Non-asymptotic Convergence of Adam-type Reinforcement Learning Algorithms under Markovian Sampling

Huaqing Xiong,[1*] Tengyu Xu,[1*] Yingbin Liang,[1] Wei Zhang[23†]

[1]Department of Electrical and Computer Engineering, The Ohio State University

[2]Department of Mechanical and Energy Engineering, Southern University of Science and Technology

[3]Peng Cheng Laboratory

{xiong.309, xu.3260, liang.889}@osu.edu; zhangw3@sustech.edu.cn

Abstract

Despite the wide applications of Adam in reinforcement learning (RL), the theoretical convergence of Adam-type RL algorithms has not been established. This paper provides the first such convergence analysis for two fundamental RL algorithms of policy gradient (PG) and temporal difference (TD) learning that incorporate AMSGrad updates (a standard alternative of Adam in theoretical analysis), referred to as PG-AMSGrad and TD-AMSGrad, respectively. Moreover, our analysis focuses on Markovian sampling for both algorithms. We show that under general nonlinear function approximation, PG-AMSGrad with a constant stepsize converges to a neighborhood of a stationary point at the rate of $\mathcal{O}(1/T)$ (where T denotes the number of iterations), and with a diminishing stepsize converges exactly to a stationary point at the rate of $\mathcal{O}(\log^2 T/\sqrt{T})$. Furthermore, under linear function approximation, TD-AMSGrad with a constant stepsize converges to a neighborhood of the global optimum at the rate of $\mathcal{O}(1/T)$, and with a diminishing stepsize converges exactly to the global optimum at the rate of $\mathcal{O}(\log T/\sqrt{T})$. Our study develops new techniques for analyzing the Adam-type RL algorithms under Markovian sampling.

Introduction

Reinforcement learning (RL) aims to study how an agent learns a policy through interacting with its environment to maximize the accumulative reward. RL has so far accomplished tremendous success in various applications such as playing video games (Mnih et al. 2013), bipedal walking (Castillo et al. 2019), online advertising (Pednault, Abe, and Zadrozny 2002), to name a few. In general, there are two widely used classes of RL algorithms: policy-based methods and value function based methods.

For the first class, policy gradient (PG) (Sutton et al. 2000) is a basic algorithm which has motivated many advanced policy-based algorithms including actor-critic (Konda and Tsitsiklis 2000), DPG (Silver et al. 2014), TRPO (Schulman et al. 2015), PPO (Schulman et al. 2017), etc. The idea of PG (Sutton et al. 2000) is to parameterize the policy and optimize a target accumulated reward function by (stochastic) gradient descent. The asymptotic and non-asymptotic convergence have been characterized for PG in various scenarios, which will be further discussed in Related Work.

For the second class of value function based algorithms, temporal difference (TD) learning (Sutton 1988) is a fundamental algorithm which has motivated more advanced algorithms such as Q-learning (Watkins and Dayan 1992), SARSA (Rummery and Niranjan 1994), etc. TD (Sutton 1988) typically parameterizes the value function of an unknown policy and iteratively finds the true value function or its estimator by following the (projected) Bellman operation, which is also analogous to a stochastic gradient descent (SGD) update. The theoretic analysis has been established for TD in various scenarios, which will be discussed in Related Work.

Despite extensive exploration, all the existing theoretical studies of PG and TD have focused on SGD-type updates without adaption on the stepsize. In practice, however, the adaptive momentum estimation (Adam) method (Kingma and Ba 2015) has been commonly used in RL (Bello et al. 2017; Stooke and Abbeel 2018). There is so far no theoretic guarantee established to show that RL algorithms that incorporate the Adam-type updates have provable convergence. *The goal of this paper is to theoretically characterize the convergence rate of the Adam-type PG and TD algorithms.* Such a study requires new technical tools to analyze the Adam-type algorithms under *Markovian sampling*. The analysis does not follow easily from the existing studies of Adam-type algorithms in optimization, which usually assume independent and identically distributed (i.i.d.) sampling. It does not follow from the existing studies of SGD-type RL algorithms because of the unique complication of adaptive momentum update coupled to the bias errors in Markovian sampling.

Our Contribution

We provide the first non-asymptotic convergence guarantee for Adam-type PG and TD algorithms that incorporate the update rule of AMSGrad (referred to as PG-AMSGrad and TD-AMSGrad, respectively). Our techniques also lead

[*]Equal contribution

[†]Corresponding author

to an improved result for the vanilla PG. Specifically, **(1)** we first show that under general nonlinear function approximation, PG-AMSGrad with a constant stepsize[1] converges to a neighborhood of a stationary point at a rate of $\mathcal{O}(1/T)$ (where T denotes the number of iterations), and with a diminishing stepsize converges exactly to a stationary point at a rate of $\mathcal{O}(\log^2 T/\sqrt{T})$. **(2)** Furthermore, under linear function approximation, TD-AMSGrad with a constant stepsize converges to a neighborhood of the global optimum at a rate of $\mathcal{O}(1/T)$, and with a diminishing stepsize converges exactly to the global optimum at a rate of $\mathcal{O}(\log T/\sqrt{T})$. **(3)** By adapting our technical tools to analyze the vanilla PG with the SGD update under Markovian sampling, we obtain an orderwisely better computational complexity than the existing works, which is summarized in Table 1.

Technically, we develop new techniques to analyze Adam-type RL algorithms which are not available in the existing RL and MDP literature, nor in optimization literature. Specifically, **(1)** the Adam-type (i.e., AMSGrad) update in PG and TD algorithms causes a unique bias error in gradient estimation due to Markovian sampling, which does not exist in conventional optimization with i.i.d. sampling, and takes much more challenging form than that in SGD-type RL algorithms due to the Adam-type update. Our analysis is the first to bound such a bias error. **(2)** The sampling process in PG is further subject to a time-varying Markov chain so that the sampling distribution is changing over time. Thus, we develop a novel technique to provide finite-time error bounds by jointly exploiting how fast Markov chain changes together with the ergodicity of each instantaneous Markov chain. Such a technique sharpens the existing analysis for vanilla PG with orderwise improvement. We then apply this new technique further to address the coupling introduced by AMSGrad to PG.

Related Work

Due to the rapidly growing theoretical studies on RL, we review only the most relevant studies below.

Convergence analysis of PG: Asymptotic convergence of PG based on stochastic approximation (SA) was established in Williams (1992); Baxter and Bartlett (2001); Sutton et al. (2000); Kakade (2002); Pirotta, Restelli, and Bascetta (2015); Tadić, Doucet et al. (2017). In specific RL problems such as LQR, PG has been proved to converge to the optimal policy (Fazel et al. 2018; Malik et al. 2019; Tu and Recht 2019). Under convex policy function approximation, Bhandari and Russo (2019) also showed that PG can find the optimal policy. Under the general nonconvex approximation, Shen et al. (2019); Papini, Pirotta, and Restelli (2017); Papini et al. (2018); Papini, Pirotta, and Restelli (2019); Xu, Gao, and Gu (2019, 2020) characterized the convergence rate for PG and variance reduced PG to a stationary point under finite horizon, and Zhang et al. (2019); Karimi et al. (2019) provided the convergence rate for PG in the infinite-horizon scenario. (Wang et al. 2020) studied natural PG with neural network function approximation in an overparameterized regime. Convergence analysis has also been established for the variants of PG, such as TRPO/PP0 (Shani, Efroni, and Mannor 2020; Liu et al. 2019), Actor-Critic (Xu, Wang, and Liang 2020a,b), etc. This paper studies the infinite-horizon scenario, but focuses on Adam-type PG, which has not been studied in the literature.

