PPO-Clip Attains Global Optimality: Towards Deeper Understandings of Clipping

Nai-Chieh Huang, Ping-Chun Hsieh, Kuo-Hao Ho, I-Chen Wu

Department of Computer Science, National Yang Ming Chiao Tung University, Hsinchu, Taiwan
{naich.cs09, pinghsieh, lukewayne123.cs05}@nycu.edu.tw, icwu@cs.nycu.edu.tw

Abstract

Proximal Policy Optimization algorithm employing a clipped surrogate objective (PPO-Clip) is a prominent exemplar of the policy optimization methods. However, despite its remarkable empirical success, PPO-Clip lacks theoretical substantiation to date. In this paper, we contribute to the field by establishing the first global convergence results of a PPO-Clip variant in both tabular and neural function approximation settings. Our findings highlight the $O(1/\sqrt{T})$ min-iterate convergence rate specifically in the context of neural function approximation. We tackle the inherent challenges in analyzing PPO-Clip through three central concepts: (i) We introduce a generalized version of the PPO-Clip objective, illuminated by its connection with the hinge loss. (ii) Employing entropic mirror descent, we establish asymptotic convergence for tabular PPO-Clip with direct policy parameterization. (iii) Inspired by the tabular analysis, we streamline convergence analysis by introducing a two-step policy improvement approach. This decouples policy search from complex neural policy parameterization using a regression-based update scheme. Furthermore, we gain deeper insights into the efficacy of PPO-Clip by interpreting these generalized objectives. Our theoretical findings also mark the first characterization of the influence of the clipping mechanism on PPO-Clip convergence. Importantly, the clipping range affects only the pre-constant of the convergence rate.

1 Introduction

Policy optimization is a prevalent method for solving reinforcement learning problems, involving iterative parameter updates to maximize objectives. Policy gradient methods, a prominent subset of this approach, were introduced as a direct solution using gradient descent. Their primary aim is to identify an optimal policy that maximizes the total expected reward through interactions with the environment. The selection of an appropriate step size is crucial as it significantly influences policy gradient algorithm performance. Addressing this challenge, Trust Region Policy Optimization (TRPO) was created (Schulman et al. 2015). Utilizing a trust-region approach with a second-order approximation, TRPO guarantees substantial policy improvement. Unlike computationally intensive TRPO, Proximal Policy Optimization (PPO) (Schulman et al. 2017) leverages first-order derivatives for policy improvement. PPO encompasses two main variants: PPO-KL and PPO-Clip, each with distinct characteristics. PPO-KL adds a Kullback-Leibler divergence penalty to the objective, while PPO-Clip integrates probability ratio clipping. These variants showcase remarkable performance across various environments, with PPO standing out for its computational efficiency (Chen, Peng, and Zhang 2018; Ye et al. 2020; Byun, Kim, and Wang 2020).

Given the empirical success of these policy optimization algorithms, recent works have made significant strides in enhancing their theoretical guarantees. In particular, (Agarwal et al. 2020; Bhandari and Russo 2019) prove the global convergence result of the policy gradient algorithm under different settings. Additionally, (Mei et al. 2020) establishes the convergence rates of the softmax policy gradient in both the standard and the entropy-regularized settings. Furthermore, it has been shown that various policy gradient algorithms also enjoy global convergence (Fazel et al. 2018; Liu et al. 2020; Wang et al. 2021). In the context of TRPO and PPO, (Shani, Efroni, and Mannor 2020) have utilized the mirror descent method to establish the convergence rate of adaptive TRPO under both the standard and entropy-regularized settings. Furthermore, (Liu et al. 2019) have provided the convergence rate of PPO-KL and TRPO under neural function approximation.[1] By contrast, despite that PPO-Clip is computationally efficient and empirically successful, the following question about the theory of PPO-Clip remains largely open: *Does PPO-Clip enjoy provable global convergence or have any convergence rate guarantee?*

In this paper, we answer the above question affirmatively. To begin with, we generalize the PPO-Clip objective to encompass a wider range of variants, enhancing our comprehension of its efficacy. Accordingly, we present the first-ever global convergence guarantee for a PPO-Clip variant under both tabular and neural function approximation. Notably, through convergence analysis, we offer two pivotal insights into the clipping mechanism: (i) Under PPO-Clip, the policy updates scale with advantage magnitudes, while the sign dictates whether to increase or decrease the action probabilities. Notably, given the representation power of neural networks, incorrect signs typically emerge when the advan-

[1] For the detailed discussion about related work, please refer to Appendix H.

tage magnitudes are nearly zero. In such cases, these values insignificantly contribute to the objective, preserving the objective accuracy despite the incorrect signs. This perspective illuminates the robustness and empirical success of PPO-Clip. (ii) Through our convergence analysis, we demonstrate that the clipping range merely affects the pre-constant of the convergence rate, not the asymptotic behavior. All the code is available at https://github.com/NYCU-RL-Bandits-Lab/Neural-PPO-Clip and the full version is provided at https://arxiv.org/abs/2312.12065.

Our Contributions. We summarize the main contributions of this paper as follows:

- To establish the global convergence of PPO-Clip, we leverage the connection between PPO-Clip and the hinge loss, leading to the formulation of generalized PPO-Clip objectives. Additionally, we harness the power of the entropic mirror descent (EMDA) (Beck and Teboulle 2003) for tabular PPO-Clip under direct policy parameterization, thereby demonstrating its asymptotic convergence.
- Inspired by the tabular analysis, we present a two-step policy improvement framework based on EMDA for Neural PPO-Clip. This framework enhances the manageability of the analysis by effectively separating policy search from policy parameterization. Accordingly, we establish the first global convergence result and explicitly characterize the $O(1/\sqrt{T})$ min-iterate convergence rate for the generalized PPO-Clip and hence provide an affirmative answer to one critical open question about PPO-Clip.
- We gain deeper insights into the PPO-Clip performance. Our theoretical findings yield two key insights into the clipping mechanism, as mentioned earlier. Furthermore, our analysis extends seamlessly to various Neural PPO-Clip variants with different classifiers, guided by the provided sufficient conditions.

