# On the Sample Complexity of Actor-Critic Method for Reinforcement Learning with Function Approximation

Harshat Kumar*, Alec Koppel†, and Alejandro Ribeiro*

January 31, 2023

**Abstract**

Reinforcement learning, mathematically described by Markov Decision Problems, may be approached either through dynamic programming or policy search. Actor-critic algorithms combine the merits of both approaches by alternating between steps to estimate the value function and policy gradient updates. Due to the fact that the updates exhibit correlated noise and biased gradient updates, only the asymptotic behavior of actor-critic is known by connecting its behavior to dynamical systems. This work puts forth a new variant of actor-critic that employs Monte Carlo rollouts during the policy search updates, which results in controllable bias that depends on the number of critic evaluations. As a result, we are able to provide for the first time the convergence rate of actor-critic algorithms when the policy search step employs policy gradient, agnostic to the choice of policy evaluation technique. In particular, we establish conditions under which the sample complexity is comparable to stochastic gradient method for non-convex problems or slower as a result of the critic estimation error, which is the main complexity bottleneck. These results hold in continuous state and action spaces with linear function approximation for the value function. We then specialize these conceptual results to the case where the critic is estimated by Temporal Difference, Gradient Temporal Difference, and Accelerated Gradient Temporal Difference. These learning rates are then corroborated on a navigation problem involving an obstacle and the pendulum problem which provide insight into the interplay between optimization and generalization in reinforcement learning.

## 1 Introduction

Actor-critic refers to a family of two time-scale algorithms for reinforcement learning where one alternates between policy gradient updates (actor) and action-value function estimation in an online fashion (critic). These approaches form the bedrock of several practical advances in reinforcement learning, as in supply chain management (Giannoccaro and Pontrandolfo, 2002), power systems (Jiang et al., 2014), robotic manipulation (Kober and Peters, 2012), and games of various kinds (Tesauro et al., 1995; Brockman et al., 2016; Mnih et al., 2016; Silver et al., 2017). While their asymptotic stability has been known for decades (Konda and Borkar, 1999; Konda and Tsitsiklis, 2000), their sample complexity is relatively unexplored. In this work, we establish the statistical behavior of actor-critic algorithms for a number of canonical settings, which to our knowledge is the first time a comprehensive accounting has been conducted.

We focus on reinforcement learning problems over possibly continuous state and action spaces, which are defined by a Markov Decision Process (Puterman, 2014): each time, starting from one state, an agent selects an action, and then it transitions to a new state according to a distribution Markov in the current state and action. Then, the environment reveals a reward informing the quality of that decision. The goal of the agent is to select an action sequence which yields the largest expected accumulation of rewards, defined as the value (Bellman, 1954; Bertsekas, 2005). Actor-critic algorithms adapt the merits of reinforcement

*Department of Electrical and Systems Engineering, University of Pennsylvania, Philadelphia, PA 19104
†JPMorgan AI Research, New York, NY

learning algorithms based on approximate dynamic programming with those based on policy search, the two dominant model-free approaches in the literature (Sutton et al., 2017).

For finite spaces, one may obtain the globally optimal policy, and therefore it is possible quantify sample complexity in terms of the gap to the optimal value function (regret) as, e.g., a polynomial function of the cardinality of the state and action spaces – see Jin et al. (2018) and references therein. This is possible because these quantities have finite cardinality; however, in continuous spaces, these analyses break down because policy parameterization is required, and the value function becomes non-convex with respect to the policy parameters (unless it is parameterized by a sufficiently high-dimensional neural model (Wang et al., 2019)).

More specifically, in the *actor step* of actor-critic, stochastic gradient steps with respect to the value function over a parameterized family of policies are conducted. Via the Policy Gradient Theorem (Sutton et al., 2000), the gradient with respect to policy parameters (policy gradient) is the product of two factors: the score function and the $Q$ function. One may employ Monte Carlo rollouts to estimate $Q$-factors, which under careful choice of rollout horizon, can be shown to be unbiased (Paternain, 2018). As a result, linking policy gradient methods to more standard stochastic programming results for non-convex optimization, namely, sublinear $\mathcal{O}(k^{-1/2})$ rates to stationarity have recently been established (Zhang et al., 2019). Doing so, however, requires an inordinate amount of querying to the environment in order to generate trajectory data. In actor-critic, we replace Monte Carlo rollouts with online estimates for the action-value function.

More specifically, in actor-critic, the *critic step* estimates the action-value ($Q$) function through stochastic approximation, i.e., temporal difference (TD) (Sutton, 1988), approaches to solving Bellman's evaluation equation (Watkins and Dayan, 1992; Tsitsiklis, 1994). Combining temporal difference iterations with non-linear function parameterizations may cause instability, as shown by Baird (1995); Tsitsiklis and Van Roy (1997). This motivates the majority of TD algorithms to focus on the case where the $Q$ function is parameterized by a linear basis expansion over given universal features, which is common in practice (Sutton et al., 2017), and can be satisfied by radial basis function (RBF) networks or auto-encoders Park and Sandberg (1991). We consider this setting of universal features given *a priori*.

The asymptotic stability of linear TD algorithms hinges upon dynamical systems tools to encapsulate the mean estimation error sequence – see Borkar and Meyn (2000); Kushner and Yin (2003). By contrast, a number of finite-time characterizations of various TD algorithms have appeared recently, i.e., those based on stochastic fixed point iterations and gradient-based approximations known as gradient temporal difference (GTD) (Sutton et al., 2009a). For TD algorithms, finite-time sublinear rates have been derived both in the case where samples (state-action-reward triples) are independent and identically distributed (i.i.d.) (Dalal et al., 2018b; Bhandari et al., 2018; Lakshminarayanan and Szepesvari, 2018) and when they exhibit Markovian dependence (Srikant and Ying, 2019). Further, the convergence of GTD was established in (Koppel et al., 2017; Tolstaya et al., 2018) by employing coupled supermartingales (Wang et al., 2017a), which permits us to derive the rates of convergence in expectation of GTD as corollaries. As a result, we may explicitly derive the bias due to critic estimation error in terms of the number of critic steps. This is in contrast to the use of an unbiased estimate from a Monte Carlo rollout, as in pure policy gradient methods. We further note that contemporaneously of beginning this work, several analyses of GTD have been developed (Liu et al., 2015; Dalal et al., 2018b, 2020) that refine the rates employed in this analysis; however, these results focus on concentration bounds ("lock-in probability"), a weaker metric of stability than convergence in mean, i.e., convergence in Lebesgue integral implies convergence in measure. Since in this work we focus on the intuitive and broadly interpretable *global convergence* to stationarity in terms of the expected gradient norm of the value function, we seek to employ policy evaluation rates that are compatible with this goal, and defer refined lock-in probability results, for which tighter bounds of convergence on the critic exist, to future work.

**Convergence of Actor-Critic** In this work, we link the behavior of actor-critic to gradient ascent algorithms with biased gradient directions. This bias is controllable and depends on the step-size and number of critic iterations per actor update. We perform this analysis for the setting that samples are i.i.d, which may be explicitly guaranteed through the introduction of a new Monte Carlo rollout step for each actor update. As a result, we establish that actor-critic, independent of any critic method, exhibits convergence to stationary points of the value function that are comparable to stochastic gradient ascent in the non-convex

| Critic Method | Convergence Rate | State-Action Space | Smoothness Assumptions | Algorithm |
|---|---|---|---|---|
| GTD (SCGD) | $O\left(\epsilon^{-3}\right)$ | Continuous | Assumption 3 | Alg 2 |
| GTD (A-SCGD) | $O\left(\epsilon^{-5/2}\right)$ | Continuous | Assumptions 3 and 4 | Alg 3 |
| TD(0) | $O\left(\epsilon^{-2/\sigma}\right)$ | Continuous | None | Alg 4 |
| TD(0) | $O\left(\epsilon^{-2}\right)$ | Finite | None | Alg 4 |

Table 1: Rates of Actor Critic with Policy Gradient Actor updates and different critic-only methods.The term $\sigma$ is the critic stepsize for TD(0) with continuous state-action space, and should be chosen according to conditioning of the feature space (see Section 6.1).

regime. A key distinguishing feature from standard non-convex stochastic programming is that the rates are inherently tied to the bias of the search direction which is determined by the choice of critic scheme. In fact, our methodology is such that a rate for actor-critic can be derived for any critic-only method for which a convergence rate in expectation on the parameters can be expressed. In particular, we establish the rates for actor-critic with temporal difference (TD) (Sutton, 1988) and gradient TD (GTD) (Sutton et al., 2009a) critic steps. Furthermore, we propose an Accelerated GTD (A-GTD) method derived from accelerations of stochastic compositional gradient descent (Wang et al., 2017a), which converges faster than TD and GTD.

In summary, for the continuous spaces, we establish that A-GTD converges faster than GTD, and the effective convergence rate of TD(0) varies as a result of the feature space representation selected *a priori*. In particular, this introduces a trade off between the smoothness assumptions and the rates derived (see Table 1). TD has no additional smoothness assumptions, and it achieves a rate of $O(\epsilon^{-2/\sigma})$. This rate is analogous to the non-convex analysis of stochastic compositional gradient descent when $\sigma$ is equal to 0.5, which is a conservative estimate (see Figure 1). Adding a smoothness assumption, GTD achieves the faster rate of $O(\epsilon^{-3})$. By requiring an additional smoothness assumption, we find that A-GTD achieves the fastest convergence rate of $O(\epsilon^{-5/2})$. For the case of finite state action space, actor critic achieves a convergence rate of $O(\epsilon^{-2})$. Overall, the contribution in terms of sample complexities of different actor-critic algorithms may be found in Table 1.

Relative to existing convergence results, actor-critic is classically studied as a form of two time-scale algorithm (Borkar, 1997), whose asymptotic stability is well-known via dynamical systems (Kushner and Yin, 2003; Borkar, 2009). To wield these approaches to establish finite-time performance, however, concentration probabilities and geometric ergodicity assumptions of the Markov dynamics are required – see Borkar (2009). We obviate these complications by focusing on the case where independent trajectory samples are acquirable through querying the environment, for which recent unbiased sampling procedures gave proved adept (Paternain, 2018; Zhang et al., 2019). Relative to existing finite-time characterizations of actor-critic, (Cai et al., 2019) proposes Neural TD updates, which converges to global optimality under a suitably over-parameterized deep neural network (DNN) and initialization. One quandary is how to find these initializations or design DNN architectures to satisfy these conditions. In separate work, the sample complexity of actor-critic has been established in terms of the value function gradient norm when the critic parameters are estimated with non-linear function approximation in a *batch* fashion (Yang et al., 2018). It is well-known that non-linear function approximators may diverge given by various counterexamples (Baird, 1995; Tsitsiklis and Van Roy, 1997). Our work circumvents this obstacle by considering only well-behaved and well-studied linear function approximation, which includes commonly chosen radial basis function (RBF) networks and auto-encoders fixed at the outset of RL training.

Since the original date of submission, efforts to refine the analysis in this work exist: for instance, relaxations of assumptions on the sampling distribution to allow Markovian dependence (Qiu et al., 2021; Xu et al., 2020; Wu et al., 2020) and augmentations of the critic objective for practical variance reduction (Parisi et al., 2019). However, these works require the Markov transition density to mix at an exponentially fast rate in order to establish convergence. Thus, while i.i.d. sampling may be difficult to justify, exponentially fast mixing often does not hold either, unless algorithm step-sizes are sent to null at an exponential rate. These intricacies have motivated experimental techniques to mitigate correlation among samples using replay buffers

(Wang et al., 2017b) and parallelization of queries to a generative model (Gruslys et al., 2018). However, their exact relationship to mixing rates is opaque. Therefore, for simplicity, in this work we focus on the i.i.d. case.

Moreover, sharper sample complexities for actor-critic have been developed Qiu et al. (2021); Xu et al. (2020); Wu et al. (2020); however, they do not address the possibility of designing alternate policy evaluation schemes than TD(0) updates, and instead focus only on actor-critic in its vanilla form. This is because their perspective is on understanding the sample complexity of actor-critic alone, whereas we provide a unified perspective upon the basis of biased stochastic gradient iteration. In doing so, we are able incorporate a variety of critic updates and illuminate the interplay of problem smoothness, cardinality, and the choice of critic parameterization. In particular, the sample complexity of actor-critic with TD(0) updates for the tabular case given in Corollary 4 matches Xu et al. (2020); Wu et al. (2020), but in continuous spaces, depending on the conditioning of the feature map covariance and other problem smoothness conditions, GTD or A-GTD may yield faster convergence, a facet elsewhere unaddressed in the literature.

Even more recently, efforts have been made to improve upon the rate of convergence by considering regularized MDP's with overparametrized networks (Cayci et al., 2022), single critic step (Olshevsky and Gharesifard, 2022), and single trajectory actor updates (Chen et al., 2022a). Decentralized convergence rates have also been established (Chen et al., 2022b; Zeng et al., 2022). Shen et al. (2020) show that for both i.i.d. and markovian sampling, there is a linear speedup for the decentralized setting whose is bottleneck is the slowest mixing chain. All of the aforementioned results require the assumption that the probability for any action given a state is strictly positive, which we do not require.

We evaluate actor-critic with TD, GTD, and A-GTD critic updates on both a navigation problem and the canonical pendulum problem. For the navigation problem, we find that indeed A-GTD converges faster than both GTD and TD. Interestingly, the stationary point it reaches is worse than GTD or TD. This suggests that the choice of critic scheme illuminates an interplay between optimization and generalization that is less-well understood in reinforcement learning (Boyan and Moore, 1995; Bousquet and Elisseeff, 2002). For the pendulum problem, we also find that A-GTD converges fastest with respect to the gradient norm, which is consistent with our main convergence results. In particular, we again find that the faster convergence in gradient norm results the stationary point having a lower cumulative reward. We additinally consider advantage actor-critic in our simulations (Mnih et al., 2016). A detailed discussion on the results and implications can be found in section 7. The remainder of the paper is organized as follows. Section 2 describes the problem of reinforcement learning and defines common assumptions which we use in our analysis. In section 3, we derive a generic actor-critic algorithm from an optimization perspective and describe how the algorithm would be amended given different policy evaluation methods. The derivation of the convergence rate for generic actor-critic is presented in section 4, and the specific analysis for Gradient, Accelerated Gradient, and vanilla Temporal Difference are characterized in sections 5 and 6.

## 2 Reinforcement Learning

We consider the Reinforcement Learning (RL) problem where an agent moves through a state space $\mathcal{S}$ and takes actions that belong to some action set $\mathcal{A}$, and the state/action spaces are assumed to be continuous compact subsets of Euclidean space: $\mathcal{S} \subset \mathbb{R}^q$ and $\mathcal{A} \subset \mathbb{R}^p$. Every time an action is taken, the agent transitions to its next state that depends only on its current state and action. Moreover, a reward is revealed by the environment. In this situation, the agent would like to accumulate as much reward as possible in the long term, which is referred to as value. Mathematically this problem definition may be encapsulated as a Markov decision process (MDP), which is a tuple $(\mathcal{S}, \mathcal{A}, \mathbb{P}, R, \gamma)$ with Markov transition density $\mathbb{P}(s' \mid s, a) : \mathcal{S} \times \mathcal{A} \to \mathbb{P}(\mathcal{S})$ that determines the probability of moving to state $s'$. Here, $\gamma \in (0, 1)$ is the discount factor that parameterizes the value of a given sequence of actions, which we will define shortly.