Convergence analysis of TD: Originally proposed in Sutton (1988), TD learning with function approximation aroused great interest in analyzing its convergence. While a general TD may not converge as pointed out in Baird (1995); Györfi and Walk (1996), Tsitsiklis and Van Roy (1997) provided conditions to ensure asymptotic convergence of TD with linear function approximation under i.i.d. sampling. Other results on asymptotic convergence using the tools from linear SA were provided in Kushner and Yin (2003); Benveniste, Métivier, and Priouret (2012). Non-asymptotic convergence was established for TD under i.i.d. sampling in, e.g., Dalal et al. (2018); Bhandari, Russo, and Singal (2018); Lakshminarayanan and Szepesvari (2018), and under Markovian sampling in, e.g., Bhandari, Russo, and Singal (2018); Srikant and Ying (2019); Hu and Syed (2019). The convergence rate of TD with nonlinear function approximation has recently been studied in Cai et al. (2019) for overparameterized neural networks using i.i.d. samples. In contrast to the aforementioned work on TD with the SGD-type updates, this paper studies Adam-type TD under Markovian sampling.

Adaptive reinforcement learning algorithms: Adaptivity has been applied to RL algorithms to improve the performance. (Shani, Efroni, and Mannor 2020) used an adaptive proximity term to study the convergence of TRPO. An adaptive batch size was adopted to improve the policy performance (Papini, Pirotta, and Restelli 2017) and reduce the variance (Ji et al. 2020) of PG. The aforementioned papers did not study how the adaptive learning rate can affect the performance of PG or TD. More recently, concurrent works also analyzed TD(0) and TD(λ) (Sun et al. 2020) incorporating adaptive gradient descent (AdaGrad) updates and Q-learning (Weng et al. 2020) with AMSGrad updates. However, this paper provides the first convergence guarantee when Adam-type updates are applied to PG and TD.

Convergence analysis of Adam-type algorithms in conventional optimization: Adam was proposed in Kingma and Ba (2015) to speed up the training of deep neural networks, but the vanilla Adam was shown not to converge in Reddi, Kale, and Kumar (2018). Instead, AMSGrad was proposed as a slightly modified version to justify the theoretic performance of Adam. Its regret bounds were characterized in Reddi, Kale, and Kumar (2018); Tran and Phong (2019) for online convex optimization. Recently, AMSGrad was proved to converge to a stationary point for nonconvex optimization (Zou et al. 2019; Zhou et al. 2018; Chen et al. 2019a). Our study provides the first convergence guarantee for Adam-type algorithms in RL, where time-varying Markovian sampling poses the key difference and challenge in our analysis from conventional optimization.

[1]The "stepsize" here refers to the basic stepsize α in the AMSGrad update (4). The overall learning rate of the algorithm is determined by the basic stepsize α, hyperparameters β_1 and β_2, and the first and second moments of gradients as given in (1)-(4), and is hence adaptive during the AMSGrad iteration.

Preliminary

In this section, we provide the necessary background for the problems that we study in this paper.

Markov Decision Process

We consider the standard RL setting, where an agent interacts with a (possibly stochastic) environment (e.g. process or system dynamics). This interaction is usually modeled as a discrete-time discounted Markov Decision Processes (MDPs), described by a tuple $(\mathcal{S}, \mathcal{A}, \mathbb{P}, R, \gamma, \zeta)$, where $\mathcal{S}$ is the state space which is possibly countably infinite, $\mathcal{A}$ is the finite action space with cardinality $|\mathcal{A}|$, $\mathbb{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \mapsto [0, 1]$ is the probability kernel for the state transitions, e.g., $\mathbb{P}(\cdot|s, a)$ denotes the probability distribution of the next state given the current state s and action a. In addition, $R : \mathcal{S} \times \mathcal{A} \mapsto [0, R_{\max}]$ is the reward function mapping state-action pairs to a bounded subset of $\mathbb{R}$, $\gamma \in (0, 1)$ is the discount factor, and ζ denotes the initial state distribution. The agent's decision is captured by the policy $\pi := \pi(\cdot|s)$ which characterizes the density function over the action space $\mathcal{A}$ at the state $s \in \mathcal{S}$. We denote $\nu := \nu_\pi$ as the stationary distribution of the transition kernel $\mathbb{P}$ for a given π. In addition, we define the γ-discounted stationary visitation distribution of the policy π as $\mu_\pi(s) = \sum_{t=1}^{\infty} \gamma^t P_{\zeta,\pi}(s_t = s)$. Further, we denote $\mu_\pi(s, a) = \mu_\pi(s)\pi(a|s)$ as the (discounted) state-action visitation distribution.

Update Rule of AMSGrad

Although Adam (Kingma and Ba 2015) has gained great success in practice, it was shown not to converge even in the simple convex setting (Reddi, Kale, and Kumar 2018). Instead, a slightly modified version called AMSGrad (Reddi, Kale, and Kumar 2018) has been widely used to understand the success of adaptive momentum optimization algorithms. Given a gradient g_t at time t, the generic form of AMSGrad is given by

$$m_t = (1 - \beta_1)m_{t-1} + \beta_1 g_t; \tag{1}$$

$$v_t = (1 - \beta_2)\hat{v}_{t-1} + \beta_2 g_t^2; \tag{2}$$

$$\hat{v}_t = \max(\hat{v}_{t-1}, v_t); \quad \hat{V}_t = diag(\hat{v}_{t,1}, \ldots, \hat{v}_{t,d}); \tag{3}$$

$$\theta_{t+1} = \theta_t - \alpha_t \hat{V}_t^{-\frac{1}{2}} m_t, \tag{4}$$

where α_t is the stepsize, and β_1, β_2 are two hyper-parameters. In addition, m_t, v_t given in (2) are viewed as the estimation of the first moment and second moment, respectively, which play important roles in adapting the learning rate as in (4). Compared to Adam, the main difference of AMSGrad lies in the first equation of (4), which guarantees the sequence $\hat{v}_t$ to be non-decreasing, whereas Adam does not require this. Such a difference is considered to be a central reason causing the non-convergent behavior of Adam (Reddi, Kale, and Kumar 2018; Chen et al. 2019a).

Notations

We use $\|x\| := \|x\|_2 = \sqrt{x^T x}$ to denote the ℓ_2-norm of a vector x, and use $\|x\|_\infty = \max_i |x_i|$ to denote the infinity norm. When x, y are both vectors, $x/y, xy, x^2, \sqrt{x}$ are all calculated in the element-wise manner, which are used in the update of AMSGrad. We denote $[n] = \{1, 2, \ldots, n\}$, and $\lceil x \rceil \in \mathbb{Z}$ as the integer such that $\lceil x \rceil - 1 \leq x < \lceil x \rceil$.

Convergence of PG-AMSGrad under Markovian Sampling

In this section, we study the convergence of an Adam-type policy gradient algorithm (PG-AMSGrad) with nonlinear function approximation and under non-i.i.d. sampling.

