Wasserstein Flow Meets Replicator Dynamics: A Mean-Field Analysis of Representation Learning in Actor-Critic

Anonymous Author(s)
Affiliation
Address
email

Abstract

Actor-critic (AC) algorithms, empowered by neural networks, have had significant empirical success in recent years. However, most of the existing theoretical support for AC algorithms focuses on the case of linear function approximations, or linearized neural networks, where the feature representation is fixed throughout training. Such a limitation fails to capture the key aspect of representation learning in neural AC, which is pivotal in practical problems. In this work, we take a mean-field perspective on the evolution and convergence of feature-based neural AC. Specifically, we consider a version of AC where the actor and critic are represented by overparameterized two-layer neural networks and are updated with two-timescale learning rates. The critic is updated by temporal-difference (TD) learning with a larger stepsize while the actor is updated via proximal policy optimization (PPO) with a smaller stepsize. In the continuous-time and infinite-width limiting regime, when the timescales are properly separated, we prove that neural AC finds the globally optimal policy at a sublinear rate. Additionally, we prove that the feature representation induced by the critic network is allowed to evolve within a neighborhood of the initial one.

1 Introduction

In reinforcement learning (RL) [56], an agent aims to learn the optimal policy that maximizes the expected total reward by interacting with the environment. Policy-based RL algorithms achieve such a goal by directly optimizing the expected total reward as a function of the policy, which often involves two components: policy evaluation and policy improvement. Specifically, policy evaluation refers to estimating the value function of the current policy, which characterizes the performance of the current policy and reveals the updating direction for finding a better policy, which is known as policy improvement. Algorithms with these two ingredients are also called actor-critic (AC) methods [36], where the actor and the critic refer to the policy and its corresponding value function, respectively.

Recently, in RL applications with large state spaces, actor-critic empowered by expressive function approximators such as neural networks have achieved striking empirical successes [3, 4, 9, 20, 51, 52, 60]. These successes benefit from the data-dependent representations learned by neural networks. Unfortunately, however, the theoretical understanding of this data-dependent benefit is very limited. The classical theory of AC focuses on the case of linear function approximation, where the actor and critic are represented using linear functions with the feature mapping fixed throughout learning [10, 11, 36]. Meanwhile, a few recent works establish convergence and optimality of AC with overparameterized neural networks [26, 39, 61], where the neural network training is captured by the Neural Tangent Kernel (NTK) [30]. Specifically, with properly designed parameter initialization and stepsizes, and sufficiently large network widths, the neural networks employed by both actor and critic can be assumed to be well approximated by linear functions of a random feature determined by initial parameters. In other words, concerning representation learning, the features induced by

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

these algorithms are by assumption infinitesimally close to the initial featural representation, which is data-independent.

In this work, we make initial steps towards understanding how representation learning comes into play in neural AC. Specifically, we address the following questions:

Going beyond the NTK regime, does neural AC provably find the globally optimal policy? How does the feature representation associated with the neural network evolve along with neural AC?

We focus on a version of AC where the critic performs temporal difference (TD) learning [55] for policy evaluation and the actor improves its policy via proximal policy optimization (PPO) [49], which corresponds to a Kullback-Leibler (KL) divergence regularized optimization problem, with the critic providing the update direction. Moreover, we utilize two-timescale updates where both the actor and critic are updated at each iteration but with the critic having a much larger stepsize. In other words, the critic is updated at a faster timescale. Meanwhile, we represent the critic explicitly as a two-layer overparameterized neural network and parameterize the actor implicitly via the critic and PPO updates. To examine convergence, we study the evolution of the actor and critic in the continuous-time limiting regime with the network width going to infinity. In such a regime, the actor update is closely connected to replicator dynamics [12, 28, 50] and the critic update is captured by a semigradient flow in the Wasserstein space [59]. Moreover, the semigradient flow runs at a faster timescale according to the two-timescale mechanism.

It turns out that the separation of timescales plays an important role in the convergence analysis. In particular, in the continuous-time limit, it enables us to first separately analyze the evolution of actor and critic and then combine these results to get final theoretical guarantees. Specifically, focusing solely on the actor, we prove that the time-averaged suboptimality of the actor converges sublinearly to zero up to the time-averaged policy evaluation error associated with critic updates. Moreover, for the critic, under proper regularity conditions, we connect the Bellman error to the Wasserstein distance and show that the time-averaged policy evaluation error also converges sublinearly to zero. Therefore, we show that neural AC provably achieves global optimality at a sublinear rate. Furthermore, regarding representation learning, we show that the critic induces a data-dependent feature representation within an $O(1/\alpha)$ neighborhood of the initial representation in terms of the Wasserstein distance, where α is a sufficiently large scaling parameter.

The key to our technical analysis reposes on three ingredients: (i) infinite-dimensional variational inequalities with a one-point monotonicity [27], (ii) a mean-field perspective on neural networks [19, 41, 42, 53, 54], and (iii) the two-timescale stochastic approximation [13, 37]. In particular, in the infinite-width limit, the neural network and its induced feature representation are identified with a distribution over the parameter space. The mean-field perspective enables us to characterize the evolution of such a distribution within the Wasserstein space via a partial differential equation (PDE) [5, 6, 58, 59]. For policy evaluation, such a PDE is given by the semigradient flow induced by TD learning. We characterize the error of policy evaluation by showing that mean-field Bellman error satisfies a version of one-point monotonicity tailored to the Wasserstein space. Moreover, our actor analysis utilizes the geometry of policy optimization, which shows that the expected total reward, as a function of the policy, also enjoys the property of one-point monotonicity in the policy space. Finally, the actor and critic errors are connected via two-timescale stochastic approximation. To the best of our knowledge, this is the first time that convergence and global optimality guarantees have been obtained for neural AC.

Related Work. AC with linear function approximation has been studied extensively in the literature. In particular, using a two-timescale stochastic approximation via ordinary differential equations, [10, 11, 36] establish asymptotic convergence guarantees in the continuous-time limiting regime. More recently, using more sophisticated optimization techniques, various works [29, 35, 64–66] have established discrete-time convergence guarantees that show that linear AC converges sublinearly to either a stationary point or the globally optimal policy. Furthermore, when overparameterized neural networks are employed, [26, 39, 61] prove that neural AC converges to the global optimum at a sublinear rate. In these works, the initial value of the network parameters and the learning rates are chosen such that both actor and critic updates are captured by the NTK. In other words, when the network width is sufficiently large, such a version of neural AC is well approximated by its linear counterpart via the neural tangent feature. In comparison, we establish a mean-field analysis that has a different scaling than the NTK regime. We also establish finite-time convergence to global

optimality, and more importantly, the feature representation induced by the critic is data-dependent and allowed to evolve within a much larger neighborhood around the initial one.

Furthermore, our work is also related to the recent line of research on understanding stochastic gradient descent (SGD) for supervised learning problems involving an overparameterized two-layer neural network under the mean-field regime. See, e.g., [16, 19, 21, 22, 31, 40–42, 53, 54, 63] and the references therein. In the continuous-time and infinite-width limit, these works show that SGD for neural network training is captured by a Wasserstein gradient flow [5, 6, 59] of an energy function that corresponds to the objective function in supervised learning. In contrast, our analysis combines such a mean-field analysis with TD learning and two-timescale stochastic approximation, which are tailored specifically to AC. Moreover, our critic is updated via TD learning, which is a semigradient algorithm and there is no objective functional making TD learning a gradient-based algorithm. Thus, in the mean-field regime, our critic is given by a Wasserstein semigradient flow, which also differs from these existing works.

