \|\theta_n\| < \infty$. Then, using a simple induction argument, it follows that $\sup_{n \geq 0} \|\theta_n^-\| < \infty$.

Next, we look at θ_n^-'s asymptotic behavior. Clearly, (2) and (4) can be jointly viewed as a two-timescale algorithm. Specifically, since $\alpha_n/\tau_n \to 0$, it follows that (θ_n) is updated on a slower timescale relative to (θ_n^-); hence, the (θ_n) sequence would appear static from the viewpoint of (4). Now, for the case where $\theta_n \equiv \theta$ for some $\theta \in \mathbb{R}^d$, (4)'s limiting ODE $\dot{x}(t) = \Delta(\theta - x(t))$ has θ as its unique globally asymptotically stable equilibrium. This limit, as a function of θ, is trivially Lipschitz continuous. Hence, from [Borkar, 2022, Lemma 8.1], we have

$$\|\theta_n^- - \theta_n\| \to 0. \tag{19}$$

Next, we discuss the asymptotic behavior of (θ_n) obtained via (2). Let $T > 0$ be an arbitrary horizon. Further, for $n \geq 0$, let m_n be the smallest index k such that $T \leq \sum_{j=n}^{n+k} \alpha_j \leq T + 1$. Then, for any $n \geq 0$ and m such that $0 \leq m \leq m_n$, we have that

$$\theta_{n+m+1} = \theta_n + \sum_{j=n}^{n+m} \alpha_j f(\theta_n) + \sum_{j=n}^{n+m} \alpha_j M_{j+1} + \sum_{j=n}^{n+m} \alpha_j \rho_j. \tag{20}$$

The above expression is of the form given in [Borkar, 2022, (2.1.6)][5], except for the additional sum involving the perturbation terms. Hence, if we can show that $\sup_{0 \leq m \leq m_n} \| \sum_{j=n}^{n+m} \alpha_j \rho_j \| \to 0$ as $n \to \infty$, then Theorem 6 would follow by using similar arguments as in [Borkar, 2022, Corollary 5.4].

We now show the above claim. For this, we decompose $\alpha_n \rho_n$ into terms that arise due to experience replay and those arise due to the target network. From (8), (3), (12), and (13), we have that, for any $\theta^-, \theta^{(0)}, \ldots, \theta^{(-\ell)}$,

$$\begin{aligned}
&v(\theta^-, \theta^{(0)}, \ldots, \theta^{(-\ell)}) \\
&\quad = \mathbb{E}\left[\delta_0 \phi(s_0, a_0) \middle| \theta_0^- = \theta^-, \theta_k = \theta^{(k)}, k = -\ell, \ldots, 0\right] \\
&\quad = \sum_{k=0}^{\ell} \mu_k \Bigg(\sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} \left[(b_{\mathbf{a}} - A_{\mathbf{a}} \theta^{(0)}) \mathbb{1}[\theta^{(-k)} \in \mathcal{R}_{\mathbf{a}}] - \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} P_{\mathbf{a}}^{\epsilon'} \Phi \theta^{(0)} \mathbb{1}[\theta^{(-k)} \in \mathcal{R}_{\mathbf{a}}] \right] \\
&\qquad\qquad + \sum_{\mathbf{a}, \mathbf{a}' \in \mathcal{A}^{\mathcal{S}}} \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} P_{\mathbf{a}'}^{\epsilon'} \Phi \theta^- \mathbb{1}[\theta^{(-k)} \in \mathcal{R}_{\mathbf{a}}, \theta^- \in \mathcal{R}_{\mathbf{a}'}] \Bigg).
\end{aligned}$$

Therefore, using (10), we get that

$$\begin{aligned}
\alpha_n \rho_n &= \alpha_n \left[v(\theta_n^-, \theta_n, \ldots, \theta_{n-\ell}) - f(\theta_n) \right] \\
&= \alpha_n \sum_{k=0}^{\ell} \mu_k \Bigg(\sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} (b_{\mathbf{a}} - A_{\mathbf{a}} \theta_n) \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}] \\
&\qquad + \sum_{\mathbf{a}, \mathbf{a}' \in \mathcal{A}^{\mathcal{S}}} \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} \left[P_{\mathbf{a}'}^{\epsilon'} \Phi \theta_n^- - P_{\mathbf{a}}^{\epsilon'} \Phi \theta_n \right] \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}, \theta_n^- \in \mathcal{R}_{\mathbf{a}'}] - f(\theta_n) \Bigg) \\
&= \rho_n^{(1)} + \rho_n^{(2)} + \rho_n^{(3)} + \rho_n^{(4)},
\end{aligned} \tag{21}$$