Policy Gradient and PG-AMSGrad

Suppose that policies are parameterized by $\theta \in \mathbb{R}^d$ and form a policy class $\Pi := \{\pi_\theta | \theta \in \mathbb{R}^d\}$, which in general is a nonlinear function class. The policy gradient method is usually used to solve the following *infinite-horizon* optimization problem:

$$\underset{\theta \in \mathbb{R}^d}{\text{maximize}} \quad J(\theta) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t)\right]. \tag{5}$$

The gradient of $J(\theta)$ with respect to θ is captured by the policy gradient theorem for infinite-horizon MDP with the discounted reward (Sutton et al. 2000), and is given by

$$\nabla_\theta J(\theta) = \mathop{\mathbb{E}}_{\mu_\theta}\left[Q^{\pi_\theta}(s, a)\nabla_\theta \log(\pi_\theta(a|s))\right], \tag{6}$$

where the expectation is taken over the discounted state-action visitation distribution $\mu_\theta := \mu_{\pi_\theta}(s, a)$, and $Q^{\pi_\theta}(s, a)$ denotes the Q-function for an initial state-action pair (s, a) defined as

$$Q^{\pi_\theta}(s, a) = \mathbb{E}\left[\sum_{t=1}^{\infty} \gamma^t R(s_t, a_t)\Big| s_1 = s, a_1 = a\right].$$

In addition, we refer to $\nabla_\theta \log \pi_\theta(a|s)$ as the score function corresponding to the policy π_θ.

Since the transition probability is unknown, the policy gradient in (6) needs to be estimated via sampling. The Q-function $Q^{\pi_\theta}(s, a)$ and the score function are typically estimated by independent samples. First, at each time t, we draw a sample trajectory to provide an estimated Q-function $\hat{Q}^{\pi_\theta}(s, a)$ based on the algorithm EstQ (Zhang et al. 2019) (see Algorithm 3 in Appendix A for details). Such an estimator has been shown to be unbiased (Zhang et al. 2019). That is, if we use O^q to denote the randomness including the samples and horizon in EstQ, then we have

$$\mathop{\mathbb{E}}_{O^q} \hat{Q}^{\pi_\theta}(s, a) = Q^{\pi_\theta}(s, a), \quad \forall (s, a). \tag{7}$$

Next, based on the policy gradient theorem for infinite-horizon MDP with the discounted reward (Sutton et al. 2000), the gradient estimator to approximate $\nabla_\theta J(\theta)$ at time t is given by

$$g_t := g(\theta_t; s_t, a_t) = \hat{Q}^{\pi_{\theta_t}}(s_t, a_t)\nabla_{\theta_t} \log(\pi_{\theta_t}(a_t|s_t)), \tag{8}$$

where the estimated Q-function $\hat{Q}^{\pi_\theta}(s, a)$ is obtained by the algorithm EstQ (Zhang et al. 2019) (see Algorithm 3 in Appendix A for details), and the score function $\nabla_\theta \log \pi_\theta(a|s)$

Algorithm 1 PG-AMSGrad

Input: $\alpha, \theta_1, \beta_1, \beta_2, m_0 = 0, \hat{v}_0 = 0, t = 1, s_1 \sim \zeta(\cdot), a_1 \sim \pi_{\theta_1}(\cdot|s)$.
while not converge **do**
  Assign stepsize α_t.
  Obtain $\hat{Q}^{\pi_{\theta_t}}(s_t, a_t) \leftarrow \text{EstQ}(s_t, a_t, \theta_t)$.
  Compute $g_t = \hat{Q}^{\pi_{\theta_t}}(s_t, a_t) \nabla_{\theta_t} \log(\pi_{\theta_t}(a_t|s_t))$.
  $m_t = (1-\beta_1) m_{t-1} + \beta_1 g_t$.
  $v_t = (1-\beta_2)\hat{v}_{t-1} + \beta_2 g_t^2$.
  $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$, $\hat{V}_t = diag(\hat{v}_{t,1}, \ldots, \hat{v}_{t,d})$.
  $\theta_{t+1} = \theta_t - \alpha_t \hat{V}_t^{-\frac{1}{2}} m_t$.
  $t \leftarrow t+1$.
  Sample $s_t \sim \hat{P}(\cdot|s_{t-1}, a_{t-1}), a_t \sim \pi_{\theta_t}(\cdot|s_t)$.
end while

is estimated by samples $\{(s_t, a_t)\}$ drawn following the policy π_{θ_t} and the transition function $\hat{P}(\cdot|s_t, a_t) = \gamma \mathbb{P}(\cdot|s_t, a_t) + (1-\gamma)\zeta(\cdot)$ proposed in Konda (2002) with $\zeta(\cdot)$ being the initial distribution and $\mathbb{P}$ being the transition probability of the original MDP. Such a transition probability guarantees the MDP to converge to the state-action visitation distribution. We then apply such a gradient estimator to update the policy parameter by the AMSGrad update given in (1)-(4), and obtain PG-AMSGrad as in Algorithm 1.

We note that the gradient estimator obtained in (8) is *biased*, because the score function is estimated by a sequence of Markovian samples. We will show that such a biased gradient estimator is in fact computationally more efficient than the unbiased estimator used in the existing literature (Zhang et al. 2019). Our main technical novelty here lies in developing techniques to analyze the biased estimator under the AMSGrad update for PG.

Technical Assumptions

In the following, we specify some technical assumptions in our convergence analysis.

We consider a general class of parameterized policy functions that satisfy the following assumption.

Assumption 1. *Assume that the parameterized policy π_θ is differentiable with respect to θ, and the score function $\nabla_\theta \log \pi_\theta(a|s)$ corresponding to $\pi_\theta(\cdot|s)$ exists. In addition, we assume both the policy function and the score function are Lipschitz continuous with the parameters L_π and L, respectively, i.e., for all $\theta_1, \theta_2 \in \mathbb{R}^d$,*

$$|\pi_{\theta_1}(a|s) - \pi_{\theta_2}(a|s)| \le L_\pi \|\theta_1 - \theta_2\|;$$

$$\|\nabla_{\theta_1} \log(\pi_{\theta_1}(a|s)) - \nabla_{\theta_2} \log(\pi_{\theta_2}(a|s))\| \le L \|\theta_1 - \theta_2\|.$$

Further, the score function is uniformly bounded by c_Θ, i.e., for any $(s,a) \in \mathcal{S} \times \mathcal{A}$ and any θ,

$$\|\nabla_\theta \log(\pi_\theta(a|s))\| \le c_\Theta.$$

The above assumption is standard in the studies of PG with nonconvex function approximation (Zhang et al. 2019; Xu, Gao, and Gu 2019; Papini et al. 2018).

In Algorithm 1, we sample a data trajectory using the transition kernel $\hat{P}$ and the policy π_{θ_t}. Such a sequence of samples are non-i.i.d. and follow a Markovian distribution. We assume that the MDP and the policies we consider satisfy the following standard mixing property.

Assumption 2. *For any $\theta \in \mathbb{R}^d$, there exist constant $\sigma > 0$ and $\rho \in (0,1)$ such that*

$$\sup_{s \in \mathcal{S}} \|P(s_t \in \cdot | s_1 = s) - \mu_\theta(\cdot)\|_{TV} \le \sigma \rho^t \quad \forall t,$$

where $\|\mu_1 - \mu_2\|_{TV}$ denotes the total-variation norm (or the total-variation distance between two probability measures μ_1 and μ_2).

This assumption holds for irreducible and aperiodic Markov chains (Mitrophanov 2005), and is widely adopted in the theoretical analysis of RL algorithms under Markovian sampling (Bhandari, Russo, and Singal 2018; Chen et al. 2019b; Zou, Xu, and Liang 2019; Karimi et al. 2019).