Additionally, our work is closely related to [1, 69], who provide mean-field analyses for neural TD-learning and Q-learning [62]. In comparison, we focus on AC, which is a two-timescale policy optimization algorithm. Finally, [2] studies softmax policy gradient with neural network policies in the mean-field regime, where policy gradient is cast as a Wasserstein gradient flow with respect to the expected total reward. The algorithm assumes that the critic directly gets the desired value function and thus the algorithm is single-timescale. Moreover, the convergence guarantee in [2] is asymptotic. In comparison, our AC is two-timescale and we establish non-asymptotic sublinear convergence guarantees to global optimality.

Notation. We denote by $\mathscr{P}(\mathcal{X})$ the set of probability measures over the measurable space $\mathcal{X}$. Given a curve $\rho : \mathbb{R} \to \mathcal{X}$, we denote by $\dot{\rho}_s = \partial_t \rho_t \,|_{\,t=s}$ its derivative with respect to the time. For an operator $F : \mathcal{X} \to \mathcal{X}$ and a measure $\mu \in \mathscr{P}(\mathcal{X})$, we denote by $F_\sharp \mu = \mu \circ F^{-1}$ the push forward of μ through F. We denote by $\chi^2(\rho \,\|\, \mu)$ the chi-squared divergence between probability measures ρ and μ, which is defined as $\chi^2(\rho \,\|\, \mu) = \int (\rho/\mu - 1)^2 \mathrm{d}\mu$. Given two probability measures ρ and μ, we denote the Kullback-Leibler divergence or the relative entropy from μ to ρ by $\mathrm{KL}(\rho \,\|\, \mu) = \int \log(\rho/\mu)\mathrm{d}\rho$. For $\nu_1, \nu_2, \mu \in \mathscr{P}(\mathcal{X})$, we define the $\dot{H}^{-1}(\mu)$ weighted homogeneous Sobolev norm as $\|\nu_1 - \nu_2\|_{\dot{H}^{-1}(\mu)} = \sup\big\{|\langle f, \nu_1 - \nu_2\rangle| \,\big|\, \|f\|_{\dot{H}^1(\mu)} \le 1\big\}$. We denote by $\|f(x)\|_{p,\mu} = (\int |f(x)|^p \mu(\mathrm{d}x))^{1/p}$ the ℓ_p-norm with respect to probability measure μ. We denote by $\otimes$ the Kronecker product. For a function $f : \mathcal{X} \to \mathbb{R}$, we denote by $\mathrm{Lip}(f) = \sup_{x,y\in\mathcal{X}, x\neq y} |f(x) - f(y)|/\|x - y\|$ its Lipschitz constant. We denote a normal distribution on $\mathbb{R}^D$ by $\mathcal{N}(\mu, \Sigma)$, where μ is the mean value and Σ is the covariance matrix.

2 Background

In this section, we first introduce the policy optimization problem and the actor-critic method. We then present the definition of the Wasserstein space.

2.1 Policy Optimization and Actor-Critic

We consider a Markov decision process (MDP) given by $(\mathcal{S}, \mathcal{A}, \gamma, P, r, \mathcal{D}_0)$, where $\mathcal{S} \subseteq \mathbb{R}^{d_1}$ is the state and $\mathcal{A} \subseteq \mathbb{R}^{d_2}$ is the action space, $\gamma \in (0, 1)$ is the discount factor, $P : \mathcal{S} \times \mathcal{A} \to \mathscr{P}(\mathcal{S})$ is the transition kernel, $r : \mathcal{S} \times \mathcal{A} \to \mathbb{R}_+$ is the reward function, and $\mathcal{D}_0 \in \mathscr{P}(\mathcal{S})$ is the initial state distribution. Without loss of generality, we assume $\mathcal{S} \times \mathcal{A} \subseteq \mathbb{R}^d$ and $\|(s, a)\|_2 \le 1$, where $d = d_1 + d_2$. Given a policy $\pi : \mathcal{S} \times \mathcal{A} \to \mathscr{P}(\mathcal{S})$, at the m-th step, the agent takes an action a_m at state s_m according to $\pi(\cdot \,|\, s_m)$ and observes a reward $r_m = r(s_m, a_m)$. The environment then transits to the next state s_{m+1} according to the transition kernel $P(\cdot \,|\, s_m, a_m)$. Note that the policy π induces a Markov chain on both $\mathcal{S}$ and $\mathcal{S} \times \mathcal{A}$. Considering the Markov chain on $\mathcal{S}$, we denote the induced Markov transition kernel by $P^\pi : \mathcal{S} \to \mathscr{P}(\mathcal{S})$, which is defined as $P^\pi(s' \,|\, s) = \int_{\mathcal{A}} P(s' \,|\, s, a)\pi(\mathrm{d}a \,|\, s)$. Likewise, we denote the Markov transition kernel on $\mathcal{S} \times \mathcal{A}$ by $\widetilde{P}^\pi : \mathcal{S} \times \mathcal{A} \to \mathscr{P}(\mathcal{S} \times \mathcal{A})$, which is defined as $\widetilde{P}^\pi(s', a' \,|\, s, a) = \pi(a' \,|\, s')P(s' \,|\, s, a)$. Let $\widetilde{\mathcal{D}}$ be a probability measure on $\mathcal{S} \times \mathcal{A}$. We then define

the visitation measure induced by policy π and starting from $\widetilde{\mathcal{D}}$ as

$$\widetilde{\mathcal{E}}_{\widetilde{\mathcal{D}}}^{\pi}\big(\mathrm{d}(s,a)\big) = (1-\gamma)\sum_{m\geq 0}\gamma^m \cdot \mathbb{P}\big((s_m,a_m)\in \mathrm{d}(s,a)\,|\,(s_0,a_0)\sim\widetilde{\mathcal{D}}\big), \tag{2.1}$$

where (s_m, a_m) is the trajectory generated by starting from $(s_0, a_0) \sim \widetilde{\mathcal{D}}$ and following policy π thereafter. If $\widetilde{\mathcal{D}} = \mathcal{D} \otimes \pi$ holds, we then denote such a visitation measure by $\widetilde{\mathcal{E}}_{\mathcal{D}}^{\pi}$. Furthermore, we denote by $\mathcal{E}(\mathrm{d}s) = \int_{\mathcal{A}} \widetilde{\mathcal{E}}(\mathrm{d}s, \mathrm{d}a)$ the marginal distribution of visitation measure $\widetilde{\mathcal{E}}$. When $(s_0, a_0) \sim \mathcal{D} \otimes \pi$ holds in (2.1), it follows that $\widetilde{\mathcal{E}}_{\mathcal{D}}^{\pi} = \mathcal{E}_{\mathcal{D}}^{\pi} \otimes \pi$. In policy optimization, we aim to maximize the expected total rewards $J(\pi)$ defined as follows,

$$J(\pi) = \mathbb{E}^{\pi}\bigg[\sum_{m\geq 0}\gamma^m \cdot r(s_m,a_m)\,\big|\,s_0\sim\mathcal{D}_0\bigg],$$

where we denote by $\mathbb{E}^{\pi}$ the expectation with respect to $a_m \sim \pi(\cdot\,|\,s_m)$ and $s_{m+1} \sim P(\cdot\,|\,s_m, a_m)$ for $m \geq 0$. We define the action value function $Q^{\pi} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ and the state value function $V^{\pi} : \mathcal{S} \to \mathbb{R}$ as follows,

$$Q^{\pi}(s,a) = \mathbb{E}^{\pi}\bigg[\sum_{m\geq 0}\gamma^m \cdot r(s_m,a_m)\,\big|\,s_0=s, a_0=a\bigg], \quad V^{\pi}(s) = \big\langle Q^{\pi}(s,\cdot), \pi(\cdot\,|\,s)\big\rangle_{\mathcal{A}}, \tag{2.2}$$