where

$$\rho_n^{(1)} := \sum_{k=0}^{\ell} \mu_k \left[\alpha_{n-k} f(\theta_{n-k}) - \alpha_n f(\theta_n) \right]; \tag{22}$$

$$\rho_n^{(2)} := \sum_{k=0}^{\ell} \mu_k (\alpha_n - \alpha_{n-k}) f(\theta_{n-k}); \tag{23}$$

$$\begin{aligned}
\rho_n^{(3)} &:= \alpha_n \sum_{k=0}^{\ell} \mu_k \left[\sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} (b_{\mathbf{a}} - A_{\mathbf{a}} \theta_n) \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}] - f(\theta_{n-k}) \right] \\
&\qquad + \alpha_n \sum_{k=0}^{\ell} \mu_k \sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} P_{\mathbf{a}}^{\epsilon'} \Phi (\theta_{n-k} - \theta_n) \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}]
\end{aligned} \tag{24}$$

$$= \alpha_n \sum_{k=0}^{\ell} \mu_k \sum_{\mathbf{a} \in \mathcal{A}^{\mathcal{S}}} \left[A_{\mathbf{a}} + \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} P_{\mathbf{a}}^{\epsilon'} \Phi \right] (\theta_{n-k} - \theta_n) \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}]; \tag{25}$$

$$\rho_n^{(4)} := \alpha_n \sum_{k=0}^{n} \mu_k \sum_{\mathbf{a}, \mathbf{a}' \in \mathcal{A}^{\mathcal{S}}} \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} \left[P_{\mathbf{a}'}^{\epsilon'} \Phi \theta_n^- - P_{\mathbf{a}}^{\epsilon'} \Phi \theta_{n-k} \right] \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}, \theta_n^- \in \mathcal{R}_{\mathbf{a}'}]. \tag{26}$$

[5]Due to the discontinuity in f, we actually need to consider the analogous form needed to derive [Borkar, 2022, Lemma 5.1]; however, as stated in ibid, the latter's proof mimics that of [Borkar, 2022, Lemma 2.1].

The sum in (21) is our proposed decomposition for $\alpha_n \rho_n$. Note that $\rho_n^{(1)}, \rho_n^{(2)}$, and $\rho_n^{(3)}$ capture the complexities which are only due to experience replay, while $\rho_n^{(4)}$ is also due to the target network.

We now individually study the behaviors of the four terms in the decomposition. We first look at the $\rho_n^{(4)}$ term. For each (s, a) combination, we have

$$P_{\mathbf{a}}^{\epsilon'}(\cdot|s,a)\Phi\theta_{n-k}\,\mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}] = \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s,a) \left[(1-\epsilon') \max_b \phi^T(s',b)\theta_{n-k} + \frac{\epsilon'}{|\mathcal{A}|} \sum_{a' \in \mathcal{A}} \phi^T(s',a')\theta_{n-k} \right].$$

Similarly,

$$P_{\mathbf{a}'}^{\epsilon'}(\cdot|s,a)\Phi\theta_n^-\,\mathbb{1}[\theta_n^- \in \mathcal{R}_{\mathbf{a}'}] = \sum_{s' \in \mathcal{S}} \mathbb{P}(s'|s,a) \left[(1-\epsilon') \max_b \phi^T(s',b)\theta_n^- + \frac{\epsilon'}{|\mathcal{A}|} \sum_{a' \in \mathcal{A}} \phi^T(s',a')\theta_n^- \right].$$

Since $|\max_b \phi^T(s',b)\theta_n^- - \max_b \phi^T(s',b)\theta_{n-k}| \le \max_b |\phi^T(s',b)\theta_n^- - \phi^T(s',b)\theta_{n-k}|$, the above relations now show that

$$\left\| \left[P_{\mathbf{a}'}^{\epsilon'}\Phi\theta_n^- - P_{\mathbf{a}}^{\epsilon'}\Phi\theta_{n-k} \right] \mathbb{1}[\theta_{n-k} \in \mathcal{R}_{\mathbf{a}}, \theta_n^- \in \mathcal{R}_{\mathbf{a}'}] \right\|_\infty \le \|\Phi\theta_{n-k} - \Phi\theta_n^-\|_\infty.$$