Convergence of PG-AMSGrad

In this section, we provide the convergence analysis of PG-AMSGrad as given in Algorithm 1. We first consider the case with a constant stepsize, and then provide the result with a diminishing stepsize.

Although AMSGrad has been studied in conventional optimization, our analysis of PG-AMSGrad mainly deals with the following new challenges arising in RL. First, samples here are generated via an MDP and distributed in a non-i.i.d. fashion. Thus the gradient estimator is biased and we need to control the bias with a certain upper bound scaled by the stepsize. Second, the sampling distribution also changes over time, which causes additional complication. Thus, our technical development mainly handles the above two challenges under the adaptive momentum update rule of AMSGrad. We provide the convergence results that we obtain and relegate the main proofs to the appendices in the following.

We first provide the Lipschitz properties for the true policy gradient and its estimator, which are useful for establishing the convergence. Recall that in Algorithm 1, the gradient estimator $g_t = \hat{Q}^{\pi_{\theta_t}}(s_t, a_t) \nabla_{\theta_t} \log(\pi_{\theta_t}(a_t|s_t))$ at time t is obtained by using the Q-function estimator generated by the EstQ algorithm (see Appendix A). Note that $\hat{Q}^{\pi_\theta}(s,a)$ is an unbiased estimator of $Q^{\pi_\theta}(s,a)$ for all (s,a) (Zhang et al. 2019), and the samples for estimation are independent of those for other steps in PG-AMSGrad except the initial sample. Taking expectation over the randomness in EstQ at time t (denoted as O_t^q), we obtain an estimator $\nabla_{\theta_t} \tilde{J}(\theta_t; s_t, a_t)$ defined as

$$\begin{aligned}\nabla_{\theta_t} \tilde{J}(\theta_t; s_t, a_t) &:= \mathop{\mathbb{E}}_{O_t^q}[g_t] \\ &= \mathop{\mathbb{E}}_{O_t^q}\left[\hat{Q}^{\pi_{\theta_t}}(s_t, a_t) \nabla_{\theta_t} \log(\pi_{\theta_t}(a_t|s_t))\right] \\ &= Q^{\pi_{\theta_t}}(s_t, a_t) \nabla_{\theta_t} \log(\pi_{\theta_t}(a_t|s_t)). \end{aligned} \tag{9}$$

We obtain the Lipschitz properties of $\nabla_\theta \tilde{J}(\theta; s, a)$ and $\nabla_\theta J(\theta)$ in the following lemma.

Lemma 1. *(Lipschitz property of policy gradient) Under Assumptions 1 and 2, the policy gradient $\nabla_\theta J(\theta)$ defined in* (6) *is Lipschitz continuous with the parameter c_J, i.e., $\forall \theta_1, \theta_2 \in \mathbb{R}^d$,*

$$\|\nabla_{\theta_1} J(\theta_1) - \nabla_{\theta_2} J(\theta_2)\| \leq c_J \|\theta_1 - \theta_2\|, \quad (10)$$

where the constant coefficient $c_J = \frac{R_{\max} L}{1-\gamma} + \frac{(1+c_\Theta)R_{\max}}{1-\gamma} \cdot |\mathcal{A}| L_\pi \left(1 + \lceil \log_\rho \sigma^{-1} \rceil + \frac{1}{1-\rho}\right)$. Further, the policy gradient estimator $\nabla_\theta \tilde{J}(\theta; s, a)$ defined in (9) *is also Lipschitz continuous with the parameter $c_{\tilde{J}}$, i.e., $\forall \theta_1, \theta_2 \in \mathbb{R}^d, \forall (s,a) \in \mathcal{S} \times \mathcal{A}$,*

$$\left\|\nabla_{\theta_1} \tilde{J}(\theta_1; s, a) - \nabla_{\theta_2} \tilde{J}(\theta_2; s, a)\right\| \leq c_{\tilde{J}} \|\theta_1 - \theta_2\|, \quad (11)$$

where $c_{\tilde{J}} = \frac{R_{\max} L}{1-\gamma} + c_\Theta |\mathcal{A}| L_\pi \left(1 + \lceil \log_\rho \sigma^{-1} \rceil + \frac{1}{1-\rho}\right)$.

Next, we provide the main convergence results. The first theorem characterizes the convergence of PG-AMSGrad with a constant stepsize. Recall that the stepsize refers to the parameter α in AMSGrad update (4), not the overall learning rate.

Theorem 1. *(Convergence of PG-AMSGrad with constant stepsize) Fix β_1, β_2 in Algorithm 1. Initialize Algorithm 1 such that $|g_{1,i}| \geq G_0$ for all $i \in [d]$ with some $G_0 > 0$. Suppose Assumptions 1 and 2 hold. Let $\alpha_t = \alpha$ for $t = 1, \ldots, T$. Then after running T steps of PG-AMSGrad as given in Algorithm 1, we have:*

$$\min_{t \in [T]} \mathbb{E}\left[\|\nabla_{\theta_t} J(\theta_t)\|^2\right] \leq \frac{C_1}{T} + \alpha C_2,$$

where $C_1 = \frac{G_\infty \mathbb{E}[J(z_1)]}{\alpha} + \frac{dG_\infty^3}{G_0(1-\beta_1)} + \frac{2G_\infty \tau^*}{G_0} G_\infty^2 + \frac{dc_J \alpha G_\infty (3\beta_1^2 + 2(1-\beta_1)(1-\beta_1/\beta_2))}{(1-\beta_1)(1-\beta_2)(1-\beta_1/\beta_2)}$, $C_2 = \frac{G_\infty^3}{G_0}\left[\frac{(3c_J + c_{\tilde{J}})\tau^*}{G_0} + d + \frac{dL_\pi G_\infty (2\tau^* + (\tau^*)^2)}{2G_0}\right]$. with $c_J, c_{\tilde{J}}$ defined in Lemma 1 in Appendix C, $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha\}$ and $G_\infty = \frac{c_\Theta R_{\max}}{1-\gamma}$.

Theorem 1 indicates that under the constant stepsize, PG-AMSGrad converges to a neighborhood of a stationary point at a rate of $\mathcal{O}\left(\frac{1}{T}\right)$. The size of the neighborhood can be controlled by the stepsize α. One can observe that α controls a tradeoff between the convergence rate and the convergence accuracy. Decreasing α improves the convergence accuracy, but slows down the convergence, since the coefficient C_1 contains α in the denominator. To balance such a tradeoff, we set the stepsize $\alpha_t = \frac{1}{\sqrt{T}}$. In this case, the mixing time becomes $\tau^* = \mathcal{O}(\log T)$ and thus PG-AMSGrad converges to a stationary point with a rate of $\mathcal{O}\left(\frac{\log^2 T}{\sqrt{T}}\right)$.

In the following, we adopt a diminishing stepsize to eliminate the convergence error and obtain the exact convergence.