where we denote by $\langle\cdot,\cdot\rangle_{\mathcal{A}}$ the inner product on the action space $\mathcal{A}$. Correspondingly, the advantage function $A^{\pi} : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ is defined as

$$A^{\pi}(s,a) = Q^{\pi}(s,a) - V^{\pi}(s).$$

It is known that the action value function Q^{π} is the unique global minimizer to the following mean-squared Bellman error (MSBE),

$$\mathrm{MSBE}(Q;\pi) = \frac{1}{2}\mathbb{E}_{(s,a)\sim\widetilde{\Phi}^{\pi}}\Big[\big(Q(s,a) - r(s,a) - \gamma\mathbb{E}_{(s',a')\sim\widetilde{P}^{\pi}(\cdot\,|\,s,a)}[Q(s',a')]\big)^2\Big], \tag{2.3}$$

where $\widetilde{\Phi}^{\pi}$ is a weighting distribution depending on policy π and is with full support, i.e., $\mathrm{supp}(\widetilde{\Phi}^{\pi}) = \mathcal{S}\times\mathcal{A}$. Therefore, the policy optimization problem can be written as the following bilevel optimization problem,

$$\max_{\pi} J(\pi) = \mathbb{E}_{s\sim\mathcal{D}_0}\Big[\big\langle Q^{\pi}(s,\cdot), \pi(\cdot\,|\,s)\big\rangle_{\mathcal{A}}\Big], \quad \text{subject to } Q^{\pi} = \operatorname*{argmin}_{Q}\mathrm{MSBE}(Q;\pi). \tag{2.4}$$

The inner problem in (2.4) is known as a policy evaluation subproblem, while the outer problem is the policy improvement subproblem. One of the most popular way to solve the policy optimization problem is the actor-critic (AC) [56], where the job of critic is to evaluate current policy and then the actor uses this evaluation to update its policy.

2.2 Wasserstein Space

Let $\Theta \subseteq \mathbb{R}^D$ be a Polish space. We denote by $\mathscr{P}_2(\Theta) \subseteq \mathscr{P}(\Theta)$ the set of probability measures with finite second moments. Then, the Wasserstein-2 (W_2) distance between $\mu, \nu \in \mathscr{P}_2(\Theta)$ is defined as follows,

$$W_2(\mu,\nu) = \inf\Big\{\mathbb{E}\big[\|X-Y\|^2\big]^{1/2}\,\Big|\,\mathrm{law}(X)=\mu, \mathrm{law}(Y)=\nu\Big\},$$

where the infimum is taken over the random variables X and Y on Θ and we denote by $\mathrm{law}(X)$ the distribution of a random variable X. We call $\mathcal{M} = (\mathscr{P}_2(\Theta), W_2)$ the Wasserstein (W_2) space, which is an infinite-dimensional manifold [59]. See §A for more details.

3 Algorithm

Two-timescale Actor-critic. We consider the two-timescale Actor-critic (AC) [34, 45] for the policy optimization problem in (2.4). For policy evaluation, we parameter the critic Q with a neural network and update the parameter by temporal difference (TD) [55]. For policy improvement, we update the actor policy π via the proximal policy optimization (PPO) [49]. Our algorithm is two-timescale since both the actor and the critic are updated at each iteration with different stepsizes. Specifically, we parameterize the critic Q by the following neural network with width M and parameter $\widehat{\boldsymbol{\theta}} =$

$(\widehat{\theta}^{(1)}, \cdots, \widehat{\theta}^{(M)}) \in \mathbb{R}^{D\times M}$,

$$Q_{\widehat{\boldsymbol{\theta}}}(s,a) = \frac{\alpha}{M}\sum_{i=1}^{M}\sigma(s,a;\widehat{\theta}^{(i)}). \tag{3.1}$$

Here $\sigma(s,a;\theta) : \mathcal{S}\times\mathcal{A}\times\mathbb{R}^D \to \mathbb{R}$ is the activation function and $\alpha > 0$ is the scaling parameter. Such a structure also appears in [17, 18, 41]. In a descrete-time finite-width (DF) scenario, at the k-th iteration, the the critic and the actor are updated as follows,

$$\text{DF-TD:}\quad \widehat{\theta}_{k+1}^{(i)} = \widehat{\theta}_k^{(i)} - \frac{\varepsilon'}{\alpha}\big(Q_{\widehat{\boldsymbol{\theta}}_k}(s_k,a_k) - r(s_k,a_k) - \gamma Q_{\widehat{\boldsymbol{\theta}}_k}(s_k',a_k')\big)\nabla_\theta\sigma(s,a;\widehat{\theta}_k^{(i)}), \tag{3.2}$$

$$\text{DF-PPO:}\quad \widehat{\pi}_{k+1}(\cdot\,|\,s) = \operatorname*{argmax}_{\pi}\Big\{\big\langle Q_{\widehat{\boldsymbol{\theta}}_k}(s,\cdot),\pi(\cdot\,|\,s)\big\rangle_{\mathcal{A}} - \varepsilon^{-1}\cdot\mathrm{KL}\big(\pi(\cdot\,|\,s)\,\|\,\widehat{\pi}_k(\cdot\,|\,s)\big)\Big\}, \tag{3.3}$$

where $(s_k,a_k)\sim\widetilde{\Phi}^{\widehat{\pi}_k}$ and $(s_k',a_k')\sim\widetilde{P}^{\widehat{\pi}_k}(\cdot\,|\,s_k,a_k)$. Here $\widehat{\pi}_k$ is the policy for the actor at the k-th iteration, $\widetilde{\Phi}^{\widehat{\pi}_k}$ is the corresponding weighting distribution, ε and ε' are the stepsizes for the DF-PPO update and the DF-TD update, respectively. In (3.2), the scaling of α^{-1} arises since our update falls into the lazy-training regime [18]. In the sequel, we denote by $\eta = \varepsilon'/\varepsilon$ the relative TD timescale. Note that in a double-loop AC algorithm, the critic can usually be solved with high precision. However, in the two-timescale AC, the critic $Q_{\widehat{\boldsymbol{\theta}}_k}$ for updating the actor $\widehat{\pi}_k$ can be far from the true action value function $Q^{\widehat{\pi}_k}$. Since the policy evaluation problem in two-time AC is not fully solved at each iteration, the two-timescale AC can be more efficient in computation while more challenging to establish a theoretical guarantee.

Mean-field (MF) Limit. To analyze the convergence of the two-timescale AC with neural networks, we employ the analysis which studies the mean-field limit regime [41, 42]. Here, by saying mean-field limit, we refer to the infinite-width limit, i.e., $M\to\infty$ in (3.1), and the continuous-time limit, i.e., $t = k\varepsilon$ where $\varepsilon\to 0$ in (3.2) and (3.3). For $\widehat{\boldsymbol{\theta}} = \{\widehat{\theta}^{(i)}\}_{i=1}^M$ independently sampled from a distribution ρ, we can write the infinite-width limit of (3.1) as

$$Q(s,a;\rho) = \alpha\int\sigma(s,a;\theta)\rho(\mathrm{d}\theta). \tag{3.4}$$

In the sequel, we denote by $\widehat{\rho}_k$ the distribution of $\widehat{\theta}_k^{(i)}$ for the infinite-width limit of the neural network at the k-th iteration. We further let ρ_t and π_t be the continuous-time limits of $\widehat{\rho}_k$ and $\widehat{\pi}_k$, respectively. As studied in [69], the mean-field limit of the DF-TD update in (3.2) is

$$\text{MF-TD:}\quad \partial_t\rho_t = -\eta\,\mathrm{div}\big(\rho_t\cdot g(\cdot;\rho_t,\pi_t)\big), \tag{3.5}$$