However, $\|\Phi\theta_{n-k} - \Phi\theta_n^-\|_\infty \le \|\Phi\theta_n - \Phi\theta_n^-\|_\infty + \|\Phi\theta_{n-k} - \Phi\theta_n\|_\infty$. Further, for any $0 \le k \le \ell$,

$$\begin{aligned} \|\theta_{n-k} - \theta_n\| &\le \sum_{j=n-\ell}^{n-1} \|\theta_j - \theta_{j+1}\| \\ &\le \sum_{j=n-\ell}^{n-1} \alpha_j |\delta_j|\, \|\phi(s_j, a_j)\| \\ &= O(\alpha_{n-\ell}), \end{aligned}$$

where the last relation follows since i) the rewards and feature vectors are bounded (due to (15)), ii) the stepsize sequence (α_n) is non-increasing (see $\mathcal{B}_2$), and iii) $\sup_{n \ge 0} \|\theta_n\|$ and $\sup_{n \ge 0} \|\theta_n^-\|$ are finite. Therefore, $\rho_n^{(4)}/\alpha_n = O(\|\theta_n^- - \theta_n\|) + O(\alpha_{n-\ell}) \to 0$ due to (19) and the fact that $\alpha_{n-\ell} \to 0$. Now, since $\sum_{j=n}^{j=n+m_n} \alpha_j \le CT$ for some constant $C \ge 0$, it follows that $\sup_{0 \le m \le m_n} \|\sum_{j=n}^{n+m} \rho_j^{(4)}\| = O(T \sup_{j \ge n} \|\theta_j^- - \theta_j\| + T\alpha_{n-\ell}) \to 0$, as $n \to \infty$.

Next, consider $\rho_n^{(3)}$. Clearly,

$$\frac{\|\rho_n^{(3)}\|}{\alpha_n} \le \sum_{k=0}^{\ell} \mu_k \sum_{a \in \mathcal{A}^{\mathcal{S}}} \|A_{\mathbf{a}} + \gamma \Phi^T D_{\mathbf{a}}^{\epsilon} P_{\mathbf{a}}^{\epsilon'} \Phi\| \|\theta_{n-k} - \theta_n\|.$$

Again, by arguing as above, we have $\sup_{0 \le m \le m_n} \|\sum_{k=n}^{n+m} \rho_j^{(3)}\| = O(T\alpha_{n-\ell}) \to 0$ as $n \to \infty$.

Finally, we study the asymptotic behaviors of $\rho_n^{(1)}$ and $\rho_n^{(2)}$. Unlike $\rho_n^{(3)}$ and $\rho_n^{(4)}$, though, these terms are not $o(\alpha_n)$, i.e., $\rho_n^{(1)}/\alpha_n$ and $\rho_n^{(2)}/\alpha_n$ do not decay to 0. However, as we now show, there exists a telescopic sum that ensures their cumulative effect over any finite T-length horizon is negligible. Formally,

$$\sum_{j=n}^{n+m} \alpha_j \rho_j^{(1)} = \sum_{k=0}^{\ell} \mu_k \sum_{j=n}^{n+m} [\alpha_{j-k} f(\theta_{j-k}) - \alpha_j f(\theta_j)] \tag{27}$$

$$= \sum_{k=0}^{\ell} \mu_k \Bigg[\sum_{j=n}^{n+k-1} \alpha_{j-k} f(\theta_{j-k}) \tag{28}$$

$$- \sum_{j=n+m+1-k}^{n+m} \alpha_j f(\theta_j) \Bigg], \tag{29}$$

where the intermediate terms get canceled due to their telescopic nature. Note that the two inner summations contain at most ℓ many terms. Combining these statements with the facts that $\sup_{n\geq 0}\|\theta_n\| < \infty$ and that the stepsize sequence (α_n) is non-increasing (see $\mathcal{B}_2$), we get

$$\sup_{m\geq 0}\left\|\sum_{j=n}^{n+m}\alpha_j\rho_j^{(1)}\right\| = O(\alpha_{n-\ell}). \tag{30}$$

Similarly, we have $\sup_{m\geq 0}\|\sum_{j=n}^{n+m}\alpha_j\rho_j^{(2)}\| = O(\alpha_{n-\ell})$.