2 Preliminaries

Markov Decision Processes. Consider a discounted Markov Decision Process $(\mathcal{S}, \mathcal{A}, \mathcal{P}, R, \gamma, \mu)$, where $\mathcal{S}$ is the state space (possibly *infinite*), $\mathcal{A}$ is a *finite* action space, $\mathcal{P} : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \rightarrow [0, 1]$ is the transition dynamic of the environment, $R : \mathcal{S} \times \mathcal{A} \rightarrow [0, R_{\max}]$ is the bounded reward function, $\gamma \in (0, 1)$ is the discount factor, and μ is the initial state distribution. Given a policy $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$, where $\Delta(\mathcal{A})$ is the unit simplex over $\mathcal{A}$, we define the state-action value function $Q^\pi(\cdot, \cdot) := \mathbb{E}_{a_t\sim\pi(\cdot|s_t), s_{t+1}\sim\mathcal{P}(\cdot|s_t,a_t)}[\sum_{t=0}^{\infty} \gamma^t R(s_t, a_t)|s_0 = s, a_0 = a]$. Moreover, we define $V^\pi(s) := \mathbb{E}_{a\sim\pi(\cdot|s)}[Q^\pi(s, a)]$ and $A^\pi(s, a) := Q^\pi(s, a) - V^\pi(s)$. Also, we denote π^* as an optimal policy that attains the maximum total expected reward and denote π_0 as the uniform policy. We introduce $\nu_\pi(s) = (1 - \gamma)\sum_{t=0}^{\infty} \gamma^t \mathbb{P}(s_t = s|s_0 \sim \mu, \pi)$ as the discounted state visitation distribution induced by π and $\sigma_\pi(s, a) = \nu_\pi(s) \cdot \pi(a|s)$ as the state-action visitation distribution induced by π. In addition, we define the distribution ν^* and σ^* as the discounted state visitation distribution and the state-action visitation distribution induced by the optimal policy π^*, respectively. Moreover, we define $\tilde{\sigma}_\pi = \nu_\pi \pi_0$ as the state-action distribution induced by interactions with the environment through π, sampling actions from the uniform policy π_0. We use $\mathbb{E}_{\nu_\pi}[\cdot]$ and $\mathbb{E}_{\sigma_\pi}[\cdot]$ as the shorthand notations of $\mathbb{E}_{s\sim\nu_\pi}[\cdot]$ and $\mathbb{E}_{(s,a)\sim\sigma_\pi}[\cdot]$, respectively.

For the convergence property, we define the total expected reward over the state distribution ν^* as

$$\mathcal{L}(\pi) := \mathbb{E}_{\nu^*}[V^\pi(s)]. \tag{1}$$

Here, a maximizer of (1) is equivalent to the original definition of the optimal policy π^*. We will prove the global convergence by analyzing the difference in $\mathcal{L}$ between our policy and the optimal policy and show that the total expected reward monotonically increases.

Proximal Policy Optimization (PPO). PPO is an empirically successful algorithm that achieves policy improvement by maximizing a surrogate lower bound of the original objective, either through the Kullback-Leibler penalty (termed PPO-KL) or the clipped probability ratio (termed PPO-Clip). PPO-KL and PPO-Clip represent the two main branches of PPO, both aiming to enforce policy constraints during updates for policy improvement. It is crucial to emphasize that PPO-Clip represents a conceptual approach, utilizing the clipping mechanism to achieve policy constraints, rather than being a precise algorithm itself. In this paper, our focus is PPO-Clip. Let $\rho_{s,a}(\theta)$ denote the probability ratio $\frac{\pi_\theta(a|s)}{\pi_{\theta_t}(a|s)}$. PPO-Clip avoids large policy updates by applying a simple heuristic that clips the probability ratio by the clipping range ϵ and thereby removes the incentive for moving $\rho_{s,a}(\theta)$ away from 1. Specifically, the PPO-Clip objective is

$$\begin{aligned} L_t^{\text{clip}}(\theta) = \mathbb{E}_{\sigma_t}[\min\{&\rho_{s,a}(\theta)A^{\pi_{\theta_t}}(s, a), \\ &\text{clip}(\rho_{s,a}(\theta), 1 - \epsilon, 1 + \epsilon)A^{\pi_{\theta_t}}(s, a)\}]. \end{aligned} \tag{2}$$

Neural Networks. We introduce the notations and assumptions relevant to neural networks. It is important to highlight that our analysis of neural networks draws inspiration from (Liu et al. 2019), and we adopt their notations to ensure compatibility. Specifically, this paper centers around the analysis of two-layer neural networks. For simplicity, let us consider $(s, a) \in \mathbb{R}^d$ for all $(s, a) \in \mathcal{S} \times \mathcal{A}$. We represent the two-layer neural network as $\text{NN}(\alpha; m)$, where α denotes the network input weights and m represents the network width. These neural networks act as the parameterization for both our policy π_θ and the Q function. The parameterized function associated with $\text{NN}(\alpha; m)$ is depicted as follows:

$$u_\alpha(s, a) = \frac{1}{\sqrt{m}} \sum_{i=1}^{m} b_i \cdot \sigma([\alpha]_i^\top (s, a)), \tag{3}$$

where $\alpha = ([\alpha]_1^\top, \ldots, [\alpha]_m^\top)^\top \in \mathbb{R}^{md}$ is the input weights, with $[\alpha]_i \in \mathbb{R}^d$, $b_i \in \{-1, 1\}$ are the weights of the output, and $\sigma(\cdot)$ refers to the Rectified Linear Unit (ReLU) activation function. The initializations for the input weights α_0 and b_i are provided as follows:

$$b_i \sim \text{Unif}(\{1, -1\}), [\alpha_0]_i \sim \mathcal{N}(0, I_d/d), \tag{4}$$

where both b_i and $[\alpha_0]_i$ are i.i.d. for each $i \in [m]$ and I_d is the $d \times d$ identity matrix. The values of b_i remain fixed following initialization, with the training exclusively focused on adjusting the weights α. To uphold the local linearization characteristics, we employ a projection mechanism that confines the training weights α within an ℓ_2-ball centered at α_0, which is represented as $B_f = \{\alpha : \|\alpha - \alpha_0\|_2 \leq R_f\}$, where f is the canonical name of the networks (It will be f for the policy network and Q for the Q function network in the following section).