where η is the relative TD timescale and

$$g(\theta;\rho,\pi) = -\mathbb{E}_{\widetilde{\Phi}^\pi}^{\pi}\big[(Q(s,a;\rho) - r(s,a) - \gamma\cdot Q(s',a';\rho))\cdot\alpha^{-1}\nabla_\theta\sigma(s,a;\theta)\big] \tag{3.6}$$

is a vector field. Here $\mathbb{E}_{\widetilde{\Phi}^\pi}^{\pi}$ is taken with respect to $(s,a)\sim\widetilde{\Phi}^\pi$ and $(s',a')\sim\widetilde{P}^\pi(\cdot\,|\,s,a)$. It remains to characterize the mean-field limit of the DF-PPO update in (3.3). By solving the maximization problem in (3.3), the infinite-width limit of DF-PPO update can be written in closed form as

$$\varepsilon^{-1}\cdot\big(\log(\widehat{\pi}_{k+1}(a\,|\,s)) - \log(\widehat{\pi}_k(a\,|\,s))\big) = Q(s,a;\widehat{\rho}_k) - \widehat{Z}_k(s),$$

where $\widehat{Z}_k(s)$ is the normalizing factor such that $\int\widehat{\pi}_k(\mathrm{d}a\,|\,s) = 1$ for any $s\in\mathcal{S}$. By letting $t = k\varepsilon$ and $\varepsilon\to 0$, we have $\partial_t\log\pi_t = Q_t - Z_t$, which can be further written as $\partial_t\pi_t = \pi_t\cdot(Q_t - Z_t)$. Here we have $Q_t(a,s) = Q(a,s;\rho_t)$ and Z_t is the continuous-time limit of $\widehat{Z}_k$. Furthermore, noting that $\partial_t\int\pi_t(\mathrm{d}a\,|\,s) = 0$, the mean-field limit of the DF-PPO update in (3.3) is

$$\text{MF-PPO:}\quad \frac{\mathrm{d}}{\mathrm{d}t}\pi_t = \pi_t\cdot A_t,\qquad \text{where } A_t(s,a) = Q_t(s,a) - \int Q_t(s,a)\pi_t(\mathrm{d}a\,|\,s). \tag{3.7}$$

The two updates (3.5) and (3.7) correspond to the mean-field limits of (3.2) and (3.3), respectively, and together serve as the mean-field limit of the two-timescale AC. In particular, we remark that the MF-TD update in (3.5) for the critic is captured by a semigradient flow in the Wasserstein space [59] while the MF-PPO update in (3.7) for the actor resembles the replicator dynamics [12, 28, 50]. Note that such a framework is applicable to continuous state and action space. In this paper, we aim to provide a theoretical analysis of the mean-field limit of the two-timescale AC.

4 Main Result

In this section, we first establish the convergence of the MF-PPO update in §4.1. Then, with additional assumptions, we establish the optimality and convergence of the mean-field two-timescale AC in §4.2.

4.1 Convergence of Mean-field PPO

For the MF-PPO update in (3.7), we establish the following theorem on global optimality and convergence rate.

Theorem 4.1 (Convergence of MF-PPO)**.** Let $\pi^* = \operatorname{argmax}_\pi J(\pi)$ be the optimal policy and π_0 be the initial policy. Then, it holds that

$$\frac{1}{T}\int_0^T \big(J(\pi^*) - J(\pi_t)\big)\mathrm{d}t \le \frac{\zeta}{T} + 4\kappa \cdot \underbrace{\frac{1}{T}\int_0^T \|Q_t - Q^{\pi_t}\|_{2,\widetilde{\phi}^{\pi_t}}\mathrm{d}t}_{\text{policy evaluation error}}, \tag{4.1}$$

where $\widetilde{\phi}^{\pi_t} \in \mathscr{P}(\mathcal{S}\times\mathcal{A})$ is an evaluation distribution for the policy evaluation error and $\zeta = \mathbb{E}_{s\sim\mathcal{E}_{\mathcal{D}_0}^{\pi^*}}\Big[\mathrm{KL}\big(\pi^*(\cdot\,|\,s)\,\|\,\pi_0(\cdot\,|\,s)\big)\Big]$ is the expected KL-divergence between π^* and π_0. Furthermore, letting $\widetilde{\phi}^{\pi_t} = \frac{1}{2}\widetilde{\phi}_0 + \frac{1}{2}\phi_0\otimes\pi_t$, where $\widetilde{\phi}_0 \in \mathscr{P}(\mathcal{S}\times\mathcal{A})$ is a base distribution and $\phi_0 = \int_{\mathcal{A}}\widetilde{\phi}_0(\cdot,\mathrm{d}a)$, the concentrability coefficient κ is then given by

$$\kappa = \left\|\frac{\widetilde{\mathcal{E}}_{\mathcal{D}_0}^{\pi^*}}{\widetilde{\phi}_0}\right\|_\infty.$$

Proof. See §B.1 for a detailed proof. □

The concentrability coefficient commonly appears in the reinforcement learning literature [7, 23, 24, 38, 39, 43, 48, 57, 61]. In contrast to a more standard concentrability coefficient form, note that κ is irrelevant to the update of the algorithm. To show the convergence of mean-field PPO, our condition here is much weaker since we only need a given base distribution $\widetilde{\phi}_0$ such that $\kappa < \infty$.

Theorem 4.1 shows that the MF-PPO converges to the globally optimal policy at a rate of $\mathcal{O}(T^{-1})$ up to the policy evaluation error. Such a theorem implies the global optimality and convergence of a double-loop AC algorithm, where the critic Q_t is solved to high precision and the policy evaluation error is sufficiently small. In the sequel, we consider a more challenging setting, where the critic Q_t is updated simultaneously along with the update of the actor π_t.

4.2 Global Optimality and Convergence of Two-timescale AC

In what follows, we aim to characterize the upper bound of the policy evaluation error when the critic and the actor are updated simultaneously. Specifically, the actor is updated via MF-PPO in (3.7) and the critic $Q_t = Q(\cdot;\rho_t)$ is updated via the MF-TD in (3.5). For the smooth function σ in the parameterization of the Q function in (3.4), we consider it to be the following two-layer neural network,

$$\sigma(s,a;\theta) = B_\beta\cdot\beta(b)\cdot\widetilde{\sigma}\big(w^\top(s,a,1)\big), \tag{4.2}$$

where $\widetilde{\sigma}:\mathbb{R}\to\mathbb{R}$ is the activation function, $\theta = (b,w)$ is the parameter, and $\beta:\mathbb{R}\to(-1,1)$ is an odd and invertible function with scaling hyper-parameter $B_\beta > 0$. It then holds that $D = d+2$, where d and D are the dimensions of (s,a) and θ, respectively. It is worth noting that the function class of $\int\sigma(s,a;\theta)\rho(\mathrm{d}\theta)$ for $\rho\in\mathscr{P}_2(\mathbb{R}^D)$ is the same as

$$\mathcal{F} = \Big\{\int\beta'\cdot\widetilde{\sigma}\big(w^\top(s,a,1)\big)\nu(\mathrm{d}\beta',\mathrm{d}w)\,\Big|\,\nu\in\mathscr{P}_2\big((-B_\beta,B_\beta)\times\mathbb{R}^{d+1}\big)\Big\}, \tag{4.3}$$

which captures a vast function class because of the universal function approximation theorem [8, 47]. We remark that we introduce the rescaling function β in (4.2) to avoid the study of the space of probability measures over $(-B_\beta,B_\beta)\times\mathbb{R}^{d+1}$ in (4.3), which has boundary and thus lacks the

regularity in the study of optimal transport [59]. Furthermore, note that we introduce a hyper-parameter $\alpha > 1$ in the Q function in (3.4). Thus, we are using $\alpha \cdot \mathcal{F}$ to represent $\mathcal{F}$, which causes an "over-representation" when $\alpha > 1$. Such over-representation appears to be essential for our analysis. Furthermore, we consider the initialization of the TD update to be $\rho_0 = \mathcal{N}(0, I_D)$, which implies that $Q(s, a; \rho_0) = 0$. We further impose the following assumption on the two-layer neural network σ.