Theorem 2. *(Convergence of PG-AMSGrad with diminishing stepsize) Suppose the same conditions of Theorem 1 hold, and let $\alpha_t = \frac{\alpha}{\sqrt{t}}$ for $t = 1, \ldots, T$. Then running T steps of PG-AMSGrad as given in Algorithm 1, we have:*

$$\min_{t \in [T]} \mathbb{E}\left[\|\nabla_{\theta_t} J(\theta_t)\|^2\right] \leq \frac{C_1}{T} + \frac{C_2}{\sqrt{T}},$$

where $C_1 = \frac{f_1 G_\infty}{\alpha} + \frac{2dc_J \alpha G_\infty}{1-\beta_2} + \frac{2\tau^* G_\infty}{G_0} G_\infty^2 + \frac{3dc_J \beta_1^2 \alpha G_\infty}{(1-\beta_1)(1-\beta_2)(1-\beta_1/\beta_2)}$, $C_2 = \frac{R_{\max} G_\infty}{\alpha(1-\gamma)} + \frac{dG_\infty^3}{G_0(1-\beta_1)} + \frac{\alpha G_\infty^3}{G_0}\left[\frac{2(3c_J + c_{\tilde{J}})\tau^*}{G_0} + d\left(1 + \frac{L_\pi G_\infty (\tau^* + (\tau^*)^2)}{G_0}\right)\right]$ with $c_J, c_{\tilde{J}}$ defined in Lemma 1 in Appendix C, $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha_T = \frac{\alpha}{\sqrt{T}}\}$ and $G_\infty = \frac{c_\Theta R_{\max}}{1-\gamma}$.

Theorem 2 indicates that under a diminishing stepsize, PG-AMSGrad can converge exactly to a stationary point. Since $\tau^* = \mathcal{O}(\log T)$, the convergence rate is given by $\mathcal{O}\left(\frac{\log^2 T}{\sqrt{T}}\right)$.

Theorems 1 and 2 indicate that under both constant and diminishing stepsizes, PG-AMSGrad finds a stationary point efficiently with guaranteed convergence. However, there is a tradeoff between the convergence rate and accuracy. With a constant stepsize, PG-AMSGrad can converge faster but only to a neighborhood of a stationary point whose size is controlled by the stepsize, whereas a diminishing stepsize yields a better convergence accuracy, but attains a lower convergence rate.

Improved Analysis on SGD-type PG under Markovian Data

Although our focus in this paper is on the Adam-type PG, our techniques also yield an improved convergence rate for the SGD-type PG under infinite horizon Markovian sampling over the existing studies (Zhang et al. 2019; Karimi et al. 2019). In the following, we present such results and make the comparisons to illustrate the novelty of our analysis.

We consider the SGD-type PG algorithm that uses the same gradient estimator and sampling strategy as those of Algorithm 1, but adopts the SGD update (i.e., $\theta_{t+1} = \theta_t - \alpha_t g_t$) rather than the AMSGrad update. We call such an algorithm as PG-SGD. The following proposition characterizes the convergence rate for PG-SGD.

Proposition 1. *Suppose Assumptions 1 and 2 hold. After running T steps of PG-SGD with a constant stepsize $\alpha_t = \alpha$ for $t = 1, \ldots, T$, we have:*

$$\min_{t \in [T]} \mathbb{E}\left[\|\nabla_{\theta_t} J(\theta_t)\|^2\right] \leq \frac{J(\theta_1)/\alpha + 2G_\infty^2 \tau^*}{T} + \alpha C_1,$$

where $C_1 = G_\infty^2 \left[\frac{c_J}{2} + (3c_J + c_{\tilde{J}})\tau^* + 1 + \frac{L_\pi G_\infty (2\tau^* + (\tau^*)^2)}{2}\right]$, with $c_J, c_{\tilde{J}}$ defined in Lemma 1 in Appendix C, $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha\}$ and $G_\infty = \frac{c_\Theta R_{\max}}{1-\gamma}$. Furthermore, after running T steps of PG-SGD with a diminishing stepsize $\alpha_t = \frac{1-\gamma}{\sqrt{t}}$ for $t = 1, \ldots, T$, we have:

$$\min_{t \in [T]} \mathbb{E}\left[\|\nabla_{\theta_t} J(\theta_t)\|^2\right] \leq \frac{C_2}{(1-\gamma)T} + \frac{C_3}{(1-\gamma)^2 \sqrt{T}},$$

where $C_2 = J(\theta_1) + 2(1-\gamma)^2 G_\infty^2 \tau^*$, $C_3 = R_{\max} + (1-\gamma)^3 G_\infty^2 \left[c_J + 2(3c_J + c_{\tilde{J}})\tau^* + 1 + L_\pi G_\infty (\tau^* + (\tau^*)^2)\right]$, with $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha_T = \frac{1-\gamma}{\sqrt{T}}\}$.

We next compare Proposition 1 with two recent studies on the infinite-horizon PG under non-i.i.d. sampling.

PG algorithms	Stepsize	Convergence rate
Karimi et al. (2019)	$\frac{c_1}{\sqrt{t}}$	$\tilde{\mathcal{O}}\left(\frac{1}{(1-\gamma)^2\sqrt{T}}+c_2\right)$
Zhang et al. (2019)	$(1-\gamma)^2(1-\sqrt{\gamma})$	$\tilde{\mathcal{O}}\left(\frac{1}{(1-\gamma)^3(1-\sqrt{\gamma})T}+\frac{1}{(1-\gamma)^3(1-\sqrt{\gamma})}\right)$
	$\frac{1}{\sqrt{t}}$	$\tilde{\mathcal{O}}\left(\frac{1}{(1-\gamma)^5(1-\sqrt{\gamma})^2\sqrt{T}}\right)$
This work	$1-\gamma$	$\tilde{\mathcal{O}}\left(\frac{1}{(1-\gamma)^2T}+\frac{1}{(1-\gamma)^2}\right)$
	$\frac{1-\gamma}{\sqrt{t}}$	$\tilde{\mathcal{O}}\left(\frac{1}{(1-\gamma)^2\sqrt{T}}\right)$

Table 1: Comparison of convergence rate (with diminishing stepsize) and error bound (with constant stepsize) for policy gradient with non-i.i.d. sampling. Some remarks on the table are as follows. (a) c_1, c_2 are time-independent positive constants. (b) The convergence rate of Karimi et al. (2019) includes the best existing dependence of Lipshitz constant on $1-\gamma$ for fair comparison with other studies.

Table 1 summarizes such the comparison of the convergence rate and error bound among the relevant studies. First, Karimi et al. (2019) studied infinite-horizon PG with a biased gradient estimator. Their convergence analysis has a *non-vanishing* error even with a diminishing stepsize. In contrast, we obtain a fine-grained bound on the bias and show that PG converges *exactly* to a stationary point under the diminishing stepsize.

Another closely related such study was by Zhang et al. (2019), but their algorithm adopts an unbiased gradient estimator at the cost of using more samples. As a comparison, Proposition 1 indicates that PG-SGD with a biased gradient estimator attains a better convergence rate and accuracy. More specifically, under the constant stepsize, (Zhang et al. 2019, Corollary 4.4) showed that their PG algorithm converges with an optimized error bound of $\mathcal{O}\left(\frac{1}{(1-\gamma)^3(1-\sqrt{\gamma})}\right)$, whereas PG-SGD with a biased gradient estimator achieves a much smaller error bound $\mathcal{O}\left(\frac{1}{(1-\gamma)^2}\right)$ by taking $\alpha = 1-\gamma$. Similarly, under the diminishing stepsize, (Zhang et al. 2019, Theorem 4.3) showed that their PG algorithm converges at a rate of $\mathcal{O}\left(\frac{1}{(1-\gamma)^5(1-\sqrt{\gamma})^2\sqrt{T}}\right)$, whereas our PG-SGD converges at a rate of $\mathcal{O}\left(\frac{\log^2(\sqrt{T}/(1-\gamma))}{(1-\gamma)^2\sqrt{T}}\right)$, which is much faster since γ is usually close to 1, and $\log T$ is considered to be less influential in practice.