Our examination of neural networks is grounded in the subsequent assumptions, which are widely adopted regularity conditions for neural networks in the reinforcement learning literature (Liu et al. 2019; Antos, Szepesvári, and Munos 2007; Farahmand et al. 2016):

Assumption 1 (Q Function Class)**.** We assume that the our neural network class possesses sufficient representational capacity to model the Q function of any given policy π. Specifically, for any $R > 0$, define a function class

$$\mathcal{F}_{R,m} = \Big\{ \frac{1}{\sqrt{m}} \sum_{i=1}^{m} b_i \cdot \mathbb{1}\{[\alpha_0]_i^\top (s,a) > 0\} \cdot [\alpha]_i^\top (s,a) \Big\}, \tag{5}$$

for all α satisfying $\|\alpha - \alpha_0\|_2 \leq R$, where b_i and α_0 are initialized as (4). We assume that $Q^\pi(s,a) \in \mathcal{F}_{R_Q, m_Q}$ for any policy π, where R_Q and m_Q are the projection radius and width of the neural network for Q function.

Given that $\mathcal{T}^\pi Q^\pi$ remains a Q function, Assumption 1 affords us the property of completeness within our function class under the Bellman operator $\mathcal{T}^\pi$.

Notations: We use $\langle a, b\rangle$ and $a \circ b$ to denote the inner product and the Hadamard product, respectively.

3 Generalized PPO-Clip Objectives

Connecting PPO-Clip and Hinge Loss. According to (Hu et al. 2020; Pi et al. 2020), the original PPO-Clip objective could be connected with the hinge loss. Specifically, the gradient of the clipped objective is indeed the negative of the gradient of hinge loss objective, i.e.,

$$\begin{aligned}&\frac{\partial}{\partial \theta} \min\{\rho_{s,a}(\theta) A(s,a), \operatorname{clip}(\rho_{s,a}(\theta), 1-\epsilon, 1+\epsilon) A(s,a)\} \\ &= -\frac{\partial}{\partial \theta} \,|A(s,a)|\, \ell(\operatorname{sign}(A(s,a)), \rho_{s,a}(\theta) - 1, \epsilon),\end{aligned} \tag{6}$$

where $\ell(y_i, f_\theta(x_i), \epsilon)$ is the hinge loss defined as $\max\{0, \epsilon - y_i \cdot f_\theta(x_i)\}$, ϵ is the margin, $y_i \in \{-1, 1\}$ the label corresponding to the data x_i, and $f_\theta(x_i)$ serves as the binary classifier. For completeness, please see Appendix I for a detailed comparison of the two objectives. From the above, maximizing the objective in (2) can be rewritten as minimizing the following loss:

$$\begin{aligned}L(\theta) = \sum_{s \in \mathcal{S}} d_\mu^\pi(s) \sum_{a \in \mathcal{A}} \Big(&\pi(a|s) |A^\pi(s,a)| \cdot \\ &\ell(\operatorname{sign}(A^\pi(s,a)), \rho_{s,a}(\theta) - 1, \epsilon) \Big).\end{aligned} \tag{7}$$

In practice, we draw a batch of state-action pairs and use the sample average to approximately minimize the loss in (7).

Generalized PPO-Clip Objectives. Based on the above reinterpretation of PPO-Clip, we provide a general form of the PPO-Clip loss function from a hinge loss perspective as follows,

$$L_{\text{Hinge}}(\theta) = \frac{1}{|\mathcal{D}|} \sum_{(s,a) \in \mathcal{D}} \text{weight} \times \ell(\text{label}, \text{classifier}, \text{margin}). \tag{8}$$

Different combinations of classifiers, margins, and weights lead to different loss functions, thereby representing diverse algorithms. PPO-Clip is a special case of (8) with a specific classifier $\rho_{s,a}(\theta) - 1$. Another variant, termed PPO-Clip-sub in this paper, can be obtained by employing a subtraction classifier, i.e., $\pi_\theta(a|s) - \pi_{\theta_t}(a|s)$. There are several other variants under this generalized objective by employing distinct classifiers, e.g., $\log(\pi_\theta(a|s)) - \log(\pi_{\theta_t}(a|s))$ and $\sqrt{\rho_{s,a}(\theta)} - 1$. We demonstrate the empirical evaluation of these variants in Section 6. Given the above examples, the proposed objective provides to generalizing PPO-Clip via various classifiers, thereby expanding the objective choices within the context of PPO-Clip. This generalization also connects the PPO-Clip with the classifier selection paradigm. Additionally, this generalized objective provide an intution to understand more about the clipping mechanism. Please refer to Section 5.4.

4 Tabular PPO-Clip

4.1 Direct Policy Parameterization

In this section, we study the global convergence of PPO-Clip with direct parameterization, i.e., policies are parameterized by $\pi(a|s) = \theta_{s,a}$, where $\theta_s \in \Delta(\mathcal{A})$ denotes the vector $\theta_{s,\cdot}$ and $\theta \in \Delta(\mathcal{A})^{|\mathcal{S}|}$. We use $V^{(t)}(s)$ and $A^{(t)}(s,a)$ as the shorthands for $V^{\pi^{(t)}}(s)$ and $A^{\pi^{(t)}}(s,a)$, respectively.

For the sake of clarity, we focus our discussion on the original PPO-Clip rather than delving into the broader scope of the generalized objective (8). Furthermore, we also provide additional analysis for other PPO-Clip variants with different classifiers in Appendix F. Note that by choosing the weight as $|A^{(t)}(s,a)|$, the classifier as $\rho_{s,a}^{(t)}(\theta) - 1$, and the margin as ϵ in (8) at the t-th iteration, the generalized objective would recover the form of the objective of PPO-Clip, which denoted as $\hat{L}^{(t)}(\theta)$. The detailed algorithm is shown in Appendix A as Algorithm 7.

In each iteration, PPO-Clip updates the policy by minimizing the loss $\hat{L}^{(t)}(\theta)$ via the EMDA (Beck and Teboulle 2003). While there are alternative ways to minimize the loss $\hat{L}^{(t)}(\theta)$ over $\Delta(\mathcal{A})^{|\mathcal{S}|}$ (e.g., the projected subgradient method), we leverage EMDA for the following two reasons: (i) PPO-Clip achieves policy improvement by increasing or decreasing the probability of those state-action pairs in $\mathcal{D}^{(t)}$ based on the sign of $A^{(t)}(s,a)$ as well as properly reallocating the probabilities of those state-action pairs *not* contained in the batch (to ensure the probability sum is one). Using EMDA enforces a proper reallocation in PPO-Clip, as shown later in the proof of Theorem 1 in Appendix E; (ii) The exponentiated gradient scheme of EMDA guarantees $\pi^{(t)}$ remains strictly positive for all state-action pairs in each iteration t, ensuring the well-defined nature of the probability ratio $\rho_{s,a}(\theta)$ used in PPO-Clip. In this section, we consider the stylized setting with tabular policy and true advantage mainly for motivating the PPO-Clip method and its analysis.