Assumption 4.2 (Regularity of the Neural Network)**.** For the two-layer neural network σ defined in (4.2), we assume that the following properties hold.

(i) The rescaling function $\beta : \mathbb{R} \to (-1, 1)$ is odd, $L_{0,\beta}$-Lipschitz continuous, $L_{1,\beta}$-smooth, and invertible. Meanwhile, the inverse β^{-1} is locally Lipschitz continuous. In particular, we assume that β^{-1} is ℓ_β-Lipschitz continuous in $[-2/3, 2/3]$.

(ii) The activation function $\widetilde{\sigma} : \mathbb{R} \to \mathbb{R}$ is odd, $B_{\widetilde{\sigma}}$-bounded, $L_{0,\widetilde{\sigma}}$-Lipschitz continuous, and $L_{1,\widetilde{\sigma}}$-smooth.

We remark that Assumption 4.2 is not restrictive and can be satisfied by a large family of neural networks, e.g., $\widetilde{\sigma}(x) = \tanh(x)$ and $\beta(b) = \tanh(b)$. Noting that $\|(s, a)\|_2 \leq 1$, Assumption 4.2 implies that the function $\sigma(s, a; \theta)$ in (4.2) is odd and is also bounded, Lipschitz continuous, and smooth in the parameter domain. Thus, we have

$$|\sigma(s, a; \theta)| < B_1, \quad |\nabla^2_{\theta\theta}\sigma(s, a; \theta)| < B_2. \tag{4.4}$$

We then impose the following assumption on the MDP.

Assumption 4.3 (Regularity of the MDP)**.** For the MDP $(\mathcal{S}, \mathcal{A}, \gamma, P, r, \mathcal{D}_0)$, we assume the following properties hold.

(i) The reward function r and the transition kernel P admit the following representations with respect to the activation function $\widetilde{\sigma}$,

$$r(s, a) = B_r \cdot \int \widetilde{\sigma}\big((s, a, 1)^\top w\big)\mu(\mathrm{d}w), \tag{4.5}$$

$$P(s' \mid s, a) = \int \widetilde{\sigma}\big((s, a, 1)^\top w\big)\varphi(s')\psi(s'; \mathrm{d}w), \tag{4.6}$$

where μ and $\psi(s'; \cdot)$ are probability measures in $\mathscr{P}_2(\mathbb{R}^{d+1})$ for any $s' \in \mathcal{S}$, B_r is a positive scaling parameter, and $\varphi(s') : \mathcal{S} \to \mathbb{R}_+$ is a nonnegative function.

(ii) The reward function r satisfies that $r(s, a) \geq 0$ for any $(s, a) \in \mathcal{S} \times \mathcal{A}$. For the representation of r in (4.5) and the representation of the transition kernel P in (4.6), we assume that

$$\chi^2(\mu \,\|\, \rho_{w,0}) < M_\mu, \qquad \chi^2\big(\psi(s; \cdot) \,\|\, \rho_{w,0}\big) < M_\psi, \quad \forall s \in \mathcal{S},$$
$$\int \varphi(s)\mathrm{d}s \leq M_{1,\varphi}, \qquad \int \varphi(s)^2\mathrm{d}s \leq M_{2,\varphi},$$

where $\rho_{w,0}$ is the marginal distribution of ρ_0 with respect to w, i.e., $\rho_{w,0} = \int \rho_0(\mathrm{d}b, \cdot)$, χ^2 is the chi-squared divergence, and M_μ, M_ψ, $M_{1,\varphi}$, $M_{2,\varphi}$ are absolute constants.

(iii) We assume that there exists an absolute constant $\mathcal{G}$ such that

$$\big\|\psi(s; \cdot) - \psi(s'; \cdot)\big\|_{\dot{H}^{-1}(\mu)} < \mathcal{G}, \qquad \big\|\psi(s; \cdot) - \mu\big\|_{\dot{H}^{-1}(\mu)} < \mathcal{G},$$
$$\big\|\psi(s; \cdot) - \mu\big\|_{\dot{H}^{-1}(\psi(s'; \cdot))} < \mathcal{G}, \qquad \forall s, s' \in \mathcal{S},$$

where $\|\cdot\|_{\dot{H}^{-1}(\cdot)}$ is the weighted homogeneous Sobolev norm.

We remark that by assuming ψ to be a probability measure and $\varphi(s') \geq 0$ in (4.6), the representation of the transition kernel does not loss generality. Specifically, the function class of (4.6) is the same as

$$\mathcal{P} = \Big\{ \int \widetilde{\sigma}((s, a, 1)^\top w)\psi'(s'; \mathrm{d}w) \,\Big|\, \psi'(s'; \cdot) \text{ is a signed measure for any } s' \in \mathcal{S} \Big\}.$$

See §C.1 for a detailed proof. Assumption 4.3 generalizes the linear MDP in [14, 32, 67, 68]. In contrast, our representation of the reward function and the transition kernel benefits from the universal function approximation theorem and is thus not as restrictive as the original linear MDP assumption. Note that the infinite-width neural network has a two-layer structure by (4.2). We establish the

following lemma on the regularity of the representation of the action value function Q^π by such a network.

Lemma 4.4 (Regularity of Representation of Q^π)**.** Suppose that Assumptions 4.2 and 4.3 hold. For any policy π, there exists a probability measure $\rho_\pi \in \mathscr{P}_2(\mathbb{R}^D)$ for the representation of Q^π with the following properties.

(i) For function $Q(s, a; \rho_\pi)$ defined by (3.4) with $\rho = \rho_\pi$ and the action value function $Q^\pi(s, a)$ defined by (2.2), we have $Q(s, a; \rho_\pi) = Q^\pi(s, a)$ for any $(s, a) \in \mathcal{S} \times \mathcal{A}$.

(ii) By letting $B_\beta \geq 2(B_r + \gamma(1-\gamma)^{-1} B_r M_{1,\varphi})$ for the neural network defined in (4.2) and $\rho_0 \sim \mathcal{N}(0, I_D)$ for the initial distribution, we have $\widetilde{W}_2(\rho_\pi, \rho_0) \leq \bar{D}$ for any policy π, where we define $\widetilde{W}_2(\cdot, \cdot) = \alpha W_2(\cdot, \cdot)$ as the scaled W_2 metric. Here constant $\bar{D}$ depends on the discount factor γ and the absolute constants $L_{0,\beta}, L_{1,\beta}, l_\beta, B_r, M_\mu, M_\psi, M_{1,\varphi}, M_{2,\varphi}$ defined in Assumptions 4.2 and 4.3.

Proof. See §B.2 for a detailed proof. □

Property (i) of Lemma 4.4 shows that the action value function Q^π can be parameterized with the infinite-width two-layer neural network $Q(\cdot; \rho_\pi)$ in (3.4). Note that a larger B_β captures a larger function class in (4.3). Without loss of generality, we consider that $B_\beta \geq 2(B_r + \gamma(1-\gamma)^{-1} B_r M_{1,\varphi})$ holds in the sequel. Hence, by Property (ii), it holds that $\widetilde{W}_2(\rho_\pi, \rho_0) \leq O(1)$ for any policy π. In particular, it holds by Property (i) of Lemma 4.4 that $\|Q_t - Q^{\pi_t}\|_{2,\widetilde{\phi}^{\pi_t}} = \|Q(\cdot; \rho_t) - Q(\cdot; \rho_{\pi_t})\|_{2,\widetilde{\phi}^{\pi_t}}$ and we have the following theorem to characterize such an error with regard to the W_2 space.