At each time $t$, the agent executes an action $a_t \in \mathcal{A}$ given the current state $s_t \in \mathcal{S}$, following a stochastic policy $\pi : \mathcal{S} \to \mathbb{P}(\mathcal{A})$, i.e., $a_t \sim \pi(\cdot \mid s_t)$. Then, given the state-action pair $(s_t, a_t)$, the agent observes a (deterministic) reward $r_t = R(s_t, a_t)$ and transitions to a new state $s'_t \sim \mathbb{P}(\cdot \mid s_t, a_t)$ according to a Markov

transition density. For any policy $\pi$, define the value function $V_\pi : \mathcal{S} \to \mathbb{R}$ as

$$V_\pi(s) := \mathbb{E}_{a_t \sim \pi(\cdot|s_t), s_{t+1} \sim \mathbb{P}(\cdot|s_t,a_t)} \Big( \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s \Big), \tag{1}$$

which is a measure of the long term average reward accumulation discounted by $\gamma$. We can further define the value $V_\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ conditioned on a given initial action as the action-value, or Q function as $Q_\pi(s,a) = \mathbb{E}\big( \sum_{t=0}^{\infty} \gamma^t r_t \mid s_0 = s, a_0 = a \big)$. Given any initial state $s_0$, the goal of the agent is to find the optimal policy $\pi$ that maximizes the long-term return $V_\pi(s_0)$, i.e., to solve the following optimization problem

$$\max_{\pi \in \Pi} \; J(\pi) \;, \text{ where } \qquad J(\pi) := V_\pi(s_0). \tag{2}$$

In this work, we investigate actor-critic methods to solve (2), which is a hybrid RL method that fuses key properties of policy search and approximate dynamic programming. To ground the discussion, we first derive the canonical policy search technique called policy gradient method, and explain how actor-critic augments policy gradient. Begin by noting that to address (2), one must search over an arbitrarily complicated function class $\Pi$ which may include those which are unbounded and discontinuous. To mitigate this issue, we parameterize the policy $\pi$ by a vector $\theta \in \mathbb{R}^d$, i.e., $\pi = \pi_\theta$, yielding RL tools called *policy gradient methods* (Konda and Tsitsiklis, 2000; Bhatnagar et al., 2009; Castro and Meir, 2010). Under this specification, the search over arbitrarily complicated function class $\Pi$ to (2) may be reduced to Euclidean space $\mathbb{R}^d$, i.e., a vector-valued optimization, $\max_{\theta \in \mathbb{R}^d} J(\pi_\theta) := V_{\pi_\theta}(s_0)$. Subsequently, we denote $J(\pi_\theta)$ by $J(\theta)$ for notational convenience.

We now make the following standard assumption on the regularity of the MDP problem and the parameterized policy $\pi_\theta$, which are the same conditions as Zhang et al. (2020), as well as an assumption to bound the state-action feature representation.

**Assumption 1.** *Suppose the reward function $R$ and the parameterized policy $\pi_\theta$ satisfy the following conditions:*

*(i) The absolute value of the reward $R$ is bounded uniformly by $U_R$, i.e., $|R(s,a)| \in [0, U_R]$ for any $(s,a) \in \mathcal{S} \times \mathcal{A}$.*

*(ii) The policy $\pi_\theta$ is differentiable with respect to $\theta$, and the score function $\nabla \log \pi_\theta(a \mid s)$ is $L_\Theta$-Lipschitz and has bounded norm, i.e., for any $(s,a) \in \mathcal{S} \times \mathcal{A}$,*

$$\|\nabla \log \pi_{\theta^1}(a \mid s) - \nabla \log \pi_{\theta^2}(a \mid s)\| \le L_\Theta \cdot \|\theta^1 - \theta^2\|, \quad \textit{for any} \;\; \theta^1, \theta^2, \tag{3}$$

$$\|\nabla \log \pi_\theta(a \mid s)\| \le B_\Theta, \quad \textit{for any} \;\; \theta. \tag{4}$$

Note that the boundedness of the reward function in Assumption 1(i) is standard in policy search algorithms (Bhatnagar et al., 2008, 2009; Castro and Meir, 2010; Zhang et al., 2018). Observe that with $R$, we have the Q function is absolutely upper bounded by $U_R/(1-\gamma)$, since by definition

$$|Q_{\pi_\theta}(s,a)| \le \sum_{t=0}^{\infty} \gamma^t \cdot U_R = \frac{U_R}{1-\gamma}, \quad \text{for any} \;\; (s,a) \in \mathcal{S} \times \mathcal{A}. \tag{5}$$

The same bound also applies for $V_{\pi_\theta}(s)$ for any $\pi_\theta$ and $s \in \mathcal{S}$ and thus the objective $J(\theta)$ which is defined as $V_{\pi_\theta}(s_0)$, satisfies,

$$|V_{\pi_\theta}(s)| \le \frac{U_R}{1-\gamma}, \quad \text{for any } s \in \mathcal{S}, \qquad |J(\theta)| \le \frac{U_R}{1-\gamma}. \tag{6}$$

We note that the conditions (3) and (4) have appeared in recent analyses of policy search (Castro and Meir, 2010; Pirotta et al., 2015; Papini et al., 2018), and are satisfied by canonical policy parameterizations such as

Boltzmann policy (Konda and Borkar, 1999) and Gaussian policy (Doya, 2000). For example, for Gaussian policy[1] in continuous spaces, $\pi_\theta(\cdot \mid s) = \mathcal{N}(\phi(s)^\top \theta, \sigma^2)$, where $\mathcal{N}(\mu, \sigma^2)$ denotes the Gaussian distribution with mean $\mu$ and variance $\sigma^2$ and $\phi(s)$ denotes some state feature representation. Then the score function has the form $[a - \phi(s)^\top \theta]\phi(s)/\sigma^2$, which satisfies (3) and (4) if the feature vectors $\phi(s)$ have bounded norm, the parameter $\theta$ lies some bounded set, and the action $a \in \mathcal{A}$ is bounded.

Generally, the value function is nonconvex with respect to the parameter $\theta$, meaning that obtaining a globally optimal solution to (2) is out of reach unless the problem has additional structured properties, as in phase retrieval (Sun et al., 2016), matrix factorization (Li et al., 2016), and tensor decomposition (Ge et al., 2015), among others. Thus, our goal is to design actor-critic algorithms to attain stationary points of the value function $J(\theta)$. Moreover, we characterize the sample complexity of actor-critic, a noticeable gap in the literature for an algorithmic tool decades old (Konda and Borkar, 1999) at the heart of the recent innovations of artificial intelligence architectures (Silver et al., 2017).

# 3 From Policy Gradient to Actor-Critic

In this section, we derive actor-critic method (Konda and Borkar, 1999) from an optimization perspective: we view actor-critic as a way of doing stochastic gradient ascent with biased ascent directions, and the magnitude of this bias is determined by the number of critic evaluations done in the inner loop of the algorithm. The building block of actor-critic is called policy gradient method, a type of direct policy search, based on stochastic gradient ascent. Begin by noting that the gradient of the objective $J(\theta)$ with respect to policy parameters $\theta$, owing to the Policy Gradient Theorem (Sutton et al., 2000), has the following form:

$$\begin{aligned}
\nabla J(\theta) &= \int_{s\in\mathcal{S}, a\in\mathcal{A}} \sum_{t=0}^{\infty} \gamma^t \cdot p(s_t = s \mid s_0, \pi_\theta) \cdot \nabla \pi_\theta(a \mid s) \cdot Q_{\pi_\theta}(s,a) ds da \qquad (7) \\
&= \frac{1}{1-\gamma} \int_{s\in\mathcal{S}, a\in\mathcal{A}} (1-\gamma) \sum_{t=0}^{\infty} \gamma^t \cdot p(s_t = s \mid s_0, \pi_\theta) \cdot \nabla \pi_\theta(a \mid s) \cdot Q_{\pi_\theta}(s,a) ds da \\
&= \frac{1}{1-\gamma} \int_{s\in\mathcal{S}, a\in\mathcal{A}} \rho_{\pi_\theta}(s) \cdot \pi_\theta(a \mid s) \cdot \nabla \log[\pi_\theta(a \mid s)] \cdot Q_{\pi_\theta}(s,a) ds da \\
&= \frac{1}{1-\gamma} \cdot \mathbb{E}_{(s,a)\sim\rho_\theta(\cdot,\cdot)} \big[\nabla \log \pi_\theta(a \mid s) \cdot Q_{\pi_\theta}(s,a)\big]. \qquad (8)
\end{aligned}$$

This expression follows from rolling the sum forward, repeatedly applying Bellman's evaluation equation, and exploiting the Markov property of the transition kernel, together with multiplying and dividing by $\pi_\theta$ and rewriting the denominator in terms of the score function via the fact that $\nabla_x \log(x) = 1/x$, as in (Sutton et al., 2000; Zhang et al., 2019). In the preceding expression, $p(s_t = s \mid s_0, \pi_\theta)$ denotes the probability of state $s_t$ equals $s$ given initial state $s_0$ and policy $\theta$, which is occasionally referred to as the occupancy measure, or the Markov chain transition density induced by policy $\pi$. Moreover, $\rho_{\pi_\theta}(s) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t p(s_t = s \mid s_0, \pi_\theta)$ is the ergodic distribution associated with the MDP for fixed policy, which is shown to be a valid distribution (Sutton et al., 2000). For future reference, we define $\rho_\theta(s,a) = \rho_{\pi_\theta}(s) \cdot \pi_\theta(a \mid s)$. The derivative of the logarithm of the policy $\nabla \log[\pi_\theta(\cdot \mid s)]$ is usually referred to as the *score function* corresponding to the probability distribution $\pi_\theta(\cdot \mid s)$ for any $s \in \mathcal{S}$.

Next, we discuss how (8) can be used to develop stochastic methods to address (2). Unbiased samples of the gradient $\nabla J(\theta)$ are required to perform the stochastic gradient ascent, which hopefully converges to a stationary solution of the nonconvex maximization. One way to obtain an estimate of the gradient $\nabla J(\theta)$ is to evaluate the score function and $Q$ function at the end of a rollout whose length is drawn from a geometric distribution with parameter $1-\gamma$ (Zhang et al., 2020)[Theorem 4.3]. If the $Q$ function evaluation is unbiased,

[1] We observe that in practice, the action space $\mathcal{A}$ is bounded, which requires a truncated Gaussian policy to be used over $\mathcal{A}$, as in (Papini et al., 2018).

then the stochastic estimate of the gradient $\nabla J(\theta)$ is unbiased as well. We therefore define the stochastic estimate by

$$\hat{\nabla} J(\theta) := \frac{1}{1-\gamma} \hat{Q}_{\pi_\theta}(s_T, a_T) \nabla \log \pi_\theta(a_T | s_T), \tag{9}$$

where the tuple $(s_T, a_T)$ is drawn from end of the geometric rollout of length $T \sim \mathbf{Geom}(1-\gamma)$. Of course, such an approach is very inefficient with respect to samples, as it does not utilize the state action transitions up until the final tuple. Using the entire trajectory for the actor update comes at the cost of a biased gradient estimate. Before we characterize this bias, we will discuss how to evaluate the $Q$ function using the single point estimation for simplicity.

We consider the case where the $Q$ function admits a linear parametrization of the form $\hat{Q}_{\pi_\theta}(s, a) = \xi^\top \varphi(s, a)$, which in the literature on policy search is referred to as the *critic* (Konda and Borkar, 1999), as it "criticizes" the performance of actions chosen according to policy $\pi$. We let $\varphi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^p$ be a (possibly nonlinear) feature map such as a network of radial basis functions or an auto-encoder known *a priori*. The choice to consider the $Q$ function with a linear function approximator comes from the well known convergence results of linear critic-only methods. In contrast, nonlinear function approximators suffer from the possibility of divergence, as is demonstrated by well known counterexamples (Baird, 1995; Tsitsiklis and Van Roy, 1997).

The critic parameter $\xi$ belongs to a bounded set $\xi \in \Xi \subset \mathbb{R}^p$ such that

$$\|\xi\| \leq C_\xi \text{ for all } \xi \in \Xi \tag{10}$$

This is reasonable because (5) guarantees boundeness of the true $Q$ function. The boundedness of the estimate $\hat{Q}$ follows from requiring the feature map $\varphi(s, a)$ to be bounded, an assumption which can be achieved through normalization, which we subsequently state

**Assumption 2.** *For any state action pair $(s, a) \in \mathcal{S} \times \mathcal{A}$, the norm of the feature representation $\varphi(s, a)$ is bounded by a constant $C_\varphi \in \mathbb{R}_+$.*

We also bound the true gradient of the objective function

$$\|\nabla J(\theta_k)\| \leq C_\nabla, \tag{11}$$

which is established by (8) being bounded as a result of $|Q| \leq U_R/(1-\gamma)$ [c.f. (5)] and $\|\nabla \log \pi_\theta(a|s)\| \leq B_\Theta$ [c.f. (4)].

Moreover, for each actor update $k$, we estimate the parameter $\xi_k$ that defines the $Q$ function from an online policy evaluation (critic-only) method after some $T_C(k)$ iterations, where $k$ denotes the number of policy gradient updates. Thus, we may write the stochastic gradient estimate as

$$\hat{\nabla} J(\theta) = \frac{1}{1-\gamma} \xi_k^\top \varphi(s_T, a_T) \nabla \log \pi_\theta(a_T | s_T). \tag{12}$$

If the estimate of the $Q$ function is unbiased, i.e., $\mathbb{E}[\xi_k^\top \varphi(s_T, a_T) \,|\, \theta, s, a] = Q(s, a)$, then $\mathbb{E}[\hat{\nabla} J(\theta) \,|\, \theta] = \nabla J(\theta)$ (c.f. (Zhang et al., 2020)[Theorem 4.3]). Typically, critic-only methods do not give unbiased estimates of the $Q$ function; however, in expectation the rate at which their bias decays is proportional to the number of $Q$ estimation steps. In particular, denote $\xi_*$ as the parameter for which the $Q$ estimate is unbiased:

$$\mathbb{E}[\xi_*^\top \varphi(s, a)] = \mathbb{E}[\hat{Q}_{\pi_\theta}(s, a)] = Q(s, a). \tag{13}$$

Hence, by adding and subtracting the true estimate of the parametrized $Q$ function to (12), we arrive at the fact the policy search direction admits the following decomposition:

$$\hat{\nabla} J(\theta) = \frac{1}{1-\gamma} (\xi_k - \xi_*)^\top \varphi(s_T, a_T) \nabla \log \pi_\theta(a_T | s_T) + \frac{1}{1-\gamma} \xi_*^\top \varphi(s_T, a_T) \nabla \log \pi_\theta(a_T | s_T). \tag{14}$$

The second term is the unbiased estimate of the gradient $\nabla J(\theta)$, whereas the first defines the difference of the critic parameter at iteration $k$ with the true estimate $\xi_*$. For linear parameterizations of the $Q$ function, policy evaluation methods establish convergence in mean of the bias

$$\mathbb{E}[\|\xi_k - \xi_*\|] \leq g(k), \tag{15}$$

where $g(k)$ is some decreasing function. We address cases where the critic bias decays at rate $k^{-b}$ for $b \in (0, 1]$, due to the fact that several state of the art works on policy evaluation may be mapped to the form (15) for this specification (Wang et al., 2017a; Dalal et al., 2018b). We formalize this with the following proposition.