4.2 Global Convergence of PPO-Clip with Direct Parameterization

We first make the following assumptions. Note that we only consider these assumptions in the tabular case.

Assumption 2 (Infinite Visitation to Each State-Action Pair)**.** Each state-action pair (s, a) appears infinitely often in $\{\mathcal{D}^{(\tau)}\}$, i.e., $\lim_{t\to\infty}\sum_{\tau=0}^{t}\mathbb{1}\{(s,a)\in\mathcal{D}^{(\tau)}\}=\infty$, with probability one.

Assumption 3. In each iteration t, the state-action pairs in $\mathcal{D}^{(t)}$ have distinct states.

Assumption 2 resembles the standard infinite-exploration condition commonly used in the temporal-difference methods, such as Sarsa (Singh et al. 2000). Assumption 3 is rather mild: (i) This can be met by post-processing the mini-batch of state-action pairs via an additional sub-sampling step; (ii) In most RL problems with discrete actions, the state space is typically much larger than the action space.

Theorem 1 (Global Convergence of PPO-Clip)**.** *Under PPO-Clip, we have $V^{(t)}(s) \to V^{\pi^*}(s)$ as $t \to \infty$, $\forall s \in \mathcal{S}$, with probability one.*

The proof of Theorem 1 is provided in Appendix E. We highlight the main ideas behind the proof of Theorem 1: (i) *State-wise policy improvement:* Through the lens of generalized objective, we show that PPO-Clip enjoys state-wise policy improvement in every iteration with the help of the EMDA subroutine. This property greatly facilitates the rest of the convergence analysis. (ii) *Quantifying the probabilities of those actions with positive or negative advantages in the limit*: By (i), we know the limits of the value functions and the advantage function all exist. Then, we proceed to show that the actions with positive advantages in the limit cannot exist by establishing a contradiction. The above also manifests how reinterpreting PPO-Clip helps with establishing the convergence guarantee.

5 Neural PPO-Clip

In this section, we begin by illustrating the process of decoupling policy search and policy parameterization, drawing inspiration from the tabular case. Subsequently, we provide a comprehensive overview of the neural PPO-Clip algorithm. We proceed to delineate the intricacies posed by our analysis and present our results on the min-iterate convergence rate, both for the generalized PPO-Clip. In particular, the convergence rate of PPO-Clip can be view as a special case of our general results. Lastly, we offer a profound insight into the understanding of the clipping mechanism.

5.1 EMDA-Based Policy Search

Drawing inspiration from the tabular case, we proceed to present our two-step policy improvement scheme based on EMDA, and we call it EMDA-based Policy Search. Specifically, this scheme consists of two subroutines:

- **Direct policy search**: In this step, we directly search for an improved policy in the policy space by EMDA. More specifically, in each iteration t, we do a policy search by applying EMDA with direct parameterization to minimize the generalized PPO-Clip objective in (8) for finitely many iterations K and thereby obtain an improved policy $\widehat{\pi}_{t+1}$ as the target policy. The pseudo code of EMDA is provided in Algorithm 2. Notably, under EMDA, we can obtain an explicit expression of the target policy $\widehat{\pi}_{t+1}$.
- **Neural approximation for the target policy**: Given the target policy $\widehat{\pi}_{t+1}$ obtained by EMDA, we then approximate it in the parameter space by utilizing the representation power of neural networks via a regression-based policy update scheme (e.g., by using the mean-squared error loss). The detailed neural parameterization will be described in the next subsection.

While the decision to employ EMDA is inspired by the tabular case, there are two primary motivations and benefits for integrating EMDA with direct parameterization:

- **Decoupling improvement and approximation:** One major goal of this paper is to provide rigorous theoretical guarantees for PPO-Clip under neural function approximation. To make the analysis tractable and general, we would like to decouple policy improvement and function approximation of the policy. To achieve this, we adopt the EMDA-based two-step approach outlined previously.
- **EMDA-induced closed-form expression of the target policy:** For policy optimization analysis, the goal is often to derive a closed-form optimal solution for the policy improvement objective as the ideal target policy. However, such a closed-form optimal solution of an *arbitrary* objective function does not always exist. A case in point is the loss function of PPO-Clip. From this view, EMDA, which enjoys closed-form updates, substantially facilitates the convergence analysis, as can be observed in Proposition 1 presented in the subsequent subsection 5.2.

5.2 Neural PPO-Clip

Parameterization Setting. At each iteration t, we parameterize our policy as an energy-based policy $\pi_{\theta_t}(a|s) \propto \exp\{\tau_t^{-1} f_{\theta_t}(s,a)\}$, where τ_t denotes the temperature parameter and $f_{\theta_t}(s,a) = \mathrm{NN}(\theta_t; m_f)$ corresponds to the energy functions. The width of the neural network f_θ is denoted as m_f, as defined in Section 2. Likewise, we parameterize our state-action value function as $Q_\omega(s,a) = \mathrm{NN}(\omega; m_Q)$, with width m_Q of the neural network Q_ω. Concurrently, we define $V_\omega(s)$ as the value function derived from the Bellman Expectation Equation. Also, we define $A_\omega(s,a) := Q_\omega(s,a) - V_\omega(s)$ to be the advantage function.

Policy Improvement. According to the EMDA-based Policy Search framework presented above, we first give the closed-form of the obtained target policy of Neural PPO-Clip as follows. The detailed proof is in Appendix B.

Proposition 1 (EMDA Target Policy)**.** *For the target policy obtained by the EMDA subroutine at the t-th iteration, we have*

$$\log \widehat{\pi}_{t+1}(a|s) \propto C_t(s,a) A_{\omega_t}(s,a) + \tau_t^{-1} f_{\theta_t}(s,a), \quad (9)$$

where $C_t(s,a)A_{\omega_t}(s,a) = -\sum_{k=0}^{K-1} \eta g_{s,a}^{(k)}$ as given in Algorithm 2.