Theorem 4.5 (Upper Bound of Policy Evaluation Error)**.** Suppose that Assumptions 4.2 and 4.3 hold and $\rho_0 \sim \mathcal{N}(0, I_D)$ is the initial distribution. We specify the weighting distribution $\widetilde{\Phi}^{\pi_t}$ in MF-TD (3.5) as $\widetilde{\Phi}^{\pi_t} = \widetilde{\mathcal{E}}^{\pi_t}_{\widetilde{\phi}^{\pi_t}}$, where $\widetilde{\phi}^{\pi_t} \in \mathscr{P}(\mathcal{S} \times \mathcal{A})$ is the evaluation distribution for the policy evaluation error in Theorem 4.1. Then, it holds that

$$(1 - \sqrt{\gamma}) \cdot \|Q_t - Q^{\pi_t}\|^2_{2,\widetilde{\phi}^{\pi_t}} \leq -\frac{\partial}{\partial t} \frac{\widetilde{W}_2^2(\rho_t, \rho_{\pi_t})}{2\eta} + \Delta_t, \tag{4.7}$$

where

$$\begin{aligned}\Delta_t =& 2\alpha^{1/2}\eta^{-1}\mathcal{B}B_1 \cdot \widetilde{W}_2(\rho_t, \rho_0)\widetilde{W}_2(\rho_t, \rho_{\pi_t}) \\ &+ \alpha^{-1}B_2 \cdot \Big(4B_1 \max\big\{\widetilde{W}_2(\rho_{\pi_t}, \rho_0), \widetilde{W}_2(\rho_t, \rho_0)\big\} + B_r\Big)\widetilde{W}_2(\rho_t, \rho_\pi)^2.\end{aligned}$$

Here B_1 and B_2 are defined in (4.4) of Assumption 4.2, η is the relative TD timescale, α is the scaling parameter of the neural network, and $\widetilde{W}_2 = \alpha W_2$ is the scaled W_2 metric. Moreover, constant $\mathcal{B}$ depends on the dicount factor γ, the scaling parameter B_β in (4.2), and the absolute constants $l_\beta, B_r, M_{1,\varphi}, \mathcal{G}$ defined in Assumptions 4.2 and 4.3.

Proof. See §B.3 for a detailed proof. □

Here we give a nonrigorous discussion on how to upper bound Δ_t in (4.7). If $\widetilde{W}_2(\rho_t, \rho_0) \leq O(1)$ holds for any $t \in [0, T]$, by $\widetilde{W}_2(\rho_{\pi_t}, \rho_0) \leq O(1)$ in Lemma 4.4 and the triangle inequality of W_2 distance [59], it follows that $\widetilde{W}_2(\rho_t, \rho_{\pi_t}) \leq O(1)$ and $\Delta_t \leq O(\alpha^{1/2}\eta^{-1} + \alpha^{-1})$. Taking a time average of integration on both sides of (4.7), the policy evaluation error $\frac{1}{T}\int_0^T \|Q_t - Q^{\pi_t}\|_{2,\widetilde{\phi}^{\pi_t}} \mathrm{d}t$ is then upper bounded by $O(\eta^{-1}T^{-1} + \alpha^{1/2}\eta^{-1} + \alpha^{-1})$. Inspired by such a fact, we introduce the following restarting mechanism to ensure $\widetilde{W}_2(\rho_t, \rho_0) \leq O(1)$.

Restarting Mechanism. Let $\widetilde{W}_0 = \lambda\bar{D}$ be a threshold, where $\bar{D}$ is the upper bound for $\widetilde{W}_2(\rho_\pi, \rho_0)$ by Lemma 4.4, $\lambda \geq 3$ is a constant scaling parameter for the restarting threshold, ρ_t is the distribution of the parameters in the neural network at time t, and ρ_0 is the initial distribution. Whenever we detect that $\widetilde{W}_2(\rho_t, \rho_0)$ reaches $\widetilde{W}_0$ in the update, we pause and reset ρ_t to ρ_0 by resampling the parameters from ρ_0. Then, we reset the critic with the newly sampled parameters while keeping the actor's policy π_t unchanged and continue the update.

The restarting mechanism guarantees $\widetilde{W}_2(\rho_t, \rho_0) \leq \lambda \bar{D}$ by restricting the distribution ρ_t of the parameters to be close to ρ_0. Moreover, by letting $\lambda \geq 3$, we ensure that ρ_{π_t} is realizable by ρ_t since $\widetilde{W}_2(\rho_{\pi_t}, \rho_0) \leq \bar{D} \leq \lambda \bar{D}$, which means that the neural network is capable of capturing the representation of the action value function Q^{π_t}. We remark that by letting $\widetilde{W}_0 = O(1)$, we allow ρ_t to deviate from ρ_0 up to $W_2(\rho_t, \rho) \leq O(\alpha^{-1})$ in the restarting mechanism. In contrast, the NTK regime [15] which corresponds to letting $\alpha = \sqrt{M}$ in (3.1) only allows ρ_t to deviate from ρ_0 by the chi-squared divergence $\chi^2(\rho_t \,\|\, \rho_0) \leq O(M^{-1}) = o(1)$. That is, the NTK regime fails to induce a feature representation significantly different from the initial one. Before moving on, we summarize the construction of the weighting distribution $\widetilde{\Phi}^{\pi_t}$ in Theorem 4.1 and 4.5 as follows,

$$\widetilde{\Phi}^{\pi_t} = \widetilde{\mathcal{E}}^{\pi_t}_{\widetilde{\phi}^{\pi_t}}, \qquad \widetilde{\phi}^{\pi_t} = \frac{1}{2}\widetilde{\phi}_0 + \frac{1}{2}\phi_0 \otimes \pi_t, \qquad \phi_0 = \int_{\mathcal{A}} \widetilde{\phi}_0(\cdot, \mathrm{d}a), \tag{4.8}$$

where $\widetilde{\phi}_0$ is the base distribution. Now we have the following theorem that characterizes the global optimality and convergence of the two-timescale AC with restarting mechanism.

Theorem 4.6 (Global Optimality and Convergence Rate of Two-timescale AC with Restarting Mechanism)**.** Suppose that (4.8) and Assumptions 4.2 and 4.3 hold. With the restarting mechanism, it holds that

$$\frac{1}{T}\int_0^T \big(J(\pi^*) - J(\pi_t)\big)\mathrm{d}t \leq \underbrace{\frac{\zeta}{T}}_{\text{(a)}} + \underbrace{4\kappa\sqrt{\alpha^{-1}S_1 + \alpha^{1/2}\eta^{-1}S_2 + \frac{\eta^{-1}\bar{D}^2}{2T(1-\sqrt{\gamma})}}}_{\text{(b)}}, \tag{4.9}$$

where we have

$$\zeta = \mathbb{E}_{s\sim\mathcal{E}^{\pi^*}_{\mathcal{D}_0}}\Big[\mathrm{KL}\big(\pi^*(\cdot\,|\,s)\,\|\,\pi_0(\cdot\,|\,s)\big)\Big], \quad \kappa = \bigg\|\frac{\widetilde{\mathcal{E}}^{\pi^*}_{\mathcal{D}_0}}{\widetilde{\phi}_0}\bigg\|_\infty,$$
$$S_1 = \frac{(1+\lambda)^2\bar{D}^2 B_2(4B_1\lambda\bar{D} + B_r)}{1-\sqrt{\gamma}}, \quad S_2 = \frac{2\mathcal{B}B_1\lambda(1+\lambda)\bar{D}^2}{1-\sqrt{\gamma}}.$$