**Proposition 1.** *Given some $b \in (0, 1]$, there exists a constant $L_1 > 0$ such that*

$$\mathbb{E}[\|\xi_k - \xi_*\|] \leq L_1 k^{-b}. \tag{16}$$

*This implies the expected error of the critic parameter is bounded by $O(k^{-b})$.*

Recently, alternate rates have been established as $O(\log k/k)$; however, they concede that $O(1/k)$ rates may be possible (Bhandari et al., 2018; Zou et al., 2019). Thus, we subsume recent sample complexity characterizations of policy evaluation as is described in Proposition 1. Proposition 1 is an intrinsic property of many policy evaluation schemes, and thus permits one to substitute the standard subsampling rates of a Monte Carlo-based estimator for the Q function (as in REINFORCE (Sutton et al., 2000)) with one that is estimated online using, e.g., temporal difference learning. Hence its role is critical in relating the bias of using critic estimators rather than unbiased gradient estimates to the number of critic steps.

More specifically, (14) is nearly a valid ascent direction: it is approximately an unbiased estimate of the gradient $\nabla J(\theta)$ since the first term becomes negligible as the number of critic estimation steps increases. Based upon this observation, we propose the following full trajectory variant of actor-critic method (Konda and Borkar, 1999): run a critic estimator (policy evaluator) for $T_C(k)$ steps, whose output is critic parameters $\xi_k$. We denote the critic estimator by **Critic:**$\mathbb{N} \to \mathbb{R}^p$ which returns the parameter $\xi_k \in \mathbb{R}^p$ after $T_C(k) \in \mathbb{N}$ iterations. Then, simulate a trajectory of length $H(k)$, and update the actor (policy) parameters $\theta$ as:

$$\theta_{k+1} = \theta_k + \eta_k \frac{1}{1-\gamma} \sum_{t=1}^{H(k)} \xi_k^\top \varphi(s_t, a_t) \nabla \log \pi_{\theta_k}(s_t, a_t|\theta_k). \tag{17}$$

Note that we make the number of critic estimation steps and horizon length grow with $k$. Increasing $T$ and $H$ with $k$ allows us to control the bias of the estimate as is seen in Proposition 1 for the critic evaluations and in the following theorem for horizon length.

Now, we will characterize the bias between the gradient estimate using the entire trajectory of length $H(k)$. Let $\tau = \{s_1, a_1, \ldots, s_{H-1}, a_{H-1}, s_H\}$ be a sampled trajectory of length $H$. Define $F_t$ to be the product of the true state action ($Q$) function with the score function evaluated at the tuple $(s_t, a_t)$, namely

$$F_t := Q(s_t, a_t) \nabla_\theta \log \pi_\theta(s_t, a_t). \tag{18}$$

One can consider constructing an estimate of the policy gradient using the entire trajectory of length $H$ by

$$\hat{g}_H = \sum_{t=1}^{H} \gamma^{t-1} F_t. \tag{19}$$

The following theorem establishes the bias between the true policy gradient and the finite horizon estimate.

**Theorem 1.** *Let Assumption 1 be in effect. Then it is true that for some finite $C_1$,*

$$\|\mathbb{E}_\tau [\hat{g}_H] - \nabla_\theta J(\theta)\| \leq \gamma^{H-1} C_1.$$

*Proof.* First we will show that $\mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}F_t\right] = \nabla_\theta J(\theta)$. We let $\Pr(s_t = s|s_1)$ denote the probability the state at time $t$ is equal to $s$ given the initial state $s_1$.

$$\begin{aligned}
\mathbb{E}\left[\sum_{t=1}^{\infty}\gamma^{t-1}F_t\right] &= \sum_{t=1}^{\infty}\gamma^{t-1}\int_{\mathcal{S}}\mathbb{E}\left[F_t|s_t=s\right]\Pr\left(s_t=s|s_1\right)\mathrm{d}s \\
&= \sum_{t=1}^{\infty}\gamma^{t-1}\int_{\mathcal{S}}\int_{\mathcal{A}}Q(s,a)\nabla_\theta \log\pi_\theta(s,a)\mathrm{d}a\Pr(s_t=s|s_1)\mathrm{d}s \\
&= \int_{\mathcal{S}}\int_{\mathcal{A}}Q(s,a)\nabla_\theta \log\pi_\theta(s,a)\mathrm{d}a\sum_{t=1}^{\infty}\gamma^{t-1}\Pr(s_t=s|s_1)\mathrm{d}s \\
&= \int_{\mathcal{S}}\int_{\mathcal{A}}Q(s,a)\nabla_\theta \log\pi_\theta(s,a)\mathrm{d}a\rho^{\pi_\theta}(s)\mathrm{d}s \\
&= \mathbb{E}_{s\sim\rho^{\pi_\theta}(s)}\left[\int_{\mathcal{A}}Q(s,a)\nabla_\theta \log\pi_\theta(s,a)\mathrm{d}a\right] \\
&= \mathbb{E}_{s\sim\rho^{\pi_\theta}(s),a\sim\pi_\theta(s,\cdot)}\left[Q(s,a)\nabla_\theta\log\pi_\theta(s,a)\right] \\
&= \nabla_\theta J(\theta)
\end{aligned} \tag{20}$$

By Fubini's Theorem, we are able to exchange the summation and integrals due to the regularity assumptions. Let $\hat{g}_\infty = \sum_{t=1}^{\infty}\gamma^{t-1}F_t$. Then

$$\hat{g}_\infty - \hat{g}_H = \gamma^{H-1}\sum_{t=0}^{\infty}\gamma^t F_{t+H+1} \tag{21}$$

By the regularity assumptions, we can bound $F_t$ by $U_R B_\Theta/(1-\gamma)$. As such, we establish the bound $\sum_{t=0}^{\infty}\gamma^t F_{t+H+1} \leq \sum_{t=0}^{\infty}\gamma^t U_R B_\Theta/(1-\gamma) \leq U_R B_\Theta/(1-\gamma)^2 =: C_1 \leq \infty$ Taking the norm of the expectation completes the proof. □

Theorem 1 holds under the assumption that the true $Q$ function is accessible. Of course, only a biased version of the critic is available through the uses of a critic, as described before. The algorithm we propose is the actor-critic variant of the finite horizon gradient estimate. The actor parameter update takes the following form:

$$\theta_{k+1} = \theta_k + \eta_k\hat{g}_H^{AC} = \theta_k + \frac{1}{1-\gamma}\eta_k\sum_{t=1}^{H(k)}\gamma^{t-1}\xi_k^\top\varphi(s_t,a_t)\nabla\log\pi_{\theta_k}(s_t,a_t|\theta_k). \tag{22}$$

The following theorem characterizes the bias of the stochastic gradient estimate.

**Theorem 2.** *Let Assumptions 1 and 2 be in effect. Then, when proposition 1 is in effect, it is true that for a horizon of length $H$ and $T$ critic evaluations,*

$$\left\|\mathbb{E}_\tau\left[\hat{g}_H^{AC}\right] - \nabla_\theta J(\theta)\right\| \leq C_1\gamma^H + C_2 T^{-b}$$

*Proof.* Let $F_{AC,t} := \xi_k^\top\varphi(s_t,a_t)\nabla_\theta\log\pi_\theta(s_t,a_t)$. Then

$$\begin{aligned}
\mathbb{E}_\tau\left[\hat{g}_\infty^{AC}\right] &= \mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}F_{AC,t}\right] \\
&= \mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}\left(F_t + F_{AC,t} - F_t\right)\right] \\
&= \mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}F_t\right] + \mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}\left(F_{AC,t} - F_t\right)\right] \\
&= \nabla_\theta J(\theta) + \mathbb{E}_\tau\left[\sum_{t=1}^{\infty}\gamma^{t-1}\left(F_{AC,t} - F_t\right)\right]
\end{aligned} \tag{23}$$

**Algorithm 1** Generic Actor-Critic

**Require:**

$s_0 \in \mathbb{R}^n$, $\theta_0$, $\xi_0$, stepsize $\{\eta_k\}$, Policy evaluation method **Critic**: $\mathbb{N} \to \mathbb{R}^p$, $\gamma \in (0,1)$

1: **for** $k = 1, \dots$ **do**

2: $\quad \xi_k \leftarrow$ **Critic**$(T_C(k))$ [e.g. See Alg 2-4]

3: $\quad \theta_{k+1} \leftarrow \theta_k + \frac{1}{1-\gamma}\eta_k \sum_{t=1}^{H(k)} \xi_k^\top \varphi(s_t, a_t) \nabla \log \pi_{\theta_k}(s_t, a_t|\theta_k)$

The final term can be considered an error term. Consider the difference

$$F_{AC,t} - F_t = \left(Q(s_t, a_t) - \xi_k^\top \varphi(s_t, a_t)\right) \nabla \log \pi_\theta(s_t, a_t). \tag{24}$$

Let $Q(s_t, a_t) = \xi_*^\top \varphi(s_t, a_t)$. Then by assumptions 1 and 2 and proposition 1,

$$|F_{AC,t} - F_t| \leq T^{-b} L_1 C_\varphi B_\Theta \tag{25}$$

This implies

$$\left\|\hat{g}_\infty^{AC} - \nabla_\theta J(\theta)\right\| \leq T^{-b} L_1 C_\varphi B_\Theta \frac{1}{1-\gamma} = C_2 T^{-b} \tag{26}$$

Following the same logic as Theorem 1, we can bound the difference between the finite horizon estimate and the infinite horizon actor-critic estimate by

$$\|\hat{g}_\infty^{AC} - \hat{g}_H^{AC}\| \leq C_1 \gamma^{H-1}. \tag{27}$$

We evoke triangle inequality to complete the proof.

$$\|\hat{g}_\infty - \hat{g}_H^{AC}\| = \|\hat{g}_\infty - \hat{g}_\infty^{AC} + \hat{g}_\infty^{AC} - \hat{g}_H^{AC}\| \leq \|\hat{g}_\infty - \hat{g}_\infty^{AC}\| + \|\hat{g}_\infty^{AC} - \hat{g}_H^{AC}\| \leq C_1 \gamma^{H-1} + C_2 T^{-b}. \tag{28}$$

This concludes the proof. □

The fact that the estimate $\hat{g}_H^{AC}$ is bounded comes from the fact that $\hat{g}_\infty^{AC}$ is bounded. We formalize this for use in the analysis

$$\mathbb{E}(\|\hat{g}_H^{AC}\|) \leq \mathbb{E}(\|\hat{g}_\infty^{AC}\|) \leq \frac{C_\varphi C_\xi B_\Theta}{(1-\gamma)} =: \sigma, \tag{29}$$

where $C_\varphi$, $C_\xi$ and $B_\Theta$ come from Assumption 2, (10) and Assumption 1 (ii) respectively.

Theorem 2 establishes the bias on the stochastic gradient update. The bias can be decreased by increasing $T$, the number of critic update steps per each actor step, and $H$, the horizon for the actor update. In our main result, we will set both of these quantities to grow linearly with $k$, meaning that we decrease the bias with each actor update step (see Theorem 3). In our numerical results, we show that selecting a large enough constant $T$ and $H$ is sufficient(see Section 7) .

We summarize the aforementioned procedure, which is agnostic to particular choice of critic estimator, as Algorithm 1. We acknowledge that the actor-critic algorithm proposed in Algorithm 1 differs from Konda and Borkar (1999) in that rather than updating the actor and critic in tandem, the critic learns the state-action (Q) function from scratch at each update of the actor algorithm. The classical version of the algorithm can be recovered by setting $T_C(k) = 1$ and initializing the critic parameter to the previous step. Existing convergence proofs of this format are limited to asymptotic convergence only, where the critic steps at a faster learning rate than the actor. As such, this batch-type approach emulates this behavior, as the critic must learn something meaningful before the actor can update. As such, one might relate our work to Yang et al. (2018); however, unlike their work, we are not only able to prove convergence to a stationary point of the original objective by increasing the number of critic evaluations at each actor step rather than keeping it fixed, but also, we use the entire trajectory rather than a single state action pair sampled from the discounted state distribution.

**Examples of Critic Updates** We note that **Critic:**$\mathbb{N} \to \mathbb{R}^p$ admits two canonical forms: temporal difference (TD) (Sutton, 1988) and gradient temporal difference (GTD)-based estimators (Sutton et al., 2008). The TD update for the critic is given as

$$\delta_t = r_t + \gamma \xi_t^\top \varphi(s_t', a_t') - \xi_t^\top \varphi(s_t, a_t) \ , \quad \xi_{t+1} = \xi_t + \alpha_t \delta_t \varphi(s_t, a_t) \tag{30}$$

whereas for the GTD-based estimator for the critic, we consider the update

$$\begin{aligned} \delta_t &= r_t + \gamma \xi_t^\top \varphi(s_t', a_t') - \xi_t^\top \varphi(s_t, a_t) \ , \quad z_{t+1} = (1-\beta_t) z_t + \beta_t \delta_t, \\ \xi_{t+1} &= \xi_t - 2\alpha_t z_{t+1} [\gamma \varphi(s_t', a_t') - \varphi(s_t, a_t)] \end{aligned} \tag{31}$$

We further analyze a modification of GTD updates proposed by (Wang et al., 2017a) that incorporates an extrapolation technique to reduce bias in the estimates and improve error dependency, which is distinct from accelerated stochastic approximation with Nesterov Smoothing (Nesterov, 1983). With $y_0 = 0$ and $z_t$ defined for $t = 1, \ldots,$ the accelerated GTD (A-GTD) update becomes

$$\begin{aligned} \xi_{t+1} &= \xi_t - 2\alpha_t (\gamma \varphi(s_t', a_t') - \varphi(s_t, a_t)) y_t \\ z_{t+1} &= -\left(\frac{1}{\beta_t} - 1\right) \xi_t + \frac{1}{\beta_t} \xi_{t+1} \\ y_{t+1} &= (1-\beta_t) y_t + \beta_t (r(s_t, a_t) + z_{t+1}^\top (\gamma \varphi(s_t', a_t') - \varphi(s_t, a_t)) \end{aligned} \tag{32}$$

Subsequently, we shift focus to characterizing the mean convergence of actor-critic method given any policy evaluation method satisfying (15) in Section 4. Then, we specialize the sample complexity of actor-critic to the cases associated with critic updates (30) - (32), which we respectively call Classic (Algorithm 4), Gradient (Algorithm 2), and Accelerated Actor-Critic (Algorithm 3).

**Remark 1.** *We wish to emphasize that a major advantage of this generic characterization of actor-critic admits the ability to interchange critic only methods to estimate the state-action (Q) function. The merit is twofold, as it can extend to faster convergence rates and fewer assumptions. In particular, recent works have shown tighter sample complexity bounds for critic-only methods for convergence in probability, which suggests that existing bounds on convergence in expectation are not necessarily tight. Furthermore, so long as the convergence of the critic takes the form of Proposition 1, the i.i.d. assumption for the critic can be lifted. The general conditions for stability of trajectories with Markov dependence, i.e., negative Lyapunov exponents for mixing rates, may be found in (Meyn and Tweedie, 2012).*

# 4 Convergence Rate of Generic Actor-Critic

In this section, we derive the rate of convergence in expectation for the variant of actor-critic defined in Algorithm 1, which is agnostic to the particular choice of policy evaluation method used to estimate the $Q$ function used in the actor update. Unsurprisingly, we establish that the rate of convergence in expectation for actor-critic depends on the critic update used. Therefore, we present the main result in this paper for any generic critic method. Thereafter, we specialize this result to two well-known choices of policy evaluation previously described (30) - (31), as well as a new variant that employs acceleration (32).