Recall that the target policy $\widehat{\pi}$ is the direct parameterization in the policy space, but our policy π_θ is an energy-based (softmax) policy that is proportional to the exponentiated energy function. This explains why we consider the $\log \widehat{\pi}_{t+1}(a|s)$ in Proposition 1. Another benefit of using EMDA is that it closely matches the energy-based policies considered in Neural PPO-Clip due to the inherent exponentiated gradient update.

Then, we discuss the details of the neural function approximation of our policy. After obtaining the target policy by Proposition 1, we solve the Mean Squared Error (MSE) subproblem with respect to θ to approximate the target policy as follows:

$$\mathbb{E}_{\tilde{\sigma}_t}[(f_\theta(s,a) - \tau_{t+1}(C_t(s,a)A_{\omega_t}(s,a) + \tau_t^{-1} f_{\theta_t}(s,a)))^2]. \tag{10}$$

Notice that we consider the state-action distribution $\tilde{\sigma}_t$ sampling the action through a uniform policy π_0. In this manner, we use more exploratory data to improve our current policy. In particular, we use the SGD to tackle the above subproblem, and the pseudo code is provided in Appendix A.

Policy Evaluation. To evaluate Q, we use a neural network to approximate the true state-action value function $Q^{\pi_{\theta_t}}$ by solving the Mean Square Bellman Error (MSBE) subproblem. The MSBE subproblem is to minimize the following objective with respect to ω at each iteration t:

$$\mathbb{E}_{\sigma_t}[(Q_\omega(s,a) - [\mathcal{T}^{\pi_{\theta_t}} Q_\omega](s,a))^2], \tag{11}$$

where $\mathcal{T}^{\pi_{\theta_t}}$ is the Bellman operator of policy π_{θ_t} such that

$$\begin{aligned}&[\mathcal{T}^{\pi_{\theta_t}} Q_\omega](s,a)\\ &= \mathbb{E}[r(s,a) + \gamma Q_\omega(s',a') \mid s' \sim \mathcal{P}(\cdot|s,a), a' \sim \pi_{\theta_t}(\cdot|s')].\end{aligned} \tag{12}$$

The pseudo code of neural TD update for state-action value function Q_ω is in Appendix A. It is worth mentioning that this variant of Neural PPO-Clip is not a fully on-policy algorithm. Although we interact with the environment by our current policy, we sample the actions by the uniform policy π_0 for policy improvement. We provide the pseudo code of Neural PPO-Clip as the following Algorithm 1 (please refer to Algorithm 3 in Appendix A for the complete version) and the pseudo code of EMDA as Algorithm 2. The pseudo code of Algorithms 5-6 used by Algorithm 1 is in Appendix A.

Algorithm 1: Neural PPO-Clip

```
Input: $L_{\text{Hinge}}(\theta)$, $T$, $\epsilon$, EMDA step size $\eta$, number of EMDA iterations $K$, number of SGD, TD update iterations $T_{\text{upd}}$
Initialization: uniform policy $\pi_{\theta_0}$
for $t = 1, \cdots, T-1$ do
    Set temperature parameter $\tau_{t+1}$
    Sample the tuple $\{s_i, a_i, a_i^0, s_i', a_i'\}_{i=1}^{T_{\text{upd}}}$
    Run EMDA as Algorithm 2 with $L_{\text{Hinge}}(\theta)$
    Run TD as Algorithm 5: $Q_{\omega_t} = \text{NN}(\omega_t; m_Q)$
    Calculate $V_{\omega_t}$ and the advantage $A_{\omega_t} = Q_{\omega_t} - V_{\omega_t}$
    Run SGD as Algorithm 6: $f_{\theta_{t+1}} = \text{NN}(\theta_{t+1}; m_f)$
    Update the policy $\pi_{\theta_{t+1}} \propto \exp\{\tau_{t+1}^{-1} f_{\theta_{t+1}}\}$
end for
```

Algorithm 2: EMDA

```
Input: $L_{\text{Hinge}}(\theta)$, EMDA step size $\eta$, number of EMDA iterations $K$, initial policy $\pi_{\theta_t}$, sample batch $\{s_i\}_{i=1}^{T_{\text{upd}}}$
Initialization: $\tilde{\theta}^{(0)} = \pi_{\theta_t}$, $C_t(s,a) = 0$, for all $s, a$
Output: $\widehat{\pi}_{t+1}$ and $C_t$
for $k = 0, \cdots, K-1$ do
    for each state $s$ in the batch do
        Find $g_{s,a}^{(k)} = \frac{\partial L_{\text{Hinge}}(\theta)}{\partial \theta_{s,a}}\Big|_{\theta = \tilde{\theta}^{(k)}}$, for each $a$
        Let $w_s = (e^{-\eta g_{s,1}}, \dots, e^{-\eta g_{s,|\mathcal{A}|}})$
        $\tilde{\theta}^{(k+1)} = \frac{1}{\langle w_s, \tilde{\theta}^{(k)} \rangle}(w_s \circ \tilde{\theta}^{(k)})$
        $C_t(s,a) \leftarrow C_t(s,a) - \eta g_{s,a}^{(k)} / A_{\omega_t}(s,a)$, for
                each $a$ with $A_{\omega_t}(s,a) \neq 0$
    end for
end for
$\widehat{\pi}_{t+1} = \tilde{\theta}^{(K)}$
```

Regarding our analyses, we need assumptions about distribution density. Assumption 4 states that the distribution σ_π is sufficiently regular, which is required to analyze the neural network error. Additionally, the common theory works (Antos, Szepesvári, and Munos 2007; Farahmand, Szepesvári, and Munos 2010; Farahmand et al. 2016; Chen and Jiang 2019; Liu et al. 2019) have the concentrability assumption, we also have this common regularity condition.

Assumption 4 (Regularity of Stationary Distribution)**.** Given any state-action visitation distribution σ_π, there exists a universal upper bounding constant $c > 0$ for any weight vector $z \in \mathbb{R}^d$ and $\zeta > 0$, such that $\mathbb{E}_{\sigma_\pi}[\mathbb{1}\{|z^\top(s,a)| \leq \zeta\}|z] \leq c \cdot \zeta / \|z\|_2$ holds almost surely.