We can now conclude that $\sum_{0\leq m\leq m_n}\|\sum_{j=n}^{n+m}\alpha_j\rho_j\| \to 0$, as desired. By arguing as in the proof of [Borkar, 2022, Corollary 5.4], the desired claim now follows. □

A.2 Proofs of Lemmas 3 and 5

Proof of of Lemma 3. Let $\theta \in \mathbb{R}^d$ be arbitrary. Since $\{\mathcal{R}_{\mathbf{a}}\}$ partitions $\mathbb{R}^d$, we have $\theta \in \mathcal{R}_{\mathbf{a}}$ for some unique policy $\mathbf{a}$. Now suppose the initial estimates satisfy $\theta_0^- = \theta_0 = \cdots = \theta_{-\ell} = \theta$. Then, (3) shows that

$$\delta_0\phi(s_0,a_0) = \phi(s_0,a_0)r(s_0,a_0,s_0') - \left[\phi(s_0,a_0)\phi^T(s_0,a_0) - \gamma\phi(s_0,a_0)\phi^T(s_0',a_0')\right]\theta.$$

Further, $d_k^\epsilon = d_{\mathbf{a}}^\epsilon$ for all $k = -\ell,\ldots,0$, where d_k^ϵ and $d_{\mathbf{a}}^\epsilon$ are the stationary distributions defined below $\mathcal{B}_1$ and above (12), respectively. Finally, by recalling how s_0, a_0, s_0' and a_0' are sampled from the discussion below $\mathcal{B}_1$, it follows from (8), (9), (12), and (13) that

$$\begin{aligned} f(\theta) &= \mathbb{E}\left[\delta_0\phi(s_0,a_0)\middle|\theta_0^- = \theta, \theta_k = \theta, k = -\ell,\ldots,0\right] \\ &= b_{\mathbf{a}} - A_{\mathbf{a}}\theta. \end{aligned}$$

Note that the above conclusion is not influenced by the choice of the buffer-sampling-distribution μ. The desired result now follows. □

Proof of of Lemma 5. Fix $\theta \in \mathbb{R}^d$. By definition, $h(\theta)$ is a closed convex set. Our first claim is that $b_{\mathbf{a}} - A_{\mathbf{a}}\theta \in h(\theta)$ for each $\mathbf{a} \in \text{supp}(\theta)$. From the convexity of $h(\theta)$, it will then follow that

$$h'(\theta) := \text{co}\left\{b_{\mathbf{a}} - A_{\mathbf{a}}\theta : \mathbf{a} \in \text{supp}(\theta)\right\} \subseteq h(\theta). \tag{31}$$

To see the claim, consider an arbitrary $\mathbf{a} \in \text{supp}(\theta)$. For each $\delta > 0$, $B(\theta,\delta)\cap\mathcal{R}_{\mathbf{a}} \neq \emptyset$ by the definition of supp(θ); take $\theta_\delta \in B(\theta,\delta)\cap\mathcal{R}_{\mathbf{a}}$. Since $B(\theta,\delta_1)\cap\mathcal{R}_{\mathbf{a}} \subseteq B(\theta,\delta_2)\cap\mathcal{R}_{\mathbf{a}}$ for any $\delta_1 \leq \delta_2$, it follows that $\{\theta_\delta : 0 < \delta < \delta'\} \subseteq B(\theta,\delta')$ for any $\delta' > 0$. Hence, $\{b_{\mathbf{a}} - A_{\mathbf{a}}\theta_\delta : 0 < \delta < \delta'\} \subseteq f(B(\theta,\delta'))$ for any $\delta' > 0$. Now, since $\lim_{\delta\to 0}\theta_\delta = \theta$, it follows that $b_{\mathbf{a}} - A_{\mathbf{a}}\theta \in \overline{f(B(\theta,\delta'))} \subseteq \overline{\text{co}}(f(B(\theta,\delta')))$ for any $\delta' > 0$. Hence, $b_{\mathbf{a}} - A_{\mathbf{a}}\theta \in h(\theta)$, as desired.