We begin by noting that under Assumption 1, one may establish Lipschitz continuity of the policy gradient $\nabla J(\theta)$ (Zhang et al., 2020)[Lemma 4.2].

**Lemma 1** (Lipschitz-Continuity of Policy Gradient)**.** *The policy gradient $\nabla J(\theta)$ is Lipschitz continuous with some constant $L > 0$, i.e., for any $\theta^1, \theta^2 \in \mathbb{R}^d$*

$$\|\nabla J(\theta^1) - \nabla J(\theta^2)\| \leq L \cdot \|\theta^1 - \theta^2\|. \tag{33}$$

This lemma allows us to establish an approximate ascent for the objective sequence $\{J(\theta_k)\}$.

**Lemma 2.** *Consider the actor parameter sequence defined by Algorithm 1. Further let Assumptions 1 and 2 be in effect. Define the probability space $(\Omega, \mathcal{F}, P)$. Further, let $\mathcal{F}_k$ be the $\sigma$-algebra generated by the set $\{s_u, a_u, \theta_u\}_{u<k}$, that is the states, actions, and policy parameters until time $k$. Then, the sequence $\{J(\theta_k)\}$ satisfies the inequality*

$$\mathbb{E}[J(\theta_{k+1}) \mid \mathcal{F}_k] \geq J(\theta_k) + \eta_k \|\nabla J(\theta_k)\|^2 - \eta_k C_\nabla C_1 \gamma^{H(k)-1} - \eta_k C_\nabla C_2 T(k)^{-b} - L\sigma^2 \eta_k^2. \tag{34}$$

*where $C_1$ and $C_2$ come from Theorem 2.*

*Proof.* See Appendix 8.1 □

From (34) (Lemma 2), consider taking the total expectation

$$\mathbb{E}[J(\theta_{k+1})] \geq \mathbb{E}[J(\theta_k)] + \eta_k \mathbb{E}[\|\nabla J(\theta_k)\|^2] - \eta_k C_\nabla C_1 \gamma^{H(k)-1} - \eta_k C_\nabla C_2 T(k)^{-b} - L\sigma^2 \eta_k^2. \tag{35}$$

This almost describes an ascent of $J(\theta_k)$. Because the norm of the gradient is non-negative, if the latter three terms were removed, an argument could be constructed to show that in expectation, the gradient converges to zero. Unfortunately, both the error of the finite horizon estimate and the critic error complicate the picture. However, we know that the error goes to zero in expectation as the number of critic steps and the horizon length increase. Thus, we leverage this property to derive the sample complexity of actor-critic (Algorithm 1).

We now present our main result, which is the convergence rate of actor-critic method when the algorithm remains agnostic to the particular choice of critic scheme. We characterize the rate of convergence by the smallest number $K_\epsilon$ of actor updates $k$ required to attain a value function gradient smaller than $\epsilon$, i.e. for $\epsilon > 0$,

$$K_\epsilon = \min\{k : \inf_{0 \leq m \leq k} \|\nabla J(\theta_m)\|^2 < \epsilon\}. \tag{36}$$

**Theorem 3.** *Suppose the actor step-size satisfies $\eta_k = k^{-a}$ for $a > 0$ and the critic update satisfies Proposition 1. Further let $T_C(k) = k + 1 \cdot \mathbf{1}(b = 1)$, and $H(k) = k$. Then the actor sequence defined by Algorithm 1 satisfies*

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-1/\ell}\right) \text{ , where } \ell = \min\{a, 1-a, b\} \tag{37}$$

*Minimizing over $a$ yields actor step-size $\eta_k = k^{-1/2}$. Moreover, depending on the rate $b$ of attenuation of the critic bias [cf. (15)], the resulting sample complexity is:*

$$K_\epsilon \leq \begin{cases} \mathcal{O}\left(\epsilon^{-1/b}\right) & \text{if } b \in (0, 1/2) \\ \mathcal{O}\left(\epsilon^{-2}\right). & \text{if } b \in (1/2, 1] \end{cases} \tag{38}$$

*Proof.* See Appendix 8.2 □

The analysis of Lemma 2 and Theorem 3 do not make any assumptions on the size of the state action space. Additionally, the result describes the number of actor updates required. The number of critic updates required is simply the $K_\epsilon^{\text{th}}$ triangular number, that is $\binom{K_\epsilon+1}{2}$. These results connect actor-critic algorithms with the behavior of stochastic gradient method for finding the root of a non-convex objective. Under additional conditions, actor-critic with TD updates for the critic step attains a $O(\epsilon^{-2})$ rate. However, under milder conditions on the state and action spaces but more stringent smoothness conditions on the reward function, using GTD updates for the critic yields $O(\epsilon^{-3})$ rates. These results are formally derived in the following subsections. We further note that contemporaneously of beginning this work, several refined analyses of TD and GTD have been developed (Dalal et al., 2018b, 2020) that focus on concentration bounds ("lock-in probability"), a weaker metric of stability than convergence in mean, i.e., convergence in Lebesgue integral implies convergence in measure. In this work, we focus on *global convergence* to stationarity in terms of the expected gradient norm of the value function, and thus employ policy evaluation rates that are compatible with this goal, i.e., rates in the form of attenuation of mean square error. We defer the study of lock-in probabilities to future work.

**Remark 2.** *We note that it may be possible to establish convergence in terms of asymptotic covariance or the Hessian around a stationary point, as in (Thoppe and Borkar, 2019), and thus obtain a sharper characterization of the limit points of actor-critic. However, doing so pre-supposes that the algorithm settle to a neighborhood of a local extrema, and would require a Hessian parameterization that is only* locally *valid. Hence sharper global convergence characterizations, to our knowledge, are beyond reach. In this work, our intention is to establish the* global *sample complexity of actor-critic type algorithms, and leave strengthening the local rates using, e.g., techniques developed in (Thoppe and Borkar, 2019), to future work.*

# 5 Rates of Gradient and Accelerated Actor-Critic

In this section, we show how Algorithm 1 can be applied to derive the rate of actor-critic methods using Gradient Temporal Difference (GTD) as the critic update. Thus, we proceed with deriving GTD-style updates through links to compositional stochastic programming (Wang et al., 2017a) which is also the perspective we adopted to derive rates in the previous section. For simplicity in notation, we let $Q$ stand for $Q_{\pi_\theta}$. Begin by recalling that any critic method seeks a fixed point of the Bellman evaluation operator:

$$(T^{\pi_\theta}Q)(s,a) \triangleq r(s,a) + \gamma \mathbb{E}_{s'\in\mathcal{S},a'\sim\pi_\theta(s')}[Q(s',a') \mid s,a] \tag{39}$$

Since we focus on parameterizations of the $Q$ function by parameter vectors $\xi \in \mathbb{R}^d$ with some fixed feature map $\varphi$ which is learned *a priori*, the Bellman operator simplifies

$$T^{\pi_\theta}Q_\xi(s,a) = \mathbb{E}_{s'\in\mathcal{S},a'\sim\pi_\theta(s')}[r(s,a) + \gamma \xi^\top \varphi(s',a')|s,a] \tag{40}$$

The solution of the Bellman equation is its fixed point: $T^\pi Q(s,a) = Q(s,a)$ for all $s \in \mathcal{S}, a \in \mathcal{A}$. Thus, we seek $Q$ functions that minimize the (projected) Bellman error

$$\min_{\xi\in\Xi} \|\Pi T^{\pi_\theta}Q_\xi - Q_\xi\|_\mu^2 =: F(\xi). \tag{41}$$

where $\Xi \subseteq \mathbb{R}^p$ is a closed and convex feasible set. The Bellman error quantifies distance from the fixed point for a given $Q_\xi$. Here the projection and $\mu$-norm are respectively defined as

$$\Pi\hat{Q} = \arg\min_{\mathrm{f}\in\mathcal{F}} \|\hat{Q} - \mathrm{f}\|_\mu \ , \qquad \|Q\|_\mu^2 = \int Q^2(s,a)\mu(\mathrm{d}s,\mathrm{d}a), \tag{42}$$

This parameterization of $Q$ implies that we restrict the feasible set – which is in general $B(\mathcal{S},\mathcal{A})$, the space of bounded continuous functions whose domain is $\mathcal{S}\times\mathcal{A}$ – to be $\mathcal{F} = \{Q_\xi : \xi \in \Xi \subset \mathbb{R}^d\}$ (as in (Maei et al., 2010)). Without this parameterization, one would require searching over $B(\mathcal{S},\mathcal{A})$, whose complexity scales with the dimension of the state and action spaces (Bellman, 1957), which is costly when dimensions are large, and downright impossible for continuous spaces (Powell, 2007).

Under certain mild conditions drawing tools from functional analysis, we can define a projection over a class of functions such that $\Pi\hat{Q} = \hat{Q}$. For example, Radial-Basis-Function (RBF) networks have been shown to be capable of approximating arbitrarily well functions in $L^p(\mathbb{R}^r)$ (Park and Sandberg, 1991, Theorem 1). Further, neural networks with one hidden layer and sigmoidal activation functions are known to approximate arbitrarily well continuous functions on the unit cube (Cybenko, 1989, Theorem 1).

By the definition of the $\mu$-norm, we can write $F$ [cf. (41)] as an expectation

$$F(\xi) = \mathbb{E}[(T^{\pi_\theta}Q_\xi - Q_\xi)^2]. \tag{43}$$

As such, we replace the Bellman operator in (43) with (40) to obtain

$$F(\xi) = \mathbb{E}_{s,a\sim\pi_\theta(s)}\{(\mathbb{E}_{s',a'\sim\pi_\theta(s')}[r(s,a) + \gamma\xi^\top\varphi(s',a')|s,a\sim\pi_\theta(s)] - \xi^\top\varphi(s,a))^2\}. \tag{44}$$

Pulling the last term into the inner expectation, $F(\xi)$ can be written as the function composition $F(\xi) = (f \circ g)(x) = f(g(x))$, where $f : \mathbb{R} \to \mathbb{R}$ and $g : \mathbb{R}^p \to \mathbb{R}$ take the form of expected values

$$f(y) = \mathbb{E}_{(s,a)}[f_{(s,a)}(y)] \,, \qquad g(\xi) = \mathbb{E}_{(s',a')}[g_{(s',a')}(\xi) \mid s, a \sim \pi_\theta(s)], \tag{45}$$

where

$$f_{(s,a)}(y) = y^2 \,, \qquad g_{(s',a')}(\xi) = r(s,a) + \gamma \xi^\top \varphi(s',a') - \xi^\top \varphi(s,a). \tag{46}$$

Because $F(\xi)$ can be written as a nested expectations of convex functions, we can use Stochastic Compositional Gradient Descent (SCGD) for the critic update (Wang et al., 2017a). This requires the computation of the sample gradients for both $f$ and $g$ in (45)

$$\nabla f_{(s,a)}(y) = 2y \,, \qquad \nabla g_{(s',a')}(\xi) = \gamma \varphi(s',a') - \varphi(s,a). \tag{47}$$

The specification of SCGD to the Bellman evaluation error (44) yields the GTD updates (31) defined in Section 3 – see (Sutton et al., 2008) for further details. We now turn to establishing the convergence rate in expectation for Algorithm 1 (substituting Algorithm 2 for the **Critic**$(k)$) step using Theorem 3. Doing so requires the conditions of Theorem 3 from Wang et al. (2017a) to be satisfied, which we subsequently state.

**Assumption 3.**

*(i) The outer function $f$ is continuously differentiable, the inner function $g$ is continuous, the critic parameter feasible set $\Xi$ is closed and convex, and there exists at least one optimal solution to problem (41), namely $\xi^* \in \Xi$*

*(ii) The sample first order information is unbiased. That is,*

$$\mathbb{E}[g_{(s_0',a_0')}(\xi) \mid s_0, a_0 \sim \pi_\theta(s_0)] = g(\xi)$$

*(iii) The function $\mathbb{E}[g(\xi)]$ [cf. (46)] is $C_g$-Lipshitz continuous and the samples $g(\xi)$ and $\nabla g(\xi)$ have bounded second moments*

$$\mathbb{E}[\|\nabla g_{(s_0',a_0')}(\xi)\|^2 \mid s_0, a_0 \sim \pi_\theta(s_0)] \leq C_g, \qquad \mathbb{E}[\|g_{(s_0',a_0')}(\xi) - g(\xi)\|^2] \leq V_g$$

*(iv) The $f_{(s,a)}(y)$ has a Lipschitz continuous gradient such that*

$$\mathbb{E}[\|\nabla f_{(s_0,a_0)}(y)\|^2] \leq C_f \qquad \|\nabla f_{(s_0,a_0)}(y) - f_{(s_0,a_0)}(\bar{y})\| \leq L_f \|y - \bar{y}\|$$

*for all $y, \bar{y} \in \mathbb{R}$*

*(v) The projected Bellman error is strongly convex with respect to the critic parameter $\xi$ in the sense that there exists a $\lambda$ such that*

$$\nabla^2 F(\xi) \succeq \lambda I$$

The first part of Assumption 3(i) is trivially satisfied by the forms of $f$ and $g$ in (46). Assumption 3(ii) requires that the state-action pairs used to update the critic parameter to be independently and identically distributed (i.i.d.), which is a common assumption unless one focuses on performance along a *single trajectory*. Doing so requires tools from dynamical systems under appropriate mixing conditions on the Markov transition density (Borkar, 2009; Antos et al., 2008), which we obviate here for simplicity and to clarify insights. We note that the sample complexity of policy evaluation along a trajectory has been established by Bhandari et al. (2018), but remains open for policy learning in continuous spaces. Moreover, i.i.d. sampling yields unbiasedness of certain gradient estimators and second-moment boundedness which are typical for stochastic optimization (Bottou, 1998). We note that these conditions come directly from Wang et al. (2017a) – here we translate them to the reinforcement learning context.