Assumption 5 (Concentrability Coefficient and Ratio)**.** Define the density ratio between the policy-induced distributions and the policies,

$$\phi_t^* = \mathbb{E}_{\tilde{\sigma}_t}\Big[\Big|\frac{d\pi^*}{d\pi_0} - \frac{d\pi_{\theta_t}}{d\pi_0}\Big|^2\Big]^{\frac{1}{2}}, \psi_t^* = \mathbb{E}_{\sigma_t}\Big[\Big|\frac{d\sigma^*}{d\sigma_t} - \frac{d\nu^*}{d\nu_t}\Big|^2\Big]^{\frac{1}{2}}, \tag{13}$$

where the above fractions are the Radon–Nikodym Derivatives. We assume that there exist $0 < \phi^*, \psi^* < \infty$ such that $\phi_t^* < \phi^*$ and $\psi_t^* < \psi^*$, for all t. Also, let $C_\infty < \infty$ be the concentrability coefficient. We assume that the density ratio between the optimal state distribution and any state distribution, i.e. $\|\nu^*/\nu\|_\infty < C_\infty$ for any ν.

5.3 Convergence Guarantee of Neural PPO-Clip

In this subsection, we present the convergence analysis of Neural PPO-Clip. Inspired by the analysis of (Liu et al. 2019), we analyze the convergence behavior of Neural PPO-Clip based on the neural networks analysis technique. Nevertheless, the analysis presents several unique technical challenges in establishing its convergence: (i) *Tight coupling between function approximation error and the clipping behavior*: The clipping mechanism can be viewed as an indicator function. The function approximation for advantage would significantly influence the value of the indicator function in a highly complex manner. As a result, handling the error between the neural approximated advantage and the true advantage serves as one major challenge in the analysis (please refer to the proof of Lemma 5 in Appendix C for more details); (ii) *Lack of a closed-form expression of policy update*: Due to the clipping function in the hinge loss objective and the iterative updates in the EMDA subroutine, the new policy does not have a simple closed-form expression. This is one salient difference between the analysis of Neural PPO-Clip and other neural algorithms (cf. (Liu et al. 2019)); (iii) *Neural networks analysis on advantage function*: Another technicality is that the advantage function requires the neural network projection and linearization properties to characterize the approximation error. However, since we use the neural network to approximate the state-action value function instead of the advantage function, it requires additional effort to establish the error bound of the advantage function (please refer to the proof of Lemma 3).

Given that we need to analyze the error between our approximation and the true function, we further define the target policy under the true advantage function $A^{\pi_{\theta_t}}$ as $\pi_{t+1}(a|s) := \bar{C}_t(s,a)A^{\pi_{\theta_t}}(s,a) + \tau_t^{-1} f_{\theta_t}(s,a)$, where $\bar{C}_t(s,a)$ is the $C_t(s,a)$ obtained under $A^{\pi_{\theta_t}}$. Moreover, all the expectations about A_ω throughout the analysis are with respect to the randomness of the neural network initialization. Below we state the min-iterate convergence rate and the sufficient condition of Neural PPO-Clip, which is also the main theorem of our paper. Throughout this section, we solely suppose Assumptions 1, 4, and 5 hold.

The central result of this paper is Theorem 2. In this theorem, $L_C(T)$ and $U_C(T)$ are functions influenced by T and determined by $\bar{C}_t$, a classifier-specific attribute. For detailed supporting lemmas and proofs, see Appendix C.

Theorem 2 (General Convergence Rate of Neural PPO–Clip)**.** *Consider the Neural PPO-Clip with the classifier satisfying the following conditions for all t,*

$$\textit{(i) } L_C(T)\cdot|A^{\pi}(s,a)| \le \bar{C}_t(s,a)\cdot|A^{\pi}(s,a)| \le U_C(T)\cdot|A^{\pi}(s,a)|, \tag{14}$$

$$\textit{(ii) } L_C(T) = \omega(T^{-1}), U_C(T) = O(T^{-1/2}). \tag{15}$$

Then, the policy sequence $\{\pi_{\theta_t}\}_{t=0}^{T}$ obtained by Neural PPO-Clip satisfies

$$\min_{0\le t\le T}\{\mathcal{L}(\pi^*) - \mathcal{L}(\pi_{\theta_t})\} \le \frac{\log|\mathcal{A}| + \sum_{t=0}^{T-1}(\varepsilon_t + \varepsilon'_t) + TU_C^2(2\psi^* + M)}{TL_C(1-\gamma)}, \tag{16}$$

where $\varepsilon_t = C_\infty \tau_{t+1}^{-1}\phi^*\epsilon_{t+1}^{1/2} + Y^{1/2}\psi^*\epsilon_t'^{1/2}$, $\varepsilon'_t = |\mathcal{A}|\cdot C_\infty\tau_{t+1}^{-2}\epsilon_{t+1}$, $M = 4\mathbb{E}_{\nu_t}[\max_a(Q_{\omega_0}(s,a))^2] + 4R_f^2$, *and* $Y = 2M + 2(R_{\max}/(1-\gamma))^2$.

To demonstrate that our convergence analysis is general for Neural PPO-Clip with various classifiers, we choose to state Theorem 2 in a general form utilizing the condition (14) and (15). Indeed, we show that (14) and (15) can be naturally satisfied by using the standard PPO-Clip classifier described in (7) in the following Corollary 1. Importantly, these conditions are not technical assumptions for our theorem. Notably, we also establish that PPO-Clip-sub (a variant of generalized PPO-Clip utilizing a distinct classifier) aligns with the result presented in Theorem 2. For a comprehensive statement and analysis, please refer to Appendix D.

Corollary 1 (Global Convergence of Neural PPO-Clip, Informal)**.** *Consider Neural PPO-Clip with the standard PPO-Clip classifier $\rho_{s,a}(\theta) - 1$ and the objective function $L^{(t)}(\theta)$ in each iteration t as*

$$\mathbb{E}_{\nu_t}[\langle\pi_{\theta_t}(\cdot|s), |A^{\pi_{\theta_t}}(s,\cdot)| \circ \ell(\mathrm{sign}(A^{\pi_{\theta_t}}(s,\cdot)), \rho_{s,\cdot}(\theta) - 1, \epsilon)\rangle]. \tag{17}$$

(i) If we specify the EMDA step size $\eta = T^{-\alpha}$ where $\alpha \in [1/2, 1)$ and the temperature parameter $\tau_t = T^{\alpha}/(Kt)$. Recall that K is the number of EMDA iterations. Let the neural networks' widths be m_f, m_Q, and the SGD and TD updates T_{upd} be configured as in Appendix D, we have

$$\min_{0\le t\le T}\{\mathcal{L}(\pi^*) - \mathcal{L}(\pi_{\theta_t})\} \le \frac{\log|\mathcal{A}| + K^2(2\psi^* + M) + O(1)}{T^{\alpha}(1-\gamma)}. \tag{18}$$

Hence, Neural PPO-Clip has $O(T^{-\alpha})$ convergence rate. (ii) Furthermore, let the $\alpha = 1/2$, we obtain the fastest convergence rate, which is $O(1/\sqrt{T})$.