We now prove that $h(\theta) \subseteq h'(\theta)$. Suppose not. Then there exists $x \in h(\theta)$ such that $x \notin h'(\theta)$. Since the latter is a closed set, there in fact exists some $\delta' > 0$ such that $\|x - y\|_2 \geq \delta'$ for all $y \in h'(\theta)$. Now let $\delta_0 > 0$ be the largest $\delta > 0$ such that $B(\theta,\delta)\cap\mathcal{R}_{\mathbf{a}} \neq \emptyset$ if and only if $\mathbf{a} \in \text{supp}(\theta)$ (the existence of such a δ_0 for $\theta \neq 0$ can be seen from the definition of supp(θ) and the fact that the number of $\mathbf{a}$'s is finite; for $\theta = 0$, take $\delta_0 = \infty$). Pick δ such that $0 < \delta < \min\{\delta_0, \delta'/(2\max_{\mathbf{a}}\|A_{\mathbf{a}}\|_2)\}$. Because $x \in h(\theta)$, we have $x \in \overline{\text{co}}(f(B(\theta,\delta)))$. The closure implies there exists $x' := \sum_{i=1}^m \nu_i(b_{\mathbf{a}_i} - A_{\mathbf{a}_i}\theta_i) \in \text{co}(f(B(\theta,\delta)))$ such that $\|x - x'\| < \delta'/2$, where $m \geq 1$ and, for $1 \leq i \leq m$, $\theta_i \in B(\theta,\delta)$, $\mathbf{a}_i \in \text{supp}(\theta)$, and $\nu_i \in [0,1]$ with $\sum_{i=1}^m \nu_i = 1$. Also, since $\delta < \delta'/(2\max_{\mathbf{a}}\|A_{\mathbf{a}}\|_2)$, we have $\|x' - x''\|_2 < \delta'/2$ for $x'' := \sum_{i=1}^m \nu_i(b_{\mathbf{a}_i} - A_{\mathbf{a}_i}\theta) \in h'(\theta)$. However, this implies $\|x - x''\| < \delta'$, which leads to a contradiction. Hence, it holds that $h(\theta) \subseteq h'(\theta)$. The desired claim follows. □

B MDP and DQN implementation details for Figure 2 and 3b

For Figure 2, the details are as follows.

- MDP and linear state-action-value feature matrix:
  States $\mathcal{S} = \{s_1, s_2\}$, Actions $\mathcal{A} = \{a_1, a_2\}$,

$$\{\mathbb{P}(s'|s,a_1)\}_{s,s'} = \begin{bmatrix} 0.380 & 0.620 \\ 0.786 & 0.214 \end{bmatrix}, \{\mathbb{P}(s'|s,a_2)\}_{s,s'} = \begin{bmatrix} 0.124 & 0.876 \\ 0.426 & 0.574 \end{bmatrix},$$
$$\Phi = \begin{bmatrix} 1.919 & 0.112 \\ 2.581 & -0.659 \\ 1.912 & 1.679 \\ 1.560 & -0.168 \end{bmatrix}, r = \begin{bmatrix} -0.031 \\ 0.785 \\ -0.282 \\ -0.418 \end{bmatrix}, \gamma = 0.9, \epsilon = 0.05.$$

- DQN algorithm implementation details: Replay buffer size: 10, Batch size: 8, Target update duration: 8, Step sizes: $2/n$ at iteration n.

For Figure 3b, the details are as follows.

- MDP and linear action-value feature matrix:
  States $\mathcal{S} = \{s_1, s_2\}$, Actions $\mathcal{A} = \{a_1, a_2\}$,

$$\{\mathbb{P}(s'|s,a_1)\}_{s,s'} = \begin{bmatrix} 0.355 & 0.645 \\ 0.598 & 0.402 \end{bmatrix}, \{\mathbb{P}(s'|s,a_2)\}_{s,s'} = \begin{bmatrix} 0.820 & 0.180 \\ 0.288 & 0.712 \end{bmatrix},$$
$$\Phi = \begin{bmatrix} 0.985 & 0.951 \\ 0.395 & 1.078 \\ -0.904 & 1.276 \\ 0.063 & 1.214 \end{bmatrix}, r = \begin{bmatrix} -0.599 \\ -1.427 \\ 0.658 \\ 0.300 \end{bmatrix}, \gamma = 0.75, \epsilon = 0.1.$$