Convergence of TD-AMSGrad under Markovian Sampling

In this section, we adopt AMSGrad to TD learning and analyze its convergence under Markovian sampling. The proof techniques of bounding the bias and the nature of the convergence are very different from those of PG-AMSGrad.

TD Learning and TD-AMSGrad

Policy evaluation is a fundamental task in RL, and often plays a critical role in other algorithms such as PG that we study before. The goal of policy evaluation is to obtain an accurate estimation of the accumulated reward function known as the value function $V: \mathcal{S} \mapsto \mathbb{R}$ for a given policy π defined as

$$V^\pi(s) = \mathbb{E}\left[\sum_{t=1}^{\infty}\gamma^t R(s_t,a_t)\Big|s_1=s\right].$$

Under the function approximation, the value function $V(s)$ is parameterized by $\theta \in \mathbb{R}^d$ and denoted by $V(s;\theta)$. As many recent finite-time analysis of TD (Bhandari, Russo, and Singal 2018; Xu, Zou, and Liang 2019; Srikant and Ying 2019), we consider the linear approximation class of the value function $V(s;\theta)$ defined as

$$V(s;\theta) = \phi(s)^T\theta, \tag{12}$$

where $\theta \in \mathbb{R}^d$, and $\phi: \mathcal{S} \to \mathbb{R}^d$ is a vector function with the dimension d, and the elements of ϕ represent the nonlinear kernel (feature) functions. Then the vanilla TD algorithm follows a stochastic iterative method given by

$$\theta_{t+1} = \theta_t - \alpha_t g_t, \tag{13}$$

where α_t is the stepsize, and g_t is defined as

$$\begin{aligned} g_t &:= g(\theta_t; s_t, a_t, s_{t+1}) \\ &= \left(\phi^T(s_t)\theta_t - R(s_t,a_t) - \gamma\phi^T(s_{t+1})\theta_t\right)\phi(s_t). \end{aligned} \tag{14}$$

Here, g_t serves as a stochastic pseudo-gradient, and is an estimator of the full pseudo-gradient given by

$$\bar{g}(\theta_t) = \mathop{\mathbb{E}}_{\nu}\left[\left(\phi^T(s_t)\theta_t - R(s_t,\pi(s_t)) - \gamma\phi^T(s_{t+1})\theta_t\right)\phi(s_t)\right] \tag{15}$$

where the expectation is taken over the stationary distribution of the states. We note that $\bar{g}(\theta_t)$ is not a gradient of a loss function, but plays a similar role as the gradient in the gradient descent algorithm.

Then TD-AMSGrad is obtained by replacing the update (13) by the AMSGrad update given in (2)-(4) as in Algorithm 2.

As seen in Algorithm 2, the state-action pairs are sampled as a trajectory under the transition probability $\mathbb{P}$ with unknown policy π. Therefore, the samples along the trajectory are dependent, and hence we need to analyze the convergence of TD-AMSGrad under Markovian sampling.

Algorithm 2 TD-AMSGrad

Input: $\alpha, \lambda, \theta_1, \beta_1, \beta_2, m_0 = 0, \hat{v}_0 = 0, s_1 \sim \zeta(\cdot)$.
for $t = 1, 2, \ldots, T$ **do**
  Assign α_t, $\beta_{1t} = \beta_1 \lambda^t$.
  Sample $a_t \sim \pi, s_{t+1} \sim \mathbb{P}(\cdot|s_t, a_t)$.
  Compute g_t as (14).
  $m_t = (1-\beta_{1t})m_{t-1} + \beta_{1t} g_t$.
  $v_t = (1-\beta_2)\hat{v}_{t-1} + \beta_2 g_t^2$.
  $\hat{v}_t = \max(\hat{v}_{t-1}, v_t)$, $\hat{V}_t = diag(\hat{v}_{t,1}, \ldots, \hat{v}_{t,d})$.
  $\theta_{t+1} = \Pi_{\mathcal{D}, \hat{V}_t^{1/4}} \left(\theta_t - \alpha_t \hat{V}_t^{-\frac{1}{2}} m_t\right)$,
  where $\Pi_{\mathcal{D}, \hat{V}_t^{1/4}}(\theta') = \min_{\theta \in \mathcal{D}} \left\| \hat{V}_t^{1/4} (\theta' - \theta) \right\|$.
end for
Output: $\frac{1}{T}\sum_{t=1}^{T} \theta_t$.

Technical Assumptions

In this section, we introduce some standard technical assumptions for our analysis.

We first give the following standard assumption on the kernel function in the linear function approximation (Tsitsiklis and Van Roy 1997; Bhandari, Russo, and Singal 2018; Xu, Zou, and Liang 2019; Chen et al. 2019b).

Assumption 3. *For any state $s \in \mathcal{S}$, the kernel function $\phi : \mathcal{S} \to \mathbb{R}^d$ is uniformly bounded, i.e., $\|\phi(s)\| \leq 1, \ \forall s \in \mathcal{S}$. In addition, we define a feature matrix Φ as*

$$\Phi = \begin{bmatrix} \phi^T(s_1) \\ \phi^T(s_2) \\ \vdots \end{bmatrix} = \begin{bmatrix} \phi_1(s_1) & \cdots & \phi_d(s_1) \\ \phi_1(s_2) & \cdots & \phi_d(s_2) \\ \vdots & \vdots & \vdots \end{bmatrix},$$

and assume that the columns of Φ are linearly independent.

The boundedness assumption is mild since we can always normalize the kernel functions.

The next assumption of the bounded domain is standard in theoretical analysis of the Adam-type algorithms (Reddi, Kale, and Kumar 2018; Tran and Phong 2019), where the boundedness parameter D_∞ can be chosen as discussed in Bhandari, Russo, and Singal (2018).

Assumption 4. *The domain $\mathcal{D} \subset \mathbb{R}^d$ of approximation parameters is a ball originating at $\theta = 0$ with a bounded diameter containing $\theta^\star$. That is, there exists D_∞, such that $\theta^\star \in \mathcal{D}$, and $\|\theta_m - \theta_n\| < D_\infty$, for any $\theta_m, \theta_n \in \mathcal{D}$.*

Convergence of TD-AMSGrad

In the following, we provide the convergence results for TD-AMSGrad with linear function approximation under Markovian sampling.

First consider the full pseudo-gradient $\bar{g}(\theta)$ in (15). We define θ^* as the fixed point of $\bar{g}(\theta)$, i.e., $\bar{g}(\theta^*) = 0$. Then θ^* is the unique fixed point under Assumption 3 following from the contraction property of the projected Bellman operator (Tsitsiklis and Van Roy 1997).

The following theorem provides the convergence of TD-AMSGrad under a constant stepsize coupled with diminishing hyper-parameters in the AMSGrad update.