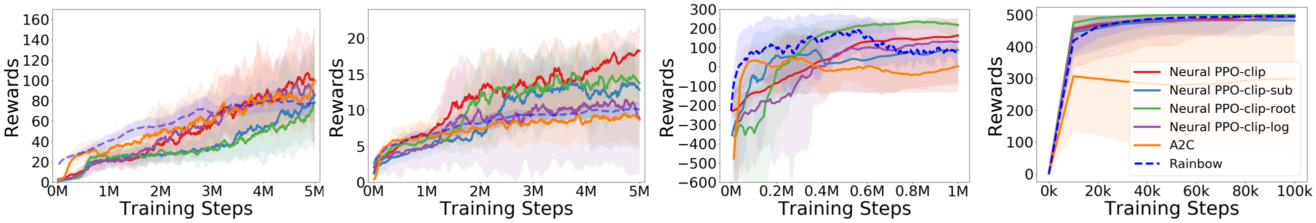
Notably, the min-iterate convergence rates presented in (16) and (18) are commonly observed in the realms of nonconvex optimization and neural network theory (Lacoste-Julien 2016; Ghadimi and Lan 2016; Liu et al. 2019), and they do not constitute stringent results. Furthermore, it is worth pointing out that in (16), the terms ε_t and ε'_t correspond to the errors introduced by policy improvement and policy evaluation, respectively. These errors can be controlled by adjusting neural network widths and the number of TD and SGD iterations T_{upd}, and they can be made arbitrarily small. Further details can be found in Appendix C.

Consequently, the convergence rate obtained by our analysis is determined by $U_C(T)^2/L_C(T)$. After a brief calculation, it becomes evident that under conditions (14) and (15), the most optimal convergence rate achievable through (16) is $O(1/\sqrt{T})$. This scenario arises when $L_C(T) = U_C(T) = O(T^{-1/2})$. This insight underscores that within our analysis, the original PPO-Clip stands as the algorithm that achieves the most favorable bound.

5.4 Understanding the Clipping Mechanism

In this subsection, we delve into the more profound understanding of the clipping mechanism.

Rationale Behind the PPO-Clip Convergence. As outlined in Section 3, the clipping mechanism establishes a connection to the hinge loss, consequently shaping the objective as (8). Notably, in the context of the original PPO-Clip, we



(a) Space Invaders (b) Breakout (c) LunarLander (d) CartPole

Figure 1: Evaluation of PPO-Clip with different classifiers and popular benchmark methods in MinAtar and OpenAI Gym.

specify the objective as follows:

$$\frac{1}{|\mathcal{D}|}\sum_{(s,a)\in\mathcal{D}} |A^{\pi}(s,a)|\,\ell(\mathrm{sign}(A^{\pi}(s,a)), \rho_{s,a}(\theta)-1,\epsilon). \tag{19}$$

We delve more deeply into this objective (19). It is important to note that if the *signs* of the advantages are incorrect, it can lead to significant errors in computing the objective value during learning. However, due to the impressive empirical performance of neural networks in approximating values, erroneous signs of advantages tend to occur mainly when $|A^{\pi}(s,a)|$ is close to zero. Moreover, when $|A^{\pi}(s,a)|$ is near zero, its contribution to the objective remains relatively insignificant. Consequently, despite incorrect signs, the objective value remains reasonably accurate. This perspective offers an explanation for the robustness and impressive empirical performance of PPO-Clip. Additionally, this notion supports the potential of PPO-Clip to achieve convergence. Furthermore, this concept is essential to comprehend the novel proof technique introduced in Lemma 5. This lemma forms the cornerstone for bounding the errors in policy improvement and evaluation. For more detailed insights, please refer to Appendix C.

Characterization of the Clipping Mechanism. Our convergence analysis reveals that clipping mechanisms solely impact the pre-constant of convergence rates. Surprisingly, our analysis and results show that the clipping range ϵ only influences the *pre-constant* of the Neural PPO-Clip convergence rate. This is unexpected since, intuitively, ϵ is considered analogous to the penalty parameter of PPO-KL (Liu et al. 2019), which directly affects convergence rates. Contrary to expectations, we discover that the EMDA step size η plays a crucial role in determining convergence rates, rather than the clipping range ϵ. This result is illustrated by the involvement of the clipping mechanism in the EMDA subroutine through the indicator functions in the gradients. Moreover, as the clipping range ϵ is contained inside the indicator function, *it only influences the number of effective EMDA updates but not the magnitude of each EMDA update*. Since we know that the convergence rate is determined by the magnitude of the gradient updates (i.e., $U_C(T), L_C(T)$, which is η-dependent and η is T-dependent), the clipping range can only affect the pre-constant of the convergence rate and the rate would still be $O(1/\sqrt{T})$. For a more comprehensive understanding, please refer to Appendices C and D.

6 Experiments

Experimental Setup. Given the convergence guarantees in Section 5.3, to better understand the empirical behavior of the generalized PPO-Clip objective, we further conduct experiments to evaluate Neural PPO-Clip with different classifiers. Specifically, we evaluate Neural PPO-Clip, Neural PPO-Clip-sub (as introduced in Section 3), and two additional classifiers, $\log(\pi_\theta(a|s)) - \log(\pi_{\theta_t}(a|s))$ and $\sqrt{\rho_{s,a}(\theta)} - 1$(termed as Neural PPO-Clip-log and Neural PPO-Clip-root), against benchmark approaches in several RL benchmark environments. Our implementations of Neural PPO-Clip are based on the RL Baseline3 Zoo framework (Raffin 2020). We test the algorithms in both MinAtar (Young and Tian 2019) and OpenAI Gym environments (Brockman et al. 2016). In addition, the algorithms are compared with popular baselines, including A2C and Rainbow. A2C follows the implementation and default settings from RL Baseline3 Zoo. For Rainbow, we adopt the configuration from (Ceron and Castro 2021). Please refer to Appendix G for more details about our experiment settings.

