Single-Loop Federated Actor-Critic across Heterogeneous Environments

Ye Zhu, Xiaowen Gong

Auburn University, Auburn, AL, USA
{yzz0211,xzg0017}@auburn.edu

Abstract

Federated reinforcement learning (FRL) has emerged as a promising paradigm, enabling multiple agents to collaborate and learn a shared policy adaptable across heterogeneous environments. Among the various reinforcement learning (RL) algorithms, the actor-critic (AC) algorithm stands out for its low variance and high sample efficiency. However, little to nothing is known theoretically about AC in a federated manner, especially each agent interacts with a potentially different environment. The lack of such results is attributed to various technical challenges: a two-level structure illustrating the coupling effect between the actor and the critic, heterogeneous environments, Markovian sampling and multiple local updates. In response, we study Single-Loop Federated Actor Critic (SFAC) where agents perform AC learning in a two-level federated manner while interacting with heterogeneous environments. We then provide bounds on the convergence error of SFAC. The results show that the convergence error asymptotically converges to a near-stationary point, with the extent proportional to environment heterogeneity. Moreover, the sample complexity exhibits a linear speed-up through the federation of agents. We evaluate the performance of SFAC through numerical experiments using common RL benchmarks, which demonstrate its effectiveness.

Introduction

In single-agent reinforcement learning (RL), policy improvement can be effectively facilitated by the Policy Gradient Theorem (Sutton et al. 1999) when using parameterized policies. This theorem represents the policy gradient as a product of the score function and the action-value function, serving as the cornerstone for numerous RL algorithms. Among these, actor-critic (AC) algorithms stand out by employing temporal difference (TD) learning to approximate the action-value function, unlike Monte-Carlo algorithms (Williams 1992), which rely on sampling entire trajectories to estimate the value function. AC algorithms not only make better use of available samples but also significantly reduce the variance in policy updates, leading to more stable and efficient learning.

In AC algorithms, two primary methods are commonly used: double-loop variants (Kumar, Koppel, and Ribeiro 2023; Xu, Wang, and Liang 2020) and single-loop variants (Qiu et al. 2021; Chen, Sun, and Yin 2021; Hong et al. 2023; Chen and Zhao 2023). In non-asymptotic analysis of AC, the double-loop variants facilitate a decoupled convergence analysis of the critic and the actor, involving a policy evaluation sub-problem in the inner loop and a perturbed gradient descent in the outer loop. While the double-loop variant simplifies the convergence analysis, it is rarely adopted in practice due to its requirement for accurate critic estimation (Kumar, Koppel, and Ribeiro 2023), making it typically sample-inefficient. The key distinction between double-loop and single-loop variants lies in their methods for achieving convergence. Double-loop AC requires a sufficient number of iterations for value evaluation in the inner loop to ensure convergence. In contrast, the single-loop method eliminates this requirement by preserving the critic's memory throughout the process, providing a significant advantage in terms of efficiency.

Federated reinforcement learning (FRL) is an emerging distributed learning framework that integrates the key concepts of federated learning (FL) and RL (Fan et al. 2021; Khodadadian et al. 2022; Xie and Song 2023; Zhu and Gong 2023). Its growing prominence is driven by its versatility in real-world applications such as autonomous driving (Shalev-Shwartz, Shammah, and Shashua 2016; Kiran et al. 2021) and resource allocation (Ye, Li, and Juang 2019; Yu et al. 2020). Existing FRL methods, such as FedSARSA (Zhang et al. 2024) and FedTD (Khodadadian et al. 2022; Wang et al. 2023), are widely applied to stochastic problems with single-level structure. Instead, FRL with nested formulations, has not been studied. In this paper, we propose single-loop federated actor critic (SFAC), where agents in heterogeneous environments perform AC learning in a two-level federated manner. Specifically, SFAC is composed of federated critics (FedC) where critics perform heterogeneous federated TD learning for policy evaluation by aggregating their local value function models, and federated actors (FedA) where the actors perform federated policy update for policy improvement by aggregating their local policy gradients. Through local information aggregation, SFAC enables the learning of a shared policy that achieves strong average performance across all agents' heterogeneous environments.