**Algorithm 2** Critic: Gradient Temporal Difference (GTD)

**Require:**

$s_0 \in \mathbb{R}^n$, $\theta$, $\xi_0$, stepsizes $\{\alpha_t\} \subset \mathbb{R}^+, \{\beta_t\} \subset (0,1]$ which satisfy $\frac{\alpha_t - 1}{\beta_t} \to 0$, Horizon $T_C$

**for** $t = 0, \ldots, T_C - 1$ **do**

  Sample $s_t$ from the ergodic distribution and draw action $a_t \sim \pi^\theta$

  Observe next state $s'_t \sim \mathbf{P}(s_t, a_t, s'_t)$ and observe reward $r_t$

  $\delta_t = r_t + \xi_t^\top (\gamma\varphi(s'_t, a'_t) - \varphi(s_t, a_t))$

  $z_{t+1} = (1 - \beta_t) z_t + \beta_t \delta_t$

  Update Critic:

$$\xi_{t+1} = \xi_t - 2\alpha_t z_{t+1}[\gamma\varphi(s'_t, a'_t) - \varphi(s_t, a_t)]$$

**Algorithm 3** Critic: Accelerated Gradient Temporal Difference (AGTD)

**Require:**

$s_0 \in \mathbb{R}^n$, $\theta$, $\xi_0$, stepsizes $\{\alpha_t\} \subset \mathbb{R}^+, \{\beta_t\} \subset (0,1]$ which satisfy $\frac{\alpha_t - 1}{\beta_t} \to 0$, Horizion $T_C$

Initialize $y_0 \leftarrow 0$

**for** $t = 0, \ldots, T_C(k) - 1$ **do**

  Sample $s_t$ from the ergodic distribution and draw action $a_t \sim \pi^\theta$

  Observe next state $s'_t \sim \mathbf{P}(s_t, a_t, s'_t)$ and observe reward $r_t$

  Update Critic:

$$\xi_{t+1} = \xi_t - 2\alpha_t(\gamma\varphi(s', a') - \varphi(s, a))y_t$$

  Update auxiliary Critic parameters $y_t$ and $z_t$

$$z_{t+1} = -\left(\frac{1}{\beta_t} - 1\right)\xi_t + \frac{1}{\beta_t}\xi_{t+1}$$
$$y_{t+1} = (1 - \beta_t)y_t + \beta_t(r(s, a) + z_{t+1}^\top(\gamma\varphi(s', a') - \varphi(s, a))$$

We further require $F(\xi)$ to be strongly convex, so that Wang et al. (2017a)[Theorem 3 and Theorem 7] holds. Consider the Hessian

$$\nabla^2 F(\xi) = \mathbb{E}_{s,a}\left[\mathbb{E}_{s',a'}\left[\gamma\varphi(s', a') - \varphi(s, a)|s, a\right]^\top \mathbb{E}_{s',a'}\left[\gamma\varphi(s', a') - \varphi(s, a)|s, a\right]\right]. \tag{48}$$

Due to its structure, and the i.i.d. assumption, the Hessian $\nabla^2 F(\xi)$ is known to be positive definite Bertsekas et al. (1995); Dalal et al. (2018b). We can now combine the convergence result (Theorem 3) from Wang et al. (2017a) with Theorem 3 to establish the rate of actor-critic with GTD updates for the critic, through connecting GTD and SCGD. We summarize the resulting method as Algorithm 2, which we call Gradient Actor-Critic.

**Corollary 1.** *Consider the actor parameter sequence defined by Algorithm 2. If the stepsize $\eta_k = k^{-1/2}$ and the critic stepsizes are $\alpha_t = 1/t\sigma$ and $\beta_t = 1/t^{2/3}$, then we have the following bound on $K_\epsilon$ defined in* (36)*:*

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-3}\right). \tag{49}$$

*Proof.* Here we invoke (Wang et al., 2017a, Theorem 3) which characterizes the rate of convergence for the critic parameter

$$\mathbb{E}[\|\xi_k - \xi_*\|^2] \leq \mathcal{O}\left(k^{-2/3}\right). \tag{50}$$

Applying Jensen's inequality, we have

$$\mathbb{E}[\|\xi_k - \xi_*\|]^2 \leq \mathbb{E}[\|\xi_k - \xi_*\|^2] \leq \mathcal{O}\left(k^{-2/3}\right), \tag{51}$$

Taking the square root gives us

$$\mathbb{E}[\|\xi_k - \xi_*\|] \leq \mathcal{O}\left(k^{-1/3}\right). \tag{52}$$

Therefore, $b = 1/3$ (c.f. Proposition 1) in Theorem 3, which determines the $\mathcal{O}\left(\epsilon^{-3}\right)$ rate on $K_\epsilon$ in the preceding expression. □

Unsurprisingly, with additional smoothness assumptions, it is possible to obtain faster convergence through accelerated variants of GTD. The corresponding actor-critic method with Accelerated GTD updates is given by substituting Algorithm 3 for **Critic**$(k)$ in Algorithm 1, which we call Accelerated Actor-Critic. The validity of accelerated rates, aside from Assumption 3, requires imposing that the inner expectation has Lipschitz gradients and that sample gradients have boundedness properties which are formally stated below.

**Assumption 4.**

*(i) There exists a constant scalar $L_g > 0$ such that*

$$\|\nabla \mathbb{E}_{s',a' \sim \pi_\theta(s')}[g(\xi_1)] - \nabla \mathbb{E}_{s',a' \sim \pi_\theta(s')}[g(\xi_2)]\| \leq L_g \|\xi_1 - \xi_2\|, \ \forall \xi_1, \xi_2 \in \Xi$$

*(ii) The sample gradients satisfy with probability 1 that*

$$\mathbb{E}\left[\|\nabla g(\xi)\|^4 \mid s_0, a_0\right] \leq C_g^2, \ \forall \xi \in \Xi\,, \qquad \mathbb{E}\left[\|\nabla f(y)\|^4\right] \leq C_f^2, \ \forall y \in \mathbb{R}^d$$

With this additional smoothness assumption, sample complexity is reduced, as we state in the following corollary.

**Corollary 2.** *Consider the actor parameter sequence defined by Algorithm 3. If the stepsize $\eta_k = k^{-1/2}$ and the critic stepsizes are $\alpha_t = 1/t\sigma$ and $\beta_t = 1/t^{4/5}$, then we have the following bound on $K_\epsilon$ defined in* (36)*:*

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-5/2}\right). \tag{53}$$

*Proof.* The proof is identical to the proof of Corollary 1 while invoking Theorem 7 from Wang et al. (2017a). □

Corollary 2 establishes a $\mathcal{O}(\epsilon^{-5/2})$ sample complexity of actor-critic when accelerated GTD steps are used for the critic update. This is the lowest complexity/fastest rate relative to all others analyzed in this work for continuous spaces. However, this fast rate requires the most stringent smoothness conditions. In the following section, we shift to the case where the critic is updated using vanilla TD(0) updates (30), which is the original form of actor-critic proposed by Konda and Borkar (1999).

# 6 Sample Complexity of Classic Actor-Critic

In this section, we derive convergence rates for actor-critic when the critic is updated using TD(0) as in (30) for two different canonical settings: the case where the state space action is continuous (Sec. 6.1) and when it is finite (Sec. 6.2). Both use TD(0) with linear function approximation in its unaltered form (Sutton, 1988). We substitute Algorithm 4 for the **Critic**$(k)$ step in Algorithm 1, which is the classical form of actor-critic given by Konda and Borkar (1999); Konda and Tsitsiklis (2000), thus the name Classic Actor-Critic.

**Algorithm 4** Critic: Classical Temporal Difference (TD(0))

**Require:**

$s_0 \in \mathbb{R}^n$, $\theta$, $\xi_0$, stepsizes $\{\alpha_t\}$, Horizon $T_C$

Initialize $z_0 \leftarrow 0$

**for** $t = 0, \ldots, T_C - 1$ **do**

  Sample $s_t$ from the ergodic distribution and draw action $a_t \sim \pi^{\theta_k}$

  Observe next state $s'_t \sim \mathbf{P}(s_t, a_t, s'_t)$ and observe reward $r_t$

  $\delta_t = r_t + \xi_t^\top \left(\gamma \varphi(s'_t, a'_t) - \varphi(s_t, a_t)\right)$

  Update Critic (Q function estimate):

$$\xi_{t+1} = \xi_t + \alpha_t \delta_t \varphi(s_t, a_t)$$

## 6.1 Continuous State and Action Spaces

The analysis for Continuous State Action space TD(0) with linear function approximation uses the analysis from Dalal et al. (2018a) to characterize the rate of convergence for the critic. Their analysis requires the following common assumption.

**Assumption 5.** *There exists a constant $K_s > 0$ such that for the filtration $\mathcal{G}_t$ defined for the TD(0) critic updates, we have*

$$\mathbb{E}[\|M_{t+1}\|^2 | \mathcal{G}_t] \leq K_s[1 + \|\xi_t - \xi_*\|^2], \tag{54}$$

*where $M_{t+1}$ is defined as*

$$M_{t+1} = \left(r_t + \gamma \xi_t^\top \varphi(s_{t+1}, a_{t+1}) - \xi_t^\top \varphi(s_t, a_t)\right) \varphi(s_t, a_t) - b + A \tag{55}$$

*where*

$$b := \mathbb{E}_{s,a\sim\pi(s)}[r(s,a)\varphi(s,a)], \text{ and } A := \mathbb{E}_{s,a\sim\pi(s)}[\varphi(s,a)(\varphi(s,a) - \gamma\varphi(s',a'))^\top] \tag{56}$$

Assumption 5 is known to hold when the samples have uniformly bounded second moments, which is a common assumption for convergence results (Sutton et al., 2009a,b). In the same way the projected Bellman error is strongly convex [see (48)], it is known that $A$ is positive definite. As such, we define $\lambda_{\text{TD}} \in (0, \lambda_{\min}(A + A^\top))$. The value of $\lambda_{\text{TD}}$ is conditioned on the feature representation of the state space, which is chosen *a priori*. However, this value plays an important role in determining the rate of convergence for TD(0), as we see in the following corollary.

**Corollary 3.** *Consider the actor parameter sequence defined by Algorithm 4. Suppose the actor step-size is chosen as $\eta_k = k^{-1/2}$ and the critic step-size takes the form $\alpha_t = 1/(t+1)^\sigma$ where $\sigma \in (0, 1)$. Then, for large enough $k$,*

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-2/\sigma}\right) \tag{57}$$

*Proof.* Here we invoke the TD(0) convergence result from (Dalal et al., 2018b, Theorem 3.1) which establishes that

$$\mathbb{E}[\|\xi_t - \xi_*\|^2] \leq K_1 e^{-\lambda_{\text{TD}} t^{1-\sigma}/2} + \frac{K_2}{t^\sigma} \tag{58}$$

for some positive constants $K_1$ and $K_2$. For $\sigma$ not close to 1, the first term is dominated by $K_2/t^\sigma$, which permits us to write that

$$\mathbb{E}[\|\xi_t - \xi_*\|^2] \leq \mathcal{O}\left(\frac{1}{t^\sigma}\right) \tag{59}$$

Applying Jensen's inequality, we have

$$\mathbb{E}[\|\xi_t - \xi_*\|]^2 \leq \mathbb{E}[\|\xi_t - \xi_*\|^2] \leq \mathcal{O}\left(\frac{1}{t^\sigma}\right). \tag{60}$$

Taking the square root on both sides gives us

$$\mathbb{E}[\|\xi_t - \xi_*\|] \leq \mathcal{O}\left(\frac{1}{t^{\sigma/2}}\right), \tag{61}$$

which means that the convergence rate statement of Proposition 1 is satisfied with parameter $b = \sigma/2$. Because $\sigma < 1/2$, this specializes Theorem 3, specifically, (38) to case (i), which yields the rate

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-2/\sigma}\right) . \tag{62}$$

Thus the claim in Corollary 3 is valid. □

The operative phrase in the proof of the previous theorem is *for $\sigma$ not close to 1.* This is because we want the first second term of (58) to dominate the first term so that Proposition 1 holds. Asymptotically, this is not a problem, however for finite sample complexity, the point at which the exponential term is dominated by the second term is highly sensitive to both $\lambda_{\text{TD}}$ and $\sigma$. The choice of $\sigma$ can be chosen to be larger as the value of $\lambda_{\text{TD}}$ grows. The choice of $\sigma$ as a function of $\lambda_{\text{TD}}$ and the number of iterates is summarized in Figure 1.

We find that as the value of $\lambda_{\text{TD}}$ increases, the critical value of $\sigma$ also increases. This means that the stepsize of the critic can be chosen to be larger, allowing for faster convergence. Again, we define the critical value of $\sigma$ to be the point at which both terms on the right hand side of (64) are equal at a specific time $t$. Therefore, the feature space representation plays a large role on the performance of actor-critic with TD(0) updates. This result becomes apparent in our numerical results (section 7). We note that, the GTD rates given in Corollary 1 hinge upon strong convexity of the projected Bellman error, which may hold for carefully chosen state-action feature maps, bounded parameter spaces, and lower bounds on the reward. These conditions are absent for TD(0) critic updates.

In the next section, we will consider analysis of actor-critic with TD(0) critic updates in the case where the state and action spaces are finite. As would be expected, this added assumption significantly improves the bound on the rate of convergence, i.e., reduces the sample complexity needed for policy parameters that are within $\epsilon$ of stationary points of the value function.

## 6.2 Finite State and Action Spaces

In this section, we characterize the rate of convergence for the actor-critic defined by Algorithm 1 with TD(0) critic updates (Algorithm 4) when the number of states and actions are finite, i.e., $|\mathcal{S}| = S < \infty$ and $|\mathcal{A}| = A < \infty$. This setting yields faster convergence. A key quantity in the analysis of TD(0) in finite spaces is the minimal eigenvalue of the covariance of the feature map $\phi(s, a)$ weighted by policy $\pi(s)$, which is defined as

$$\omega = \min\left\{ \text{eig}\left( \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} \pi(s)\varphi(s,a)\varphi(s,a)^\top \right) \right\} . \tag{63}$$

That $\omega$ exists is an artifact from the finite state action space assumption. (63) is used to define conditions on the rate of step-size attenuation for TD(0) [cf. (30)] critic updates in (Bhandari et al., 2018, Theorem 2 (c)), which we invoke to establish the iteration complexity of actor-critic in finite spaces. We do so next.

**Corollary 4.** *Consider the actor parameter sequence defined by Algorithm 4. Let the actor step-size satisfy $\eta_k = k^{-1/2}$ and the critic step-size decrease as $\alpha_t = \beta/(\lambda + t)$ where $\beta = 2/\omega(1-\gamma)$ and $\lambda = 16/\omega(1-\gamma)^2$. Then when the number of critic updates per actor update satisfies $T_C(k) = k + 1$, the following convergence rate holds*

$$K_\epsilon \leq O\left(\epsilon^{-2}\right) \tag{64}$$

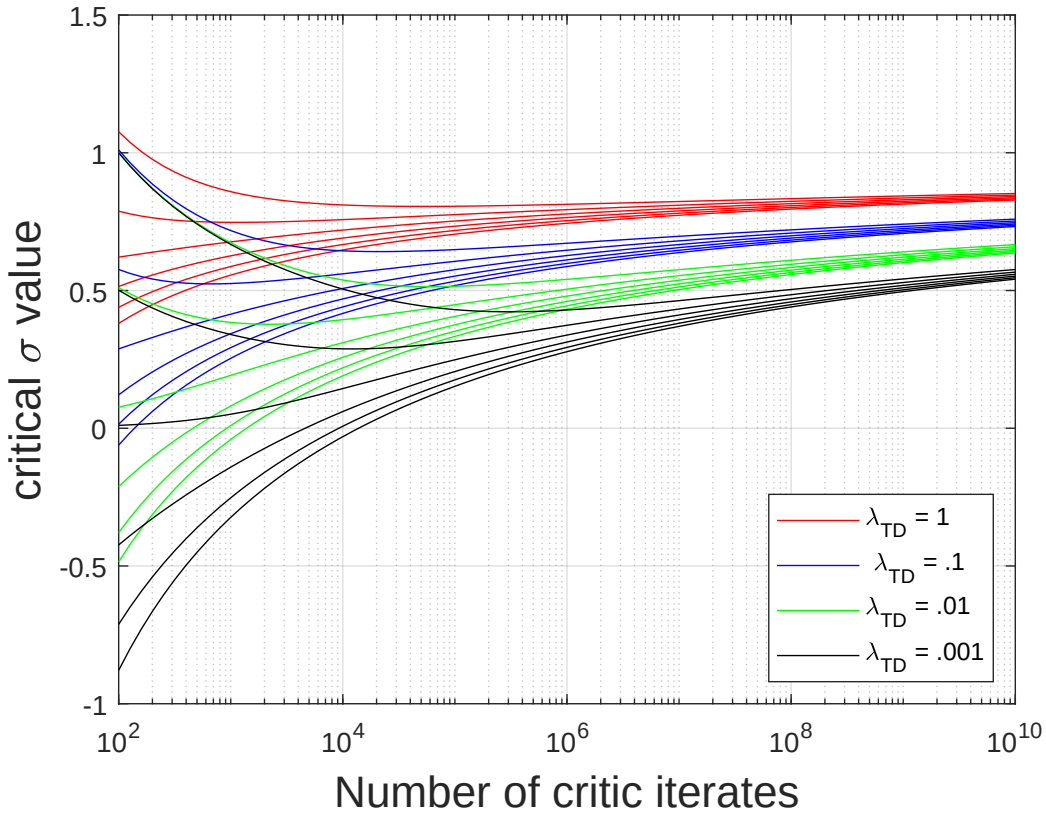



Figure 1: Plot shows the critical value of $\sigma$ for which the exponential term of (64) is dominated by the second term, thereby allowing Proposition 1 to hold. *In particular, any $\sigma > 0$ chosen between zero and the curves shown above satisfies the proposition.* We show plots for varying values of $\lambda_{\mathrm{TD}}$, which is determined by the feature space representation. For each value of $\lambda_{\mathrm{TD}}$, we vary the ratio of the constants $K_2/K_1$ from .001 to 100.