Variants of Neural PPO-Clip Achieves Comparable Empirical Performance. Figure 1 shows the training curves of Neural PPO-Clip with various classifiers and the benchmark methods. Notably, we observe that Neural PPO-Clip with various classifiers can achieve comparable or better performance than the baseline methods in both RL environments. To be mentioned, the performance of Rainbow is consistent with the results reported by (Ceron and Castro 2021). In summary, the outcomes depicted above underscore the practicality of the hinge loss reinterpretation of PPO-Clip within standard RL tasks. Furthermore, this approach positions classifier selection as a potential hyperparameter for the future deployment of PPO-Clip.

7 Concluding Remarks

The convergence behavior of PPO-Clip, a longstanding open problem, is addressed in this paper, providing the first convergence result and deeper insights. Our limitations are (i) analysis under discrete action space and (ii) reliance on NN error analysis, typically requiring large NN width. Despite the empirical success of PPO-Clip without this, our two-layer NN exploration suggests our results hold if approximation errors are well-managed. We anticipate this work will spark a deeper understanding of PPO-Clip within the RL community.

Acknowledgments

We thank Hsuan-Yu Yao, Kai-Chun Hu, and Liang-Chun Ouyang for the helpful discussion and for providing insightful advice regarding the experiment. This material is based upon work partially supported by the National Science and Technology Council (NSTC), Taiwan under Contract No. NSTC 112-2628-E-A49-023 and Contract No. NSTC 112-2634-F-A49-001-MBK and based upon work partially supported by the Higher Education Sprout Project of the National Yang Ming Chiao Tung University and Ministry of Education (MOE), Taiwan.

References

Agarwal, A.; Kakade, S. M.; Lee, J. D.; and Mahajan, G. 2020. Optimality and approximation with policy gradient methods in markov decision processes. In *Conference on Learning Theory*, 64–66. PMLR.

Antos, A.; Szepesvári, C.; and Munos, R. 2007. Fitted Q-iteration in continuous action-space MDPs. *Advances in neural information processing systems*, 20.

Beck, A.; and Teboulle, M. 2003. Mirror descent and nonlinear projected subgradient methods for convex optimization. *Operations Research Letters*, 31(3): 167–175.

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

Brockman, G.; Cheung, V.; Pettersson, L.; Schneider, J.; Schulman, J.; Tang, J.; and Zaremba, W. 2016. Openai gym. *arXiv preprint arXiv:1606.01540*.

Byun, J.-S.; Kim, B.; and Wang, H. 2020. Proximal Policy Gradient: PPO with Policy Gradient. *arXiv preprint arXiv:2010.09933*.

Ceron, J. S. O.; and Castro, P. S. 2021. Revisiting rainbow: Promoting more insightful and inclusive deep reinforcement learning research. In *International Conference on Machine Learning*, 1373–1383. PMLR.

Chen, G.; Peng, Y.; and Zhang, M. 2018. An adaptive clipping approach for proximal policy optimization. *arXiv preprint arXiv:1804.06461*.

Chen, J.; and Jiang, N. 2019. Information-theoretic considerations in batch reinforcement learning. In *International Conference on Machine Learning*, 1042–1051. PMLR.

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(1): 4809–4874.

Farahmand, A.-m.; Szepesvári, C.; and Munos, R. 2010. Error propagation for approximate policy and value iteration. *Advances in Neural Information Processing Systems*, 23.

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*, 1467–1476. PMLR.

Ghadimi, S.; and Lan, G. 2016. Accelerated gradient methods for nonconvex nonlinear and stochastic programming. *Mathematical Programming*, 156(1-2): 59–99.

Hu, K.-C.; Hsieh, P.-C.; Wei, T. H.; and Wu, I.-C. 2020. Rethinking Deep Policy Gradients via State-Wise Policy Improvement. In *”I Can’t Believe It’s Not Better!”NeurIPS 2020 workshop*.

Lacoste-Julien, S. 2016. Convergence rate of frank-wolfe for non-convex objectives. *arXiv preprint arXiv:1607.00345*.

Liu, B.; Cai, Q.; Yang, Z.; and Wang, Z. 2019. Neural trust region/proximal policy optimization attains globally optimal policy. *Advances in Neural Information Processing Systems*, 32: 10565–10576.

Liu, Y.; Zhang, K.; Basar, T.; and Yin, W. 2020. An improved analysis of (variance-reduced) policy gradient and natural policy gradient methods. *Advances in Neural Information Processing Systems*, 33: 7624–7636.

Mei, J.; Xiao, C.; Szepesvari, C.; and Schuurmans, D. 2020. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, 6820–6829.

Pi, C.-H.; Hu, K.-C.; Cheng, S.; and Wu, I.-C. 2020. Low-level autonomous control and tracking of quadrotor using reinforcement learning. *Control Engineering Practice*, 95: 104222.

Raffin, A. 2020. RL Baselines 3 Zoo. https://github.com/DLR-RM/rl-baselines3-zoo. Accessed: 2024-01-19.

Schulman, J.; Levine, S.; Abbeel, P.; Jordan, M.; and Moritz, P. 2015. Trust region policy optimization. In *International conference on machine learning*, 1889–1897. PMLR.

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 *AAAI Conference on Artificial Intelligence*, volume 34, 5668–5675.

Singh, S.; Jaakkola, T.; Littman, M. L.; and Szepesvári, C. 2000. Convergence results for single-step on-policy reinforcement-learning algorithms. *Machine learning*, 38(3): 287–308.

Wang, W.; Han, J.; Yang, Z.; and Wang, Z. 2021. Global convergence of policy gradient for linear-quadratic mean-field control/game in continuous time. In *International Conference on Machine Learning*, 10772–10782. PMLR.

Ye, D.; Liu, Z.; Sun, M.; Shi, B.; Zhao, P.; Wu, H.; Yu, H.; Yang, S.; Wu, X.; Guo, Q.; et al. 2020. Mastering complex control in moba games with deep reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, 6672–6679.

Young, K.; and Tian, T. 2019. MinAtar: An Atari-Inspired Testbed for Thorough and Reproducible Reinforcement Learning Experiments. *arXiv preprint arXiv:1903.03176*.