Here B_r, B_1 and B_2 are defined in Assumption 4.2 and 4.3, $\bar{D}$ is the upper bound for $\widetilde{W}_2(\rho_\pi, \rho_0)$ in Lemma 4.4, $\mathcal{B}$ depends on the discount factor γ and the absolute constants defined in Assumption 4.2 and 4.3, and λ is the scaling parameter for the restarting threshold. Besides, it holds for the total restarting number N that

$$N \leq (\lambda - 2)^{-1}\big((\alpha^{-1}\eta S_1 + \alpha^{1/2}S_2)2T\bar{D}^{-2}(1-\sqrt{\gamma}) + 1\big).$$

Proof. See §B.4 for a detailed proof. □

Note that for a given MDP with starting distribution $\mathcal{D}_0$, the expected KL-divergence ζ and the concentrability coefficient κ are both independent of the two-timescale update. We remark that our condition for (4.9) to be bounded is not restrictive. Specifically, we only need a given π_0 and $\widetilde{\phi}_0$ such that the KL-divergence $\zeta < \infty$ and the concentrability coefficient $\kappa < \infty$, which is weaker than the concentrability coefficient used in [7, 23, 24, 38, 39, 43, 48, 57, 61].

The first term (a) on the right-hand side of (4.9) diminishes as $T \to \infty$. The second term (b) corresponds to the policy evaluation error. We give an example to demonstrate the convergence of the two-time AC. We let the scaling parameter $\lambda = 3$ for the restarting threshold. By letting $\eta = \alpha^{3/2}$, it holds that (b) $= O(\alpha^{-1/2})$ as $\alpha \to \infty$. Thus, we have $\frac{1}{T}\int_0^T \big(J(\pi^*) - J(\pi_t)\big)\mathrm{d}t \leq O(T^{-1} + O(\alpha^{-1/2}) + O(\alpha^{-3/4}T^{-1/2}))$. Note that $\eta = \alpha^{3/2}$ shows that the critic has a larger relative TD timescale in (3.5). As for the total number of restartings N, it holds that $N \leq O(\alpha^{1/2}T)$ as $\alpha \to \infty$, which induces a tradeoff, i.e., a larger α guarantees a smaller gap in $\frac{1}{T}\int_0^T \big(J(\pi^*) - J(\pi_t)\big)\mathrm{d}t$ but yields in more restartings and a larger relative TD timescale.

References

[1] Agazzi, A. and Lu, J. (2019). Temporal-difference learning for nonlinear value function approximation in the lazy training regime. *arXiv preprint arXiv:1905.10917*.

[2] Agazzi, A. and Lu, J. (2020). Global optimality of softmax policy gradient with single hidden layer neural networks in the mean-field regime. *arXiv preprint arXiv:2010.11858*.

[3] Agostinelli, F., McAleer, S., Shmakov, A. and Baldi, P. (2019). Solving the rubik's cube with deep reinforcement learning and search. *Nature Machine Intelligence*, **1** 356–363.

[4] Akkaya, I., Andrychowicz, M., Chociej, M., Litwin, M., McGrew, B., Petron, A., Paino, A., Plappert, M., Powell, G., Ribas, R. et al. (2019). Solving rubik's cube with a robot hand. *arXiv preprint arXiv:1910.07113*.

[5] Ambrosio, L. and Gigli, N. (2013). A user's guide to optimal transport. In *Modelling and Optimisation of Flows on Networks*. Springer, 1–155.

[6] Ambrosio, L., Gigli, N. and Savaré, G. (2008). *Gradient flows: In metric spaces and in the space of probability measures*. Springer.

[7] Antos, A., Szepesvári, C. and Munos, R. (2008). Learning near-optimal policies with Bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, **71** 89–129.

[8] Barron, A. R. (1993). Universal approximation bounds for superpositions of a sigmoidal function. *IEEE Transactions on Information Theory*, **39** 930–945.

[9] Berner, C., Brockman, G., Chan, B., Cheung, V., Debiak, P., Dennison, C., Farhi, D., Fischer, Q., Hashme, S., Hesse, C. et al. (2019). Dota 2 with large scale deep reinforcement learning. *arXiv preprint arXiv:1912.06680*.

[10] Bhatnagar, S., Ghavamzadeh, M., Lee, M. and Sutton, R. S. (2008). Incremental natural actor-critic algorithms. In *Advances in Neural Information Processing Systems*.

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

[12] Börgers, T. and Sarin, R. (1997). Learning through reinforcement and replicator dynamics. *Journal of economic theory*, **77** 1–14.

[13] Borkar, V. S. (2009). *Stochastic approximation: A dynamical systems viewpoint*. Springer.

[14] Cai, Q., Yang, Z., Jin, C. and Wang, Z. (2019). Provably efficient exploration in policy optimization. *arXiv preprint arXiv:1912.05830*.

[15] 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*.

[16] Chen, M., Wang, Y., Liu, T., Yang, Z., Li, X., Wang, Z. and Zhao, T. (2020). On computation and generalization of generative adversarial imitation learning. *arXiv preprint arXiv:2001.02792*.

[17] Chen, Z., Cao, Y., Gu, Q. and Zhang, T. (2020). Mean-field analysis of two-layer neural networks: Non-asymptotic rates and generalization bounds. *arXiv preprint arXiv:2002.04026*.

[18] Chizat, L. and Bach, F. (2018). A note on lazy training in supervised differentiable programming. *arXiv preprint arXiv:1812.07956*.

[19] Chizat, L. and Bach, F. (2018). On the global convergence of gradient descent for overparameterized models using optimal transport. In *Advances in Neural Information Processing Systems*.

[20] Duan, Y., Chen, X., Houthooft, R., Schulman, J. and Abbeel, P. (2016). Benchmarking deep reinforcement learning for continuous control. In *International Conference on Machine Learning*.

[21] Fang, C., Dong, H. and Zhang, T. (2019). Over parameterized two-level neural networks can learn near optimal feature representations. *arXiv preprint arXiv:1910.11508*.

[22] Fang, C., Gu, Y., Zhang, W. and Zhang, T. (2019). Convex formulation of overparameterized deep neural networks. *arXiv preprint arXiv:1911.07626*.

[23] Farahmand, A.-m., Ghavamzadeh, M., Szepesvári, C. and Mannor, S. (2016). Regularized policy iteration with nonparametric function spaces. *The Journal of Machine Learning Research*, **17** 4809–4874.

[24] Farahmand, A.-m., Szepesvári, C. and Munos, R. (2010). Error propagation for approximate policy and value iteration. In *Advances in Neural Information Processing Systems*.

[25] Friedrichs, K. O. (1944). The identity of weak and strong extensions of differential operators. *Transactions of the American Mathematical Society*, **55** 132–151.

[26] Fu, Z., Yang, Z. and Wang, Z. (2020). Single-timescale actor-critic provably finds globally optimal policy. *arXiv preprint arXiv:2008.00483*.

[27] Harker, P. T. and Pang, J.-S. (1990). Finite-dimensional variational inequality and nonlinear complementarity problems: A survey of theory, algorithms and applications. *Mathematical Programming*, **48** 161–220.

[28] Hennes, D., Morrill, D., Omidshafiei, S., Munos, R., Perolat, J., Lanctot, M., Gruslys, A., Lespiau, J.-B., Parmas, P., Duéñez-Guzmán, E. et al. (2020). Neural replicator dynamics: Multiagent learning via hedging policy gradients. In *International Conference on Autonomous Agents and MultiAgent Systems*.