*Proof.* We begin by invoking the TD(0) convergence result (Bhandari et al., 2018, Theorem 2 (c)):

$$\mathbb{E}[\|\xi_t - \xi_*\|^2] \leq \mathcal{O}\left(\frac{K_1}{t + K_2}\right), \tag{65}$$

for some constants $K_1, K_2$ which depend on $\omega$ and $\sigma$. Applying Jensen's inequality, we have

$$\mathbb{E}[\|\xi_t - \xi_*\|]^2 \leq \mathbb{E}[\|\xi_t - \xi_*\|^2] \leq \mathcal{O}\left(\frac{K_1}{t + K_2}\right). \tag{66}$$

Taking the square root on both sides yields

$$\mathbb{E}[\|\xi_t - \xi_*\|] \leq \mathcal{O}\left(\frac{K_1^{-1/2}}{(t + K_2)^{-1/2}}\right) \lesssim \mathcal{O}(t^{-1/2}), \tag{67}$$

which means that Proposition 1 is valid with critic convergence rate parameter $b = 1/2$. Therefore, we may apply Theorem 3 to obtain the rate

$$K_\epsilon \leq \mathcal{O}\left(\epsilon^{-2}\right) \tag{68}$$

as stated in Corollary 4. □

# 7 Numerical Results

In this section, we compare the convergence rates of actor-critic with the aforementioned critic-only methods on a two-dimensional navigation problem and the inverted pendulum. Before detailing the RL problem specifics, we first discuss the metrics we use to evaluate both performance and convergence.

Because the main objective is to maximize the long term average reward accumulation, it follows naturally to measure the cumulative reward of a trajectory. We evaluate the policy without action noise ($\sigma^2 = 0$), with a fixed trajectory length, and with a fixed starting position which makes the plots easier to compare. In addition, we consider a proxy for the gradient norm. In particular, we calculate the norm of the difference between two consecutive normalized actor parameters ($\|\theta_k/\|\theta_k\| - \theta_{k+1}/\|\theta_{k+1}\|\|$). The normalization treats two scaled versions of the same parameter equivalently. This is meaningful because the action vector field induced by the parameters (see Fig 3) are similarly scaled versions of each other. In this form, the gradient norm proxy serves as the optimization metric on which our main result is based.

Along with varying the critic-only methods, we elect to consider two additional variations on policy gradient where the $Q$ function is replaced by the *advantage* and *value* functions. Recall the definition of the value function from (1). The advantage function is defined by $A(s_t, a_t) = Q(s_t, a_t) - V(s_t)$, which, by definition of the $Q$ function, can also take the form of $A(s_t, a_t) = r_{t+1} + \gamma V(s_{t+1}) - V(s_t)$ (Mnih et al., 2016). The main benefit of using the value function and advantage functions instead of the $Q$ function for actor critic is that the dimension of the function approximator domain is smaller, as the agent only needs to learn on the state space.

## 7.1 Navigating around an obstacle

We consider the problem of a point agent starting at an initial state $s_0 \in \mathbb{R}^2$ whose objective is to navigate to a destination $s^* \in \mathbb{R}^2$ while remaining in the free space at all time. The free space $\mathcal{X} \subset \mathbb{R}^2$ is defined by

$$\mathcal{X} := \left\{ s \in \mathbb{R}^2 \middle| \|s\| \in [0.5, 4] \right\}. \tag{69}$$

The feature representation of the state is determined by a radial basis (Gaussian) kernel where

$$\kappa(s, s') = \exp\left\{ \frac{-\|s - s'\|_2^2}{2\sigma^2} \right\}. \tag{70}$$

The $p$ kernel points are chosen evenly on the $[-5, 5] \times [-5, 5]$ grid so that the the feature representation becomes

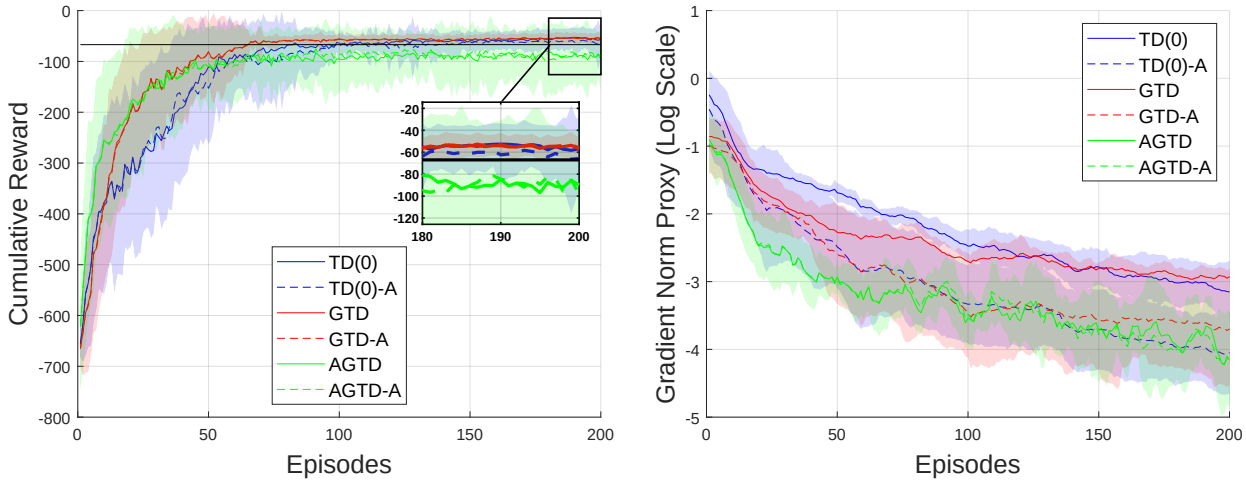
$$\varphi(s) = \begin{bmatrix} \kappa(s, s_1) & \kappa(s, s_2) & \dots & \kappa(s, s_p) \end{bmatrix}^\top, \tag{71}$$

which we normalize. Given the state $s_t$, the action is sampled from a multivariate Gaussian distribution with covariance matrix $\Sigma = 0.5 \cdot I_2$ and mean given by $\theta_k^\top \varphi(s_t)$. We let the action determine the direction in which the agent will move. As such, the state transition is determined by $s_{t+1} = s_t + 0.5 a_t / \|a_t\|$.

Because the agent's objective is to reach the target $s^*$ while remaining in $F$ for all time, we want to penalize the agent heavily for taking actions which result in the next step being outside the free space and reward the agent for being close to the target. As such, we define the reward function to be

$$r_{t+1} = \begin{cases} -11 & \text{if } s_{t+1} \notin \mathcal{X} \\ -0.1 & \text{if } \|s_{t+1} - s^*\| < 0.5 \\ -1 & \text{otherwise .} \end{cases} \tag{72}$$

The design of this reward function for the navigation problem is informed by the (Zhang et al., 2020), which suggests that the reward function should be bounded away from zero. In this simulation, we allow for the




(a) (b)

Figure 2: Navigation Problem: (a) Average reward per episode with confidence bounds over 50 trials. (b) Average gradient norm proxy over 50 trials. A-GTD converges fastest with respect to the cumulative reward *and* gradient norm proxy at the cost of converging to a suboptimal stationary point (see Fig. 3). A moving average filter of size ten has been applied on the gradient norm proxy to aid in comparison.

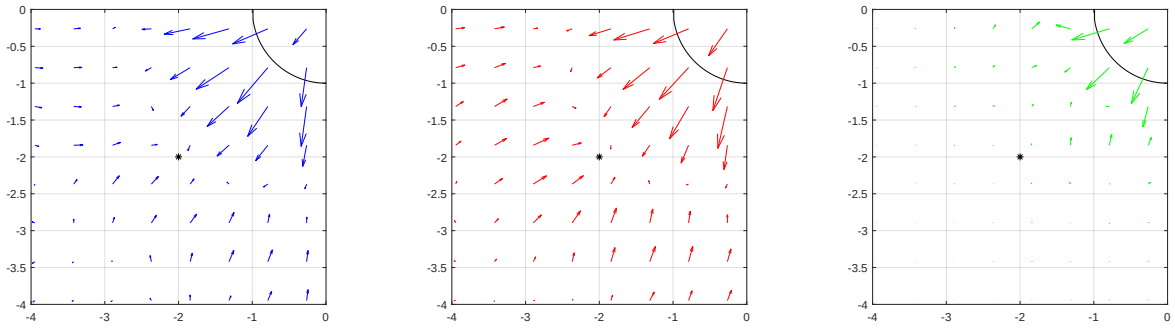
agent to continue taking actions *through* the obstacles. This formulation is similar to a car driving on a race track which has grass outside the track. The car is allow is allowed to drive off the track, however it incurs a larger cost due to the substandard driving conditions.

Although it is true that this particular formulation does not allow for generalization, that is, if the target of the agent, obstacle location, or starting point of the agent are moved, the agent would have to start from scratch to learn a new meaningful policy, we emphasize that it is the rates of convergence which are of interest in this exposition, not necessarily finding the best way to design the navigation problem.

**Algorithm Specifics:** We consider the problem with $\gamma = 0.97$. In practice, we use the entire trajectory data for the critic updates. In particular, for each actor parameter update, we run ten critic updates with rollout length $T = 66$ (comes from the expected rollout length given $\gamma = 0.97$). Similarly, we update the actor along the trajectory of rollout length H = 67. For simulations, the actor update step $\eta_t$ is chosen to be constant $\eta = 10^{-3}$. For TD(0), we let also let the critic stepsize be constant, namely $\alpha_t = \alpha = 0.05$. For GTD, we let $\alpha_t = t^{-1}$ and $\beta_t = t^{-2/3}$. For A-GTD, we set $\alpha_t = t^{-1}$ and $\beta_t = t^{-4/5}$. We draw the initial distribution uniformly at random on the grid $[-2, 2] \times [-2, 2]$, and we set the target to be $s^* = (-2, -2)$. For each critic only method, we run the algorithm 50 times. We evaluate the policy by measuring the accumulated reward of a trajectory of length $H = 66$.

## 7.2 Pendulum Problem

We also consider the canonical continuous state action space reinforcement learning problem of the pendulum. The objective is to balance the pendulum upright starting from any starting position. Given that this is a well established benchmark for reinforcement learning, we refer the reader to Brockman et al. (2016) for the specifications on reward and transition dynamics. Similar to the navigation problem, we let the feature representation of the state be determined by a radial basis (Gaussian) kernel (c.f. (70)) where the $p$ kernel points are chosen evenly on $[-1, -1, -8, -2] \times [1, 1, 8, 2]$, where the bounds come from the sine and cosine of the angle $\theta$, the time derivative of the angle $\dot{\theta}$, and the maximum torque of the action respectively. The action is chosen by a normal distribution with mean $\xi^\top \varphi(s, a)$ and variance $\sigma_a^2$. Like the navigation problem,



(a) (b) (c)

Figure 3: Visualization of the learned policy for the navigation problem. The obstacle is shown in the top right corner, and the target is located at (-2,-2). As Figure 2 (a) depicts, TD (shown in (a)) and GTD (shown in (b)) learn meaningful policies which guide the agent to the target. In contrast, A-GTD (shown in (c)) simply learns to avoid the obstacle.

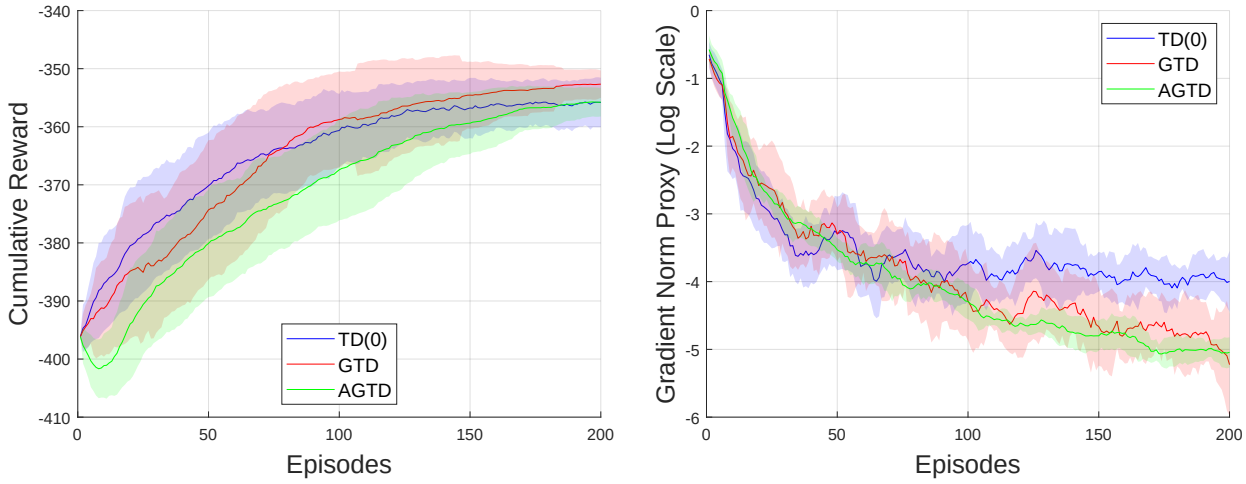
we use a linear policy and linear critic. Again, we stress that these experiments are meant to show the rates of convergence, and not necessarily finding the best way to solve the pendulum problem. For the pendulum problem, we only consider advantage actor-critic.

**Algorithm Specifics:** Similar to the navigation problem, we let $\gamma = 0.97$, and we use the entire trajectory data for the critic updates. In particular, for each actor parameter update, we run ten critic updates with rollout length $T = 66$ (comes from the expected rollout length given $\gamma = 0.97$). Similarly, we update the actor along the trajectory of rollout length H = 66. For simulations, the actor update step $\eta_t$ is chosen to be constant $\eta = 0.01$. For critic only methods, we also let also let the critic stepsize be constant. In particular, we let $\alpha_t = 0.01$ for TD(0), $(\alpha_t, \beta_t) = (0.2, 0.01)$ for GTD, and $(\alpha_t, \beta_t) = (0.05, 0.005)$ for AGTD. We evaluate the policy by measuring the average accumulated by a single trajectory starting $\theta = \pi/2$ with angular velocity $\omega = 1$. The action variance is chosen to be $\sigma_a^2 = 0.5$.