Theorem 3. *(Convergence of TD-AMSGrad with constant stepsize) Let $\beta_{1t} = \beta_1 \lambda^t$ and $\delta = \beta_1/\beta_2$ with $\delta, \lambda \in (0,1)$ in Algorithm 2. Initialize Algorithm 2 such that $|g_{1,i}| \geq G_0$ for all $i \in [d]$ with some $G_0 > 0$. Let $\alpha_t = \alpha, t = 1, \ldots, T$, and suppose Assumptions 2-4 hold. Then the output of Algorithm 2 satisfies:*

$$\mathbb{E}\,\|\theta_{out} - \theta^\star\|^2 \leq \frac{C_1}{T} + \alpha C_2,$$

where $C_1 = \frac{G_\infty D_\infty^2}{\alpha c(1-\beta)} + \frac{\beta_1 \lambda G_\infty D_\infty^2}{2\alpha c(1-\lambda)(1-\beta)} + 2((1+\gamma)D_\infty + G_\infty) \cdot \frac{G_\infty}{cG_0(1-\beta)}(\tau^*)^2\alpha$, $C_2 = \frac{4D_\infty G_\infty}{c(1-\beta)} + \frac{2G_\infty \tau^*((1+\gamma)D_\infty + G_\infty)}{cG_0(1-\beta)} + \frac{(1+\beta)G_\infty^2}{2cG_0(1-\beta)}$ *with* $c = (1-\gamma)\sqrt{\omega}$, $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha\}$ *and* $G_\infty = R_{\max} + (1+\gamma)D_\infty$.

In Theorem 3, C_1, C_2 are constants and time-independent. Therefore, under the choice of the stepsize and hyper-parameters in the theorem, Algorithm 2 converges to a neighborhood of the global optimum at a rate of $\mathcal{O}\left(\frac{1}{T}\right)$. The size of the neigborhood is controlled by the stepsize α. We can balance the tradeoff between the convergence rate and the convergence accuracy by setting the stepsize $\alpha_t = \frac{1}{\sqrt{T}}$, which yields a convergence to the global optimal solution at the rate of $\mathcal{O}\left(\frac{\log^2 T}{\sqrt{T}}\right)$.

Next, we provide the convergence result with a diminishing stepsize in the following theorem.

Theorem 4. *(Convergence of TD-AMSGrad with diminishing stepsize) Suppose the same conditions of Theorem 3 hold, and let $\alpha_t = \frac{\alpha}{\sqrt{t}}$ for $t = 1, \ldots, T$. Then the output of Algorithm 2 satisfies:*

$$\mathbb{E}\,\|\theta_{out} - \theta^\star\|^2 \leq \frac{C_1}{\sqrt{T}} + \frac{C_2}{T},$$

where $C_1 = \frac{G_\infty D_\infty^2}{2c\alpha(1-\beta)} + \frac{\alpha(1+\beta_1)G_\infty^2}{cG_0(1-\beta)} + \frac{4\alpha D_\infty G_\infty}{c(1-\beta)} + \frac{4\tau^*\alpha G_\infty((1+\gamma)D_\infty + G_\infty)}{cG_0(1-\beta)}$, $C_2 = \frac{G_\infty D_\infty^2}{\sqrt{2}c\alpha(1-\beta)} + \frac{\beta G_\infty D_\infty^2}{2c\alpha(1-\lambda)^2(1-\beta)} + \frac{2G_\infty \alpha(\tau^*)^2((1+\gamma)D_\infty + G_\infty)}{cG_0(1-\beta)}$ *with* $c = (1-\gamma)\sqrt{\omega}$, $\tau^* = \min\{\tau : m\rho^\tau \leq \alpha_T = \frac{\alpha}{\sqrt{T}}\}$ *and* $G_\infty = R_{\max} + (1+\gamma)D_\infty$.

Comparing with Theorem 3 and observing $\tau^* = \mathcal{O}(\log T)$, we conclude that TD-ASMGrad with the diminishing stepsize converges exactly to the global optimum at the rate of $\mathcal{O}\left(\frac{\log T}{\sqrt{T}}\right)$, rather than to a neighborhood.

Conclusion

This paper provides the first convergence analysis of the Adam-type RL algorithms under Markovian sampling. Several future directions along this topic are interesting. For example, the optimality of the convergence result of PG-AMSGrad is of importance to study. The analysis of PG-SGD can be also extended to actor-critic algorithms with Markovian sampling. The convergence of TD-AMSGrad with more general value function approximation is also of interest to study. We expect that the new analysis techniques that we develop here will be useful for further exploring the theoretical guarantee of other RL algorithms that incorporate the Adam-type updates.

Acknowledgements

The work was supported in part by the U.S. National Science Foundation under Grants CCF-1761506 and CCF-1900145, National Natural Science Foundation of China (Grant No. 62073159), and the Shenzhen Science and Technology Program (Grant No. JCYJ20200109141601708).

References

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

Baxter, J.; and Bartlett, P. L. 2001. Infinite-horizon policy-gradient estimation. *Journal of Artificial Intelligence Research* 15: 319–350.

Bello, I.; Zoph, B.; Vasudevan, V.; and Le, Q. V. 2017. Neural optimizer search with reinforcement learning. In *International Conference on Machine Learning (ICML)*, 459–468.

Benveniste, A.; Métivier, M.; and Priouret, P. 2012. *Adaptive algorithms and stochastic approximations*, volume 22. Springer Science & Business Media.

Bhandari, J.; and Russo, D. 2019. Global optimality guarantees for policy gradient methods. *arXiv preprint arXiv:1906.01786* .

Bhandari, J.; Russo, D.; and Singal, R. 2018. A Finite Time Analysis of Temporal Difference Learning With Linear Function Approximation. In *Conference on Learning Theory (COLT)*.

Cai, Q.; Yang, Z.; Lee, J. D.; and Wang, Z. 2019. Neural temporal-difference learning converges to global optima. In *Advances in Neural Information Processing Systems (NeurIPS)*, 11312–11322.

Castillo, G. A.; Weng, B.; Hereid, A.; Wang, Z.; and Zhang, W. 2019. Reinforcement learning meets hybrid zero dynamics: A case study for rabbit. In *2019 International Conference on Robotics and Automation (ICRA)*, 284–290.

Chen, X.; Liu, S.; Sun, R.; and Hong, M. 2019a. On the Convergence of A Class of Adam-Type Algorithms for Non-Convex Optimization. In *International Conference on Learning Representations (ICLR)*.

Chen, Z.; Zhang, S.; Doan, T. T.; Maguluri, S. T.; and Clarke, J.-P. 2019b. Finite-Time Analysis of Q-Learning with Linear Function Approximation. *arXiv preprint arXiv:1905.11425* .

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

Fazel, M.; Ge, R.; Kakade, S.; and Mesbahi, M. 2018. Global Convergence of Policy Gradient Methods for the Linear Quadratic Regulator. In *International Conference on Machine Learning (ICML)*, 1467–1476.

Györfi, L.; and Walk, H. 1996. On the averaged stochastic approximation for linear regression. *SIAM Journal on Control and Optimization* 34(1): 31–61.

Hu, B.; and Syed, U. 2019. Characterizing the exact behaviors of temporal difference learning algorithms using Markov jump linear system theory. In *Advances in Neural Information Processing Systems (NeurIPS)*, 8477–8488.

Ji, K.; Wang, Z.; Wo, B.; Zhou, Y.; Zhang, W.; and Liang, Y. 2020. Faster Stochastic Algorithms via History-Gradient Aided Batch Size Adaptation. to appear in *Proc. International Conference on Machine Learning (ICML),* also available as arXiv preprint *arXiv:1910.09670* .

Kakade, S. M. 2002. A natural policy gradient. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1531–1538.

Karimi, B.; Miasojedow, B.; Moulines, E.; and Wai, H.-T. 2019. Non-asymptotic Analysis of Biased Stochastic Approximation Scheme. In *Conference on Learning Theory (COLT)*.

Kingma, D. P.; and Ba, J. 2015. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*.