[29] Hong, M., Wai, H.-T., Wang, Z. and Yang, Z. (2020). A two-timescale framework for bilevel optimization: Complexity analysis and application to actor-critic. *arXiv preprint arXiv:2007.05170*.

[30] Jacot, A., Gabriel, F. and Hongler, C. (2018). Neural tangent kernel: Convergence and generalization in neural networks. In *Advances in Neural Information Processing Systems*.

[31] Javanmard, A., Mondelli, M. and Montanari, A. (2019). Analysis of a two-layer neural network via displacement convexity. *arXiv preprint arXiv:1901.01375*.

[32] Jin, C., Yang, Z., Wang, Z. and Jordan, M. I. (2020). Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*.

[33] Kakade, S. and Langford, J. (2002). Approximately optimal approximate reinforcement learning. In *International Conference on Machine Learning*, vol. 2.

[34] Kakade, S. M. (2002). A natural policy gradient. In *Advances in Neural Information Processing Systems*.

[35] Khodadadian, S., Doan, T. T., Maguluri, S. T. and Romberg, J. (2021). Finite sample analysis of two-time-scale natural actor-critic algorithm. *arXiv preprint arXiv:2101.10506*.

[36] Konda, V. R. and Tsitsiklis, J. N. (2000). Actor-critic algorithms. In *Advances in Neural Information Processing Systems*.

[37] Kushner, H. and Yin, G. G. (2003). *Stochastic approximation and recursive algorithms and applications*. Springer.

[38] Lazaric, A., Ghavamzadeh, M. and Munos, R. (2016). Analysis of classification-based policy iteration algorithms. *The Journal of Machine Learning Research*, **17** 583–612.

[39] Liu, B., Cai, Q., Yang, Z. and Wang, Z. (2019). Neural proximal/trust region policy optimization attains globally optimal policy. *arXiv preprint arXiv:1906.10306*.

[40] Lu, Y., Ma, C., Lu, Y., Lu, J. and Ying, L. (2020). A mean field analysis of deep resnet and beyond: Towards provably optimization via overparameterization from depth. In *International Conference on Machine Learning*. PMLR.

[41] Mei, S., Misiakiewicz, T. and Montanari, A. (2019). Mean-field theory of two-layers neural networks: dimension-free bounds and kernel limit. In *Conference on Learning Theory*. PMLR.

[42] Mei, S., Montanari, A. and Nguyen, P.-M. (2018). A mean field view of the landscape of two-layer neural networks. *Proceedings of the National Academy of Sciences*, **115** E7665–E7671.

[43] Munos, R. and Szepesvári, C. (2008). Finite-time bounds for fitted value iteration. *The Journal of Machine Learning Research*, **9** 815–857.

[44] Otto, F. and Villani, C. (2000). Generalization of an inequality by Talagrand and links with the logarithmic Sobolev inequality. *Journal of Functional Analysis*, **173** 361–400.

[45] Peters, J. and Schaal, S. (2008). Natural actor-critic. *Neurocomputing*, **71** 1180–1190.

[46] Peyre, R. (2011). Comparison between w_2 distance and $\dot{H}^{-1}$ norm, and localisation of wasserstein distance. *arXiv preprint arXiv:1104.4631*.

[47] Pinkus, A. (1999). Approximation theory of the MLP model in neural networks. *Acta Numerica*, **8** 143–195.

[48] Scherrer, B., Ghavamzadeh, M., Gabillon, V., Lesner, B. and Geist, M. (2015). Approximate modified policy iteration and its application to the game of Tetris. *The Journal of Machine Learning Research*, **16** 1629–1676.

[49] Schulman, J., Wolski, F., Dhariwal, P., Radford, A. and Klimov, O. (2017). Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

[50] Schuster, P. and Sigmund, K. (1983). Replicator dynamics. *Journal of theoretical biology*, **100** 533–538.

[51] Silver, D., Huang, A., Maddison, C. J., Guez, A., Sifre, L., Van Den Driessche, G., Schrittwieser, J., Antonoglou, I., Panneershelvam, V., Lanctot, M. et al. (2016). Mastering the game of Go with deep neural networks and tree search. *Nature*, **529** 484–489.

[52] 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** 354.

[53] Sirignano, J. and Spiliopoulos, K. (2020). Mean field analysis of neural networks: A central limit theorem. *Stochastic Processes and their Applications*, **130** 1820–1852.

[54] Sirignano, J. and Spiliopoulos, K. (2020). Mean field analysis of neural networks: A law of large numbers. *SIAM Journal on Applied Mathematics*, **80** 725–752.

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

[56] Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. MIT press.

[57] Szepesvári, C. and Munos, R. (2005). Finite time bounds for sampling based fitted value iteration. In *International Conference on Machine Learning*. ACM.

[58] Villani, C. (2003). *Topics in optimal transportation*. American Mathematical Society.

[59] Villani, C. (2008). *Optimal transport: Old and new*. Springer.

[60] Vinyals, O., Babuschkin, I., Czarnecki, W. M., Mathieu, M., Dudzik, A., Chung, J., Choi, D. H., Powell, R., Ewalds, T., Georgiev, P. et al. (2019). Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature*, **575** 350–354.

[61] Wang, L., Cai, Q., Yang, Z. and Wang, Z. (2019). Neural policy gradient methods: Global optimality and rates of convergence. *arXiv preprint arXiv:1909.01150*.

[62] Watkins, C. and Dayan, P. (1992). Q-learning. *Machine Learning*, **8** 279–292.

[63] Wei, C., Lee, J. D., Liu, Q. and Ma, T. (2019). Regularization matters: Generalization and optimization of neural nets vs their induced kernel. In *Advances in Neural Information Processing Systems*.

[64] Wu, Y., Zhang, W., Xu, P. and Gu, Q. (2020). A finite time analysis of two time-scale actor critic methods. *arXiv preprint arXiv:2005.01350*.

[65] Xu, T., Wang, Z. and Liang, Y. (2020). Improving sample complexity bounds for actor-critic algorithms. *arXiv preprint arXiv:2004.12956*.

[66] Xu, T., Wang, Z. and Liang, Y. (2020). Non-asymptotic convergence analysis of two time-scale (natural) actor-critic algorithms. *arXiv preprint arXiv:2005.03557*.

[67] Yang, L. F. and Wang, M. (2019). Reinforcement learning in feature space: Matrix bandit, kernels, and regret bound. *arXiv preprint arXiv:1905.10389*.

[68] Yang, L. F. and Wang, M. (2019). Sample-optimal parametric q-learning using linearly additive features. *arXiv preprint arXiv:1902.04779*.

[69] Zhang, Y., Cai, Q., Yang, Z., Chen, Y. and Wang, Z. (2020). Can temporal-difference and q-learning learn representation? a mean-field theory. *arXiv preprint arXiv:2006.04761*.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?
   A: [Yes] .
   (b) Did you describe the limitations of your work?
   A: [Yes] We refer to the limitations in appendix.
   (c) Did you discuss any potential negative societal impacts of your work?
   A: [No] .
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them?
   A: [Yes] we confirmed.
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results?
   A: [Yes] We explicitly stated the assumptions.
   (b) Did you include complete proofs of all theoretical results?
   A: [Yes] We provide complete proofs in supplementary materials.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)?
   A: [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)?
   A: [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)?
   A: [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)?
   A: [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators?
   A: [N/A]
   (b) Did you mention the license of the assets?
   A: [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL?
   A: [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating?
   A: [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content?
   A: [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable?
   A: [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable?
   A: [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation?
   A: [N/A]