## 7.3 Discussion

Recall that the analysis of Corollaries 1, 2, and 3 establish that the convergence rates for GTD, A-GTD, and TD(0) are $O(\epsilon^{-3})$, $O(\epsilon^{-5/2})$, and $O(\epsilon^{-2/\sigma})$ respectively [also see Table 1]. Figure 2 shows the performance of the navigation problem with value and advantage function policy gradient updates. As expected, A-GTD converges fastest with respect to the gradient norm proxy, while GTD and TD(0) are comparable. The plots highlight a disconnect between the convergence in reward and the convergence in gradient norm. Namely, TD converges faster in gradient norm, but slower with respect to the cumulative reward. Even more interesting, although AGTD converges fastest with respect to gradient norm *and* reward, its resulting stationary point is suboptimal compared to TD and GTD (see Fig 3). On the other hand, GTD and TD(0) converge the slower, and they consistently reach the *solved* region marked by the solid black line at $-66$. We say that rewards which are greater than $-66$ are *solved* trajectories because these trajectory spend time in the destination region. A trajectory which does not reach the destination region will have accumulated reward of $-66$ or less. Taken together, these theoretical and experimental results suggest a tight coupling between the choice of training methodology and the quality of learned policies. Thus, just as the choice of optimization method, statistical model, and sample size influence generalization in supervised learning, they do so in reinforcement learning. Theorem 3 characterizes the rate of convergence to a stationary point of the Bellman optimality operator, however it does not provide any guarantee on the quality of the stationary point. Figure 3 captures this trade-off convergence rate and quality of the stationary point.

The disconnect between convergence in reward and convergence in gradient norm appears again in the




(a) (b)

Figure 4: Pendulum Problem: (a) Average reward per episode with confidence bounds over 50 trials. (b) Average gradient norm proxy over 50 trials. In contrast to the navigation problem there is a significant gain in using advantage actor-critic; here, the state action ($Q$) function was used instead of the value function ($V$). A moving average filter of size ten has been applied on the gradient norm proxy to aid in comparison.

pendulum. Figure 4 (b) shows the gradient norm proxy for the advantage actor-critic applied to the pendulum problem. Consistent with Table 1, AGTD converges the fastest with followed by GTD and TD(0). Here, we again see the disconnect between convergence in gradient norm and cumulative reward. Notice how in the first few iterations, TD(0) actually converges the fastest. In tandem, the cumulative reward of TD(0) also increases quickly. By the final episode, TD(0) and AGTD perform worse than GTD. This is consistent with the convergence rate and quality of stationary point trade-off observed in the navigation problem.

There are a number of future directions to take this work. To begin, we can establish bounds on cases where the samples are not i.i.d., but instead have Markovian noise. Second, we can further generalize our results to consider a generic critic convergence rate that does not necessarily take the form of Proposition 1. Third, we can explore the choice of feature representation to explicitly characterize the convergence rate of actor-critic with TD(0) critic updates with respect to $\lambda_{\text{TD}}$. Finally, we can characterize the behavior of the variance and use such characterizations to accelerate training.

# References

Antos A, Szepesvári C, Munos R (2008) Fitted q-iteration in continuous action-space mdps. In: Advances in neural information processing systems, pp 9–16

Baird L (1995) Residual algorithms: Reinforcement learning with function approximation. In: Machine Learning Proceedings 1995, Elsevier, pp 30–37

Bellman R (1954) The theory of dynamic programming. Tech. rep., RAND Corp Santa Monica CA

Bellman RE (1957) Dynamic Programming. Courier Dover Publications

Bertsekas DP (2005) Dynamic Programming and Optimal Control, vol 1

Bertsekas DP, Bertsekas DP, Bertsekas DP, Bertsekas DP (1995) Dynamic programming and optimal control, vol 1. Athena scientific Belmont, MA

Bhandari J, Russo D, Singal R (2018) A finite time analysis of temporal difference learning with linear function approximation. In: Conference on learning theory, PMLR, pp 1691–1692

Bhatnagar S, Ghavamzadeh M, Lee M, Sutton RS (2008) Incremental natural actor-critic algorithms. In: Advances in Neural Information Processing Systems, pp 105–112

Bhatnagar S, Sutton R, Ghavamzadeh M, Lee M (2009) Natural actor-critic algorithms. Automatica 45(11):2471–2482

Borkar VS (1997) Stochastic approximation with two time scales. Systems & Control Letters 29(5):291–294

Borkar VS (2009) Stochastic approximation: a dynamical systems viewpoint, vol 48. Springer

Borkar VS, Meyn SP (2000) The ode method for convergence of stochastic approximation and reinforcement learning. SIAM Journal on Control and Optimization 38(2):447–469

Bottou L (1998) Online learning and stochastic approximations. On-line learning in neural networks 17(9):142

Bousquet O, Elisseeff A (2002) Stability and generalization. Journal of machine learning research 2(Mar):499–526

Boyan JA, Moore AW (1995) Generalization in reinforcement learning: Safely approximating the value function. In: Advances in neural information processing systems, pp 369–376

Brockman G, Cheung V, Pettersson L, Schneider J, Schulman J, Tang J, Zaremba W (2016) Openai gym. arXiv preprint arXiv:160601540

Cai Q, Yang Z, Lee JD, Wang Z (2019) Neural temporal-difference learning converges to global optima. Advances in Neural Information Processing Systems 32

Castro DD, Meir R (2010) A convergent online single-time-scale actor-critic algorithm. Journal of Machine Learning Research 11(Jan):367–410

Cayci S, He N, Srikant R (2022) Finite-time analysis of entropy-regularized neural natural actor-critic algorithm. arXiv preprint arXiv:220600833

Chen Z, Khodadadian S, Maguluri ST (2022a) Finite-sample analysis of off-policy natural actor–critic with linear function approximation. IEEE Control Systems Letters 6:2611–2616

Chen Z, Zhou Y, Chen RR, Zou S (2022b) Sample and communication-efficient decentralized actor-critic algorithms with finite-time analysis. In: International Conference on Machine Learning, PMLR, pp 3794–3834

Cybenko G (1989) Approximation by superpositions of a sigmoidal function. Mathematics of control, signals and systems 2(4):303–314

Dalal G, Szörényi B, Thoppe G, Mannor S (2018a) Finite sample analyses for td (0) with function approximation. In: Thirty-Second AAAI Conference on Artificial Intelligence

Dalal G, Thoppe G, Szörényi B, Mannor S (2018b) Finite sample analysis of two-timescale stochastic approximation with applications to reinforcement learning. In: Conference On Learning Theory, pp 1199–1233

Dalal G, Szörényi B, Thoppe G (2020) A tale of two-timescale reinforcement learning with the tightest finite-time bound. AAAI Press, pp 3701–3708, URL `https://aaai.org/ojs/index.php/AAAI/article/view/5779`

Doya K (2000) Reinforcement learning in continuous time and space. Neural Computation 12(1):219–245

Ge R, Huang F, Jin C, Yuan Y (2015) Escaping from saddle points–online stochastic gradient for tensor decomposition. In: Conference on Learning Theory, pp 797–842

Giannoccaro I, Pontrandolfo P (2002) Inventory management in supply chains: a reinforcement learning approach. International Journal of Production Economics 78(2):153–161

Gruslys A, Dabney W, Azar MG, Piot B, Bellemare M, Munos R (2018) The reactor: A fast and sample-efficient actor-critic agent for reinforcement learning. In: International Conference on Learning Representations

Jiang DR, Pham TV, Powell WB, Salas DF, Scott WR (2014) A comparison of approximate dynamic programming techniques on benchmark energy storage problems: Does anything work? In: 2014 IEEE Symposium on Adaptive Dynamic Programming and Reinforcement Learning (ADPRL), IEEE, pp 1–8

Jin C, Allen-Zhu Z, Bubeck S, Jordan MI (2018) Is q-learning provably efficient? In: Advances in Neural Information Processing Systems 31, pp 4863–4873

Kober J, Peters J (2012) Reinforcement learning in robotics: A survey. In: Reinforcement Learning, Springer, pp 579–610

Konda VR, Borkar VS (1999) Actor-critic–type learning algorithms for Markov decision processes. SIAM Journal on Control and Optimization 38(1):94–123

Konda VR, Tsitsiklis JN (2000) Actor-critic algorithms. In: Advances in Neural Information Processing Systems, pp 1008–1014

Koppel A, Warnell G, Stump E, Stone P, Ribeiro A (2017) Breaking bellman's curse of dimensionality: Efficient kernel gradient temporal difference. arXiv preprint arXiv:170904221

Kushner HJ, Yin GG (2003) Stochastic approximation and recursive algorithms and applications. Springer, New York, NY

Lakshminarayanan C, Szepesvari C (2018) Linear stochastic approximation: How far does constant step-size and iterate averaging go? In: International Conference on Artificial Intelligence and Statistics, pp 1347–1355

Li X, Wang Z, Lu J, Arora R, Haupt J, Liu H, Zhao T (2016) Symmetry, saddle points, and global geometry of nonconvex matrix factorization. arXiv preprint arXiv:161209296 1:5–1

Liu B, Liu J, Ghavamzadeh M, Mahadevan S, Petrik M (2015) Finite-sample analysis of proximal gradient td algorithms. In: Proceedings of the Thirty-First Conference on Uncertainty in Artificial Intelligence, pp 504–513

Maei HR, Szepesvári C, Bhatnagar S, Sutton RS (2010) Toward off-policy learning control with function approximation. In: Proceedings of the 27th International Conference on Machine Learning (ICML-10), pp 719–726

Meyn SP, Tweedie RL (2012) Markov chains and stochastic stability. Springer Science & Business Media

Mnih V, Badia AP, Mirza M, Graves A, Lillicrap T, Harley T, Silver D, Kavukcuoglu K (2016) Asynchronous methods for deep reinforcement learning. In: International Conference on Machine Learning, pp 1928–1937

Nesterov YE (1983) A method for solving the convex programming problem with convergence rate o (1/kˆ 2). In: Dokl. akad. nauk Sssr, vol 269, pp 543–547

Olshevsky A, Gharesifard B (2022) A small gain analysis of single timescale actor critic. arXiv preprint arXiv:220302591

Papini M, Binaghi D, Canonaco G, Pirotta M, Restelli M (2018) Stochastic variance-reduced policy gradient. In: International Conference on Machine Learning, pp 4026–4035

Parisi S, Tangkaratt V, Peters J, Khan ME (2019) Td-regularized actor-critic methods. Machine Learning 108(8-9):1467–1501

Park J, Sandberg IW (1991) Universal approximation using radial-basis-function networks. Neural computation 3(2):246–257

Paternain S (2018) Stochastic control foundations of autonomous behavior. PhD thesis, University of Pennsylvania

Pirotta M, Restelli M, Bascetta L (2015) Policy gradient in Lipschitz Markov Decision processes. Machine Learning 100(2-3):255–283

Powell WB (2007) Approximate Dynamic Programming: Solving the curses of dimensionality, vol 703. John Wiley & Sons

Puterman ML (2014) Markov decision processes: discrete stochastic dynamic programming. John Wiley & Sons

Qiu S, Yang Z, Ye J, Wang Z (2021) On finite-time convergence of actor-critic algorithm. IEEE Journal on Selected Areas in Information Theory 2(2):652–664

Shen H, Zhang K, Hong M, Chen T (2020) Asynchronous advantage actor critic: Non-asymptotic analysis and linear speedup. arXiv preprint arXiv:201215511

Silver D, Schrittwieser J, Simonyan K, Antonoglou I, Huang A, Guez A, Hubert T, Baker L, Lai M, Bolton A, et al. (2017) Mastering the game of Go without human knowledge. Nature 550(7676):354–359

Srikant R, Ying L (2019) Finite-time error bounds for linear stochastic approximation andtd learning. In: Conference on Learning Theory, PMLR, pp 2803–2830

Sun J, Qu Q, Wright J (2016) A geometric analysis of phase retrieval. In: Information Theory (ISIT), 2016 IEEE International Symposium on, IEEE, pp 2379–2383

Sutton RS (1988) Learning to predict by the methods of temporal differences. Machine learning 3(1):9–44

Sutton RS, McAllester DA, Singh SP, Mansour Y (2000) Policy gradient methods for reinforcement learning with function approximation. In: Advances in Neural Information Processing Systems, pp 1057–1063

Sutton RS, Szepesvári C, Maei HR (2008) A convergent o (n) algorithm for off-policy temporal-difference learning with linear function approximation. Advances in neural information processing systems 21(21):1609–1616

Sutton RS, Maei HR, Precup D, Bhatnagar S, Silver D, Szepesvári C, Wiewiora E (2009a) Fast gradient-descent methods for temporal-difference learning with linear function approximation. In: International Conference on Machine Learning, ACM, pp 993–1000

Sutton RS, Maei HR, Szepesvári C (2009b) A convergent $o(n)$ temporal-difference algorithm for off-policy learning with linear function approximation. In: Advances in neural information processing systems, pp 1609–1616

Sutton RS, Barto AG, et al. (2017) Reinforcement Learning: An Introduction, 2nd edn

Tesauro G, et al. (1995) Temporal difference learning and td-gammon. Communications of the ACM 38(3):58–68

Thoppe G, Borkar V (2019) A concentration bound for stochastic approximation via alekseev's formula. Stochastic Systems 9(1):1–26, DOI 10.1287/stsy.2018.0019

Tolstaya E, Koppel A, Stump E, Ribeiro A (2018) Nonparametric stochastic compositional gradient descent for q-learning in continuous markov decision problems. In: 2018 Annual American Control Conference (ACC), IEEE, pp 6608–6615

Tsitsiklis JN (1994) Asynchronous stochastic approximation and q-learning. Machine learning 16(3):185–202

Tsitsiklis JN, Van Roy B (1997) Analysis of temporal-diffference learning with function approximation. In: Advances in Neural Information Processing Systems, pp 1075–1081

Wang L, Cai Q, Yang Z, Wang Z (2019) Neural policy gradient methods: Global optimality and rates of convergence. In: International Conference on Learning Representations

Wang M, Fang EX, Liu H (2017a) Stochastic compositional gradient descent: algorithms for minimizing compositions of expected-value functions. Mathematical Programming 161(1-2):419–449

Wang Z, Bapst V, Heess N, Mnih V, Munos R, Kavukcuoglu K, de Freitas N (2017b) Sample efficient actor-critic with experience replay. In: International Conference on Learning Representations

Watkins CJ, Dayan P (1992) Q-learning. Machine learning 8(3-4):279–292

Wu YF, Zhang W, Xu P, Gu Q (2020) A finite-time analysis of two time-scale actor-critic methods. Advances in Neural Information Processing Systems 33:17617–17628

Xu T, Wang Z, Liang Y (2020) Non-asymptotic convergence analysis of two time-scale (natural) actor-critic algorithms. arXiv preprint arXiv:200503557

Yang Z, Zhang K, Hong M, Başar T (2018) A finite sample analysis of the actor-critic algorithm. In: 2018 IEEE Conference on Decision and Control (CDC), IEEE, pp 2759–2764

Zeng S, Chen T, Garcia A, Hong M (2022) Learning to coordinate in multi-agent systems: A coordinated actor-critic algorithm and finite-time guarantees. In: Learning for Dynamics and Control Conference, PMLR, pp 278–290

Zhang K, Yang Z, Liu H, Zhang T, Başar T (2018) Fully decentralized multi-agent reinforcement learning with networked agents. In: International Conference on Machine Learning, pp 5872–5881