Konda, V. 2002. *Actor-critic algorithms*. Ph.D. thesis, Massachusetts Institute of Technology.

Konda, V. R.; and Tsitsiklis, J. N. 2000. Actor-critic algorithms. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1008–1014.

Kushner, H.; and Yin, G. G. 2003. *Stochastic approximation and recursive algorithms and applications*, volume 35. Springer Science & Business Media.

Lakshminarayanan, C.; and Szepesvari, C. 2018. Linear Stochastic Approximation: How Far Does Constant Step-Size and Iterate Averaging Go? In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 1347–1355.

Liu, B.; Cai, Q.; Yang, Z.; and Wang, Z. 2019. Neural Trust Region/Proximal Policy Optimization Attains Globally Optimal Policy. In *Advances in Neural Information Processing Systems (NeurIPS)*, 10564–10575.

Malik, D.; Pananjady, A.; Bhatia, K.; Khamaru, K.; Bartlett, P.; and Wainwright, M. 2019. Derivative-Free Methods for Policy Optimization: Guarantees for Linear Quadratic Systems. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2916–2925.

Mitrophanov, A. Y. 2005. Sensitivity and convergence of uniformly ergodic Markov chains. *Journal of Applied Probability* 42(4): 1003–1014.

Mnih, V.; Kavukcuoglu, K.; Silver, D.; Graves, A.; Antonoglou, I.; Wierstra, D.; and Riedmiller, M. 2013. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602* .

Papini, M.; Binaghi, D.; Canonaco, G.; Pirotta, M.; and Restelli, M. 2018. Stochastic Variance-Reduced Policy Gradient. In *International Conference on Machine Learning (ICML)*, 4026–4035.

Papini, M.; Pirotta, M.; and Restelli, M. 2017. Adaptive batch size for safe policy gradients. In *Advances in Neural Information Processing Systems (NeurIPS)*, 3591–3600.

Papini, M.; Pirotta, M.; and Restelli, M. 2019. Smoothing policies and safe policy gradients. *arXiv preprint arXiv:1905.03231* .

Pednault, E.; Abe, N.; and Zadrozny, B. 2002. Sequential cost-sensitive decision making with reinforcement learning. In *Proceedings of the eighth ACM SIGKDD International conference on Knowledge Discovery and Data Mining*, 259–268.

Pirotta, M.; Restelli, M.; and Bascetta, L. 2015. Policy gradient in lipschitz Markov decision processes. *Machine Learning* 100(2-3): 255–283.

Reddi, S. J.; Kale, S.; and Kumar, S. 2018. On the Convergence of Adam and Beyond. In *International Conference on Learning Representations (ICLR)*.

Rummery, G. A.; and Niranjan, M. 1994. *On-line Q-learning using connectionist systems*, volume 37.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; and Moritz, P. 2015. Trust region policy optimization. In *International Conference on Machine Learning (ICML)*, 1889–1897.

Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; and Klimov, O. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347* .

Shani, L.; Efroni, Y.; and Mannor, S. 2020. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Thirty-Fourth AAAI Conference on Artificial Intelligence*.

Shen, Z.; Ribeiro, A.; Hassani, H.; Qian, H.; and Mi, C. 2019. Hessian Aided Policy Gradient. In *International Conference on Machine Learning (ICML)*, 5729–5738.

Silver, D.; Lever, G.; Heess, N.; Degris, T.; Wierstra, D.; and Riedmiller, M. 2014. Deterministic Policy Gradient Algorithms. In *International Conference on Machine Learning (ICML)*, 387–395.

Srikant, R.; and Ying, L. 2019. Finite-Time Error Bounds For Linear Stochastic Approximation andTD Learning. In *Conference on Learning Theory (COLT)*, 2803–2830.

Stooke, A.; and Abbeel, P. 2018. Accelerated methods for deep reinforcement learning. *arXiv preprint arXiv:1803.02811* .

Sun, T.; Shen, H.; Chen, T.; and Li, D. 2020. daptive Temporal Difference Learning with Linear Function Approximation. *arXiv preprint arXiv:2002.08537* .

Sutton, R. S. 1988. Learning to predict by the methods of temporal differences. *Machine Learning* 3(1): 9–44.

Sutton, R. S.; McAllester, D. A.; Singh, S. P.; and Mansour, Y. 2000. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1057–1063.

Tadić, V. B.; Doucet, A.; et al. 2017. Asymptotic bias of stochastic gradient search. *The Annals of Applied Probability* 27(6): 3255–3304.

Tran, P. T.; and Phong, L. T. 2019. On the convergence proof of AMSGrad and a new version. *IEEE Access* 7: 61706–61716.

Tsitsiklis, J. N.; and Van Roy, B. 1997. An Analysis of Temporal-Diffference Learning with Function Approximation. *IEEE Transactions on Automatic Control* 42(5): 674 – 690.

Tu, S.; and Recht, B. 2019. The Gap Between Model-Based and Model-Free Methods on the Linear Quadratic Regulator: An Asymptotic Viewpoint. In *Conference on Learning Theory (COLT)*, 3036–3083.

Wang, L.; Cai, Q.; Yang, Z.; and Wang, Z. 2020. Neural policy gradient methods: Global optimality and rates of convergence. In *International Conference on Learning Representations (ICLR)*.

Watkins, C. J.; and Dayan, P. 1992. Q-learning. *Machine Learning* 8(3-4): 279–292.

Weng, B.; Xiong, H.; Liang, Y.; and Zhang, W. 2020. Analysis of Q-learning with Adaptation and Momentum Restart for Gradient Descent. In *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence (IJCAI-20)*, 3051–3057.

Williams, R. J. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine Learning* 8(3-4): 229–256.

Xu, P.; Gao, F.; and Gu, Q. 2019. An Improved Convergence Analysis of Stochastic Variance-Reduced Policy Gradient. In *International Conference on Uncertainty in Artificial Intelligence (UAI)*.

Xu, P.; Gao, F.; and Gu, Q. 2020. Sample Efficient Policy Gradient Methods with Recursive Variance Reduction. In *International Conference on Learning Representations (ICLR)*.

Xu, T.; Wang, Z.; and Liang, Y. 2020a. Improving Sample Complexity Bounds for Actor-Critic Algorithms. *arXiv preprint arXiv:2004.12956* .

Xu, T.; Wang, Z.; and Liang, Y. 2020b. Non-asymptotic Convergence Analysis of Two Time-scale (Natural) Actor-Critic Algorithms. *arXiv preprint arXiv:2005.03557* .

Xu, T.; Zou, S.; and Liang, Y. 2019. Two time-scale off-policy TD learning: Non-asymptotic analysis over Markovian samples. In *Advances in Neural Information Processing Systems (NeurIPS)*, 10633–10643.

Zhang, K.; Koppel, A.; Zhu, H.; and Başar, T. 2019. Global Convergence of Policy Gradient Methods to (Almost) Locally Optimal Policies. *arXiv preprint arXiv:1906.08383* .

Zhou, D.; Tang, Y.; Yang, Z.; Cao, Y.; and Gu, Q. 2018. On the convergence of adaptive gradient methods for nonconvex optimization. *arXiv preprint arXiv:1808.05671* .

Zou, F.; Shen, L.; Jie, Z.; Zhang, W.; and Liu, W. 2019. A sufficient condition for convergences of adam and rmsprop. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 11127–11135.

Zou, S.; Xu, T.; and Liang, Y. 2019. Finite-sample analysis for sarsa with linear function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 8665–8675.