In addition to common challenges in FRL, such as Markovian sampling and multiple local iterations, the conver-

References	Number of Agents	Heterogeneous Environments	Target Environment	Markovian Sampling	Linear Speedup
(Chen, Sun, and Yin 2021)	1	–	–	×	–
(Chen and Zhao 2023)	1	–	–	√	–
(Shen et al. 2023)	N	×	Individual	√	IID setting
This paper	N	√	Mixture	√	√

Table 1: Comparison of Settings and Results with the Most Related Works: "Target environment" is the benchmark for evaluating FRL; "Individual" indicates the target environment is identical for all agents; "IID setting" represents the linear speedup result is only established under the IID setting.

gence analysis of SFAC encounters several unique difficulties stemming from its distinctive features. First, two levels of local model aggregations among agents' critics and actors have coupled and non-trivial impacts on the convergence error of SFAC. Second, single-loop structure brings the analysis of SFAC substantial errors in critic estimation and the tight coupling between parallel updates of critics and actors, making the algorithm more susceptible to unstable error propagation. Third, due to the environment heterogeneity, agents in SFAC have distinct optimal value functions for a given policy and also different optimal policies, causing non-trivial biases in the convergence analysis. Lastly, to measure the performance of the common policy obtained by SFAC for all agents' environments, we focus on the *mixture* environment which is *randomly drawn* from agents' heterogeneous environments. Compared to the virtual environment considered in recent works on FRL (Wang et al. 2023; Zhang et al. 2024; Jin et al. 2022) (which is defined by directly averaging transition probability kernels and reward functions of agents' environments), the mixture environment presents a more meaningful yet significantly more complex challenge for convergence analysis.

The main contributions of this paper are summarized as follows:

- **SFAC algorithm.** We devise SFAC where agents perform AC learning across heterogeneous environments in a two-level federated manner. The two-level structure introduces a non-trivial coupling between parallel updates of critics and actors. Furthermore, SFAC allows agents to perform different number of local iterations for TD learning.
- **SFAC analysis.** We develop a new analysis framework that establishes the finite-time convergence for SFAC. Technically, we exploit the property of the global gradient of the average mean-squared projected Bellman error (MSPBE) in the critic's error, so that a biased term in the critics can be eliminated in the actors. The results show that the convergence error is asymptotically determined by the heterogeneity of agents' environments, and it diminishes to 0 as the environments heterogeneity reduces to 0. Moreover, the sample complexity enjoys linear speed-up through federation.

Related Works

Actor-Critic algorithms. Most existing theoretical studies on AC focus on the single-agent setting (Kumar, Koppel, and Ribeiro 2023; Wang et al. 2019; Qiu et al. 2021; Xu, Wang, and Liang 2020). Even within multi-agent AC frameworks, the problem is frequently approached as if it were a single-agent AC issue, viewed from a global perspective. For double-loop AC, the non-asymptotic analyses have been well established in both IID sampling and Markovian sampling. For single-loop AC, through the lens of bi-level optimization, two-timescale AC (Hong et al. 2023) and single-timescale AC (Chen and Zhao 2023; Chen, Sun, and Yin 2021) have been proposed, achieving sample complexities of $\tilde{O}(\epsilon^{-2.5})$ and $\tilde{O}(\epsilon^{-2})$ respectively. The most closely related work to our paper that multiple agents executes in independent environments and collectively seek a global policy using AC algorithms is (Shen et al. 2023). However, this algorithm is executed in homogeneous environments and the architecture is the shared memory that is accessible to all agents. Besides, (Shen et al. 2023) only established the linear speedup result under IID sampling. [see Table 1 for details].

Federated bilevel optimization. Since AC algorithm with linear value function approximation can be viewed as a special case of bilevel algorithms (