Zhang K, Koppel A, Zhu H, Başar T (2019) Convergence and iteration complexity of policy gradient method for infinite-horizon reinforcement learning. IEEE Conference on Decision and Control

Zhang K, Koppel A, Zhu H, Basar T (2020) Global convergence of policy gradient methods to (almost) locally optimal policies. SIAM Journal on Control and Optimization 58(6):3586–3612

Zou S, Xu T, Liang Y (2019) Finite-sample analysis for sarsa with linear function approximation. Advances in neural information processing systems 32

# 8 Appendix

## 8.1 Proof of Lemma 2

By the Mean Value Theorem, there exists $\tilde{\theta}_k = \lambda\theta_k + (1-\lambda)\theta_{k+1}$ for some $\lambda \in [0,1]$ such that

$$J(\theta_{k+1}) = J(\theta_k) + (\theta_{k+1} - \theta_k)^\top \nabla J(\tilde{\theta}_k). \tag{73}$$

Add and subtract $(\theta_{k+1}-\theta_k)^\top \nabla J(\theta_k)$ to the right hand side of (73) to obtain

$$J(\theta_{k+1}) = J(\theta_k) + (\theta_{k+1}-\theta_k)^\top \left(\nabla J(\tilde{\theta}_k) - \nabla J(\theta_k)\right) + (\theta_{k+1}-\theta_k)^\top \nabla J(\theta_k). \tag{74}$$

By Cauchy Schwartz, we know $(\theta_{k+1}-\theta_k)^\top \left(\nabla J(\tilde{\theta}_k) - \nabla J(\theta_k)\right) \geq -\|\theta_{k+1}-\theta_k\|\|\nabla J(\tilde{\theta}_k) - J(\theta_k)\|$. Further, by the Lipschitz continuity of the gradient, we know $\|\nabla J(\tilde{\theta}_k) - \nabla J(\theta_k)\| \leq L\|\tilde{\theta}_k - \theta_k\|$. Therefore, we have

$$(\theta_{k+1}-\theta_k)^\top \left(\nabla J(\tilde{\theta}_k) - J(\theta_k)\right) \geq -L\|\tilde{\theta}_k - \theta_k\| \cdot \|\theta_{k+1}-\theta_k\| \geq -L\|\theta_{k+1}-\theta_k\|^2 , \tag{75}$$

where the second inequality comes from substituting $\tilde{\theta}_k = (1-\lambda)\theta_{k+1} + \lambda\theta_k$. We substitute this expression into the definition of $J(\theta_{k+1})$ in (74) to obtain

$$J(\theta_{k+1}) \geq J(\theta_k) + (\theta_{k+1}-\theta_k)^\top \nabla J(\theta_k) - L\|\theta_{k+1}-\theta_k\|^2. \tag{76}$$

Take the expectation with respect to the filtration $\mathcal{F}_k$, and substitute the definition for the actor update (17)

$$\mathbb{E}[J(\theta_{k+1})|\mathcal{F}_k] \geq J(\theta_k) + \mathbb{E}[\theta_{k+1}-\theta_k|\mathcal{F}_k]^\top \nabla J(\theta_k) + -L\mathbb{E}[\|\eta_k \hat{g}^{AC}_{H(k)}\|^2|\mathcal{F}_k]. \tag{77}$$

We know from (29) that $\|\hat{\nabla} J(\theta_k)\|^2 \leq \sigma^2$, as such we obtain

$$\mathbb{E}[J(\theta_{k+1})|\mathcal{F}_k] \geq J(\theta_k) + \mathbb{E}[\theta_{k+1}-\theta_k|\mathcal{F}_k]^\top \nabla J(\theta_k) - L\sigma^2\eta_k^2. \tag{78}$$

Therefore, we are left to show that the last term on the right-hand side of the preceding expression is "nearly" an ascent direction. Recall from Algorithm 1 that the $k^{\text{th}}$ update takes the form (22), that is

$$\mathbb{E}\left[\theta_{k+1}-\theta_k|\mathcal{F}_k\right] = \eta_k \mathbb{E}\left[\hat{g}^{AC}_{H(k)}|F_k\right] = \eta_k \mathbb{E}\left[\nabla_\theta J(\theta_k)|\mathcal{F}_k\right] + \eta_k \mathbb{E}\left[\hat{g}^{AC}_{H(k)} - \nabla_\theta J(\theta)|\mathcal{F}_k\right] \tag{79}$$

Substituting into (78), from Theorem 2, we obtain

$$\begin{aligned}
\mathbb{E}[J(\theta_{k+1})|\mathcal{F}_k] &\geq J(\theta_k) + \eta_k\|\nabla_\theta J(\theta_k)\|^2 + \eta_k \mathbb{E}\left[\hat{g}^{AC}_{H(k)} - \nabla_\theta J(\theta_k)|\mathcal{F}_k\right]^\top \nabla_\theta J(\theta_k) - L\sigma^2\eta_k^2 \\
&\geq J(\theta_k) + \eta_k\|\nabla_\theta J(\theta_k)\|^2 - \eta_k \left|\mathbb{E}\left[\hat{g}^{AC}_{H(k)} - \nabla_\theta J(\theta_k)|\mathcal{F}_k\right]^\top \nabla_\theta J(\theta_k)\right| - L\sigma^2\eta_k^2 \\
&\geq J(\theta_k) + \eta_k\|\nabla_\theta J(\theta_k)\|^2 - \eta_k\|\mathbb{E}\left[\hat{g}^{AC}_{H(k)}\right] - \nabla_\theta J(\theta_k)\| \cdot \|\nabla_\theta J(\theta_k)\| - L\sigma^2\eta_k^2 \\
&\geq J(\theta_k) + \eta_k\|\nabla_\theta J(\theta_k)\|^2 - \eta_k C_\nabla \left(C_1\gamma^{H(k)-1} + C_2 T(k)^{-b}\right) - L\sigma^2\eta_k^2
\end{aligned} \tag{80}$$

This concludes the proof.

## 8.2 Proof of Theorem 3

Take the total expectation of (34) from Lemma 2

$$\mathbb{E}[J(\theta_{k+1})] \geq \mathbb{E}[J(\theta_k)] + \eta_k \mathbb{E}[\|\nabla J(\theta_k)\|^2] - \eta_k C_\nabla C_1 \gamma^{H(k)-1} - \eta_k C_\nabla C_2 T_C(k)^{-b} - L\sigma^2\eta_k^2. \tag{81}$$

Define $U_k := J(\theta^*) - J(\theta_k)$ where $\theta^*$ is the solution of (2) when the policy is parameterized by $\theta$. By this definition, we know that $U_k$ is non-negative for all $\theta_k$. Add $J(\theta^*)$ to both sides of the inequality and rearrange terms

$$\eta_k \mathbb{E}[\|\nabla J(\theta_k)\|] \leq \mathbb{E}[U_k] - \mathbb{E}[U_{k+1}] + L\sigma^2\eta_k^2 + \eta_k C_\nabla C_1 \gamma^{H(k)-1} + \eta_k C_\nabla C_2 T_C(k)^{-b}. \tag{82}$$

Divide both sides by $\eta_k$ and take the sum over $\{k-N,\dots,k\}$ for some integer $1<N<k$

$$\begin{aligned}\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \sum_{j=k-N}^{k}\frac{1}{\eta_j}\left(\mathbb{E}[U_j]-\mathbb{E}[U_{j+1}]\right)+L\sigma^2\sum_{j=k-N}^{k}\eta_j \\ +\sum_{j=k-N}^{k}\left(C_\nabla C_1\gamma^{H(j)-1}C_\nabla C_2 T_C(j)^{-b}\right).\end{aligned} \tag{83}$$

Add and subtract $1/\eta_{k-N-1}\mathbb{E}[U_{k-N}]$ on the right hand side. This allows us to write

$$\begin{aligned}\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \sum_{j=k-N}^{k}\left(\frac{1}{\eta_j}-\frac{1}{\eta_{j-1}}\right)\mathbb{E}[U_j]-\frac{1}{\eta_k}\mathbb{E}[U_{k+1}]+\frac{1}{\eta_{k-N-1}}\mathbb{E}[U_{k-N}] \\ +L\sigma^2\sum_{j=k-N}^{k}\eta_j+\sum_{j=k-N}^{k}\left(C_\nabla C_1\gamma^{H(j)-1}C_\nabla C_2 T_C(j)^{-b}\right).\end{aligned} \tag{84}$$

By definition of $U_k$, $\mathbb{E}[U_{k+1}]\geq 0$. Therefore we can omit it from the right hand side of (84). Further, we know that $J(\theta^*)\leq U_R/(1-\gamma)$ as a consequence from Assumption 1(i) [see (6)]. Hence we have $U_k\leq 2U_R/(1-\gamma)=:C_3$ for all $k$. Substituting this fact into the preceding expression yields

$$\begin{aligned}\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \sum_{j=k-N}^{k}\left(\frac{1}{\eta_j}-\frac{1}{\eta_{j-1}}\right)C_3+\frac{1}{\eta_{k-N-1}}C_3+L\sigma^2\sum_{j=k-N}^{k}\eta_j \\ +\sum_{j=k-N}^{k}\left(C_\nabla C_1\gamma^{H(j)-1}C_\nabla C_2 T_C(j)^{-b}\right).\end{aligned} \tag{85}$$

By unraveling the telescoping sum, the first two terms are equal to $C_3/\eta_k$

$$\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \frac{C_3}{\eta_k}+L\sigma^2\sum_{j=k-N}^{k}\eta_j+\sum_{j=k-N}^{k}\left(C_\nabla C_1\gamma^{H(j)-1}C_\nabla C_2 T_C(j)^{-b}\right). \tag{86}$$

Substitute $\eta_k=k^{-a}$ for the step size

$$\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq C_4 k^a+L\sigma^2\sum_{j=k-N}^{k}j^{-a}+\sum_{j=k-N}^{k}\left(C_\nabla C_1\gamma^{H(j)-1}C_\nabla C_2 T_C(j)^{-b}\right). \tag{87}$$

We break the remainder of the proof into two cases due to the fact that the right-hand side of the preceding expression simplifies when $b=1$, and is more intricate when $0<b<1$. We focus on the later case first.

**Case (i):** $b\in(0,1)$ Consider the case where $b\in(0,1)$. Set $T_C(k)=k$ and $H(k)=k$. Substitute the integration rule, namely that $\sum_{j=k-N}^{k}j^{-a}\leq k^{1-a}-(k-N-1)^{1-a}$, into (87) to obtain:

$$\begin{aligned}\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq & C_4 k^a+C_\nabla C_1\gamma^{-1}\sum_{j=k-N}^{k}\gamma^j+\frac{L\sigma^2}{1-a}\left(k^{1-a}-(k-N-1)^{1-a}\right) \\ & +\frac{CL_1}{1-b}\left(k^{1-b}-(k-N-1)^{1-b}\right).\end{aligned} \tag{88}$$

Divide both sides by $k$ and set $N=k-1$

$$\begin{aligned}\frac{1}{k}\sum_{j=1}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] &\leq C_4 k^{a-1}+C_\nabla C_1\gamma^{-1}k^{-1}\sum_{j=1}^{k}\gamma^j+\frac{L\sigma^2}{1-a}k^{-a}+\frac{CL_1}{1-b}k^{-b} \\ &\leq C_4 k^{a-1}+\frac{C_\nabla C_1}{\gamma(1-\gamma)}k^{-1}+\frac{L\sigma^2}{1-a}k^{-a}+\frac{CL_1}{1-b}k^{-b}\end{aligned} \tag{89}$$

Suppose $k = K_\epsilon$ so that we may write

$$\frac{1}{K_\epsilon}\sum_{j=1}^{K_\epsilon}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \mathcal{O}\left(K_\epsilon^{a-1} + K_\epsilon^{-1} + K_\epsilon^{-a} + K_\epsilon^{-b}\right). \tag{90}$$

By definition of $K_\epsilon$ [c.f. (36)], we have that $\mathbb{E}[\|\nabla J(\theta_j)\|^2] > \epsilon$ for all $j = 1, \ldots, K_\epsilon$, so

$$\epsilon \leq \frac{1}{K_\epsilon}\sum_{j=1}^{K_\epsilon}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \mathcal{O}\left(K_\epsilon^{a-1} + K_\epsilon^{-1} + K_\epsilon^{-a} + K_\epsilon^{-b}\right). \tag{91}$$

Defining $\ell = \min\{a, 1-a, b\}$, the preceding expression then implies

$$\epsilon \leq \mathcal{O}(K_\epsilon^{-\ell}), \tag{92}$$

which by inverting the expression, yields the sample complexity

$$K_\epsilon \leq \mathcal{O}(\epsilon^{-1/\ell}). \tag{93}$$

**Case (ii):** $b = 1$ Now consider the case where $b = 1$. Set $T_C(k) = k+1$ and $H(k) = k$. Again, using the integration rule, and that $\sum_{j=k-N}^{k}(j+1)^{-1} \leq \log(k+1) - \log(k-N)$, we substitute into (87) which yields

$$\begin{aligned}\sum_{j=k-N}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq & C_4 k^a + C_\nabla C_1 \sum_{j=k-N}^{k}\gamma^j + \frac{L\sigma^2}{1-a}\left(k^{1-a} - (k-N-1)^{1-a}\right) \\ & + CL_1\left(\log(k+1) - \log(k-N)\right).\end{aligned} \tag{94}$$

Divide both sides by $k$ and fix $N = k-1$

$$\frac{1}{k}\sum_{j=1}^{k}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq C_4 k^{a-1} + C_\nabla C_1 \gamma^{-1} k^{-1}\sum_{j=1}^{k}\gamma^j + \frac{L\sigma^2}{1-a}k^{-a} + CL_1\frac{\log(k+1)}{k}. \tag{95}$$

Let $k = K_\epsilon$ in the preceding expression, which then becomes

$$\frac{1}{K_\epsilon}\sum_{j=1}^{K_\epsilon}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \mathcal{O}\left(K_\epsilon^{a-1} + K_\epsilon^{-1} + K_\epsilon^{-a} + \frac{\log(K_\epsilon+1)}{K_\epsilon}\right). \tag{96}$$

Again, by definition of $K_\epsilon$ [c.f. (36)], we have that $\mathbb{E}[\|\nabla J(\theta_j)\|^2] > \epsilon$ for all $j = 1, \ldots, K_\epsilon$, so

$$\epsilon \leq \frac{1}{K_\epsilon}\sum_{j=1}^{K_\epsilon}\mathbb{E}[\|\nabla J(\theta_j)\|^2] \leq \mathcal{O}\left(K_\epsilon^{a-1} + K_\epsilon^{-1} + K_\epsilon^{-a} + \frac{\log(K_\epsilon+1)}{K_\epsilon}\right). \tag{97}$$

Optimizing over $a$, we have

$$\epsilon \leq \mathcal{O}\left(K_\epsilon^{-\frac{1}{2}}\right) \quad \text{for } b > \frac{1}{2} \tag{98}$$

On the other hand,

$$\epsilon \leq \mathcal{O}\left(K_\epsilon^{-b}\right). \text{ for } b \leq 1/2 \tag{99}$$

Fix $\ell = \min\{1/2, b\}$, then

$$\epsilon \leq \mathcal{O}(K_\epsilon^{-\ell}), \tag{100}$$

which implies

$$K_\epsilon \leq \mathcal{O}(\epsilon^{-1/\ell}). \tag{101}$$

This concludes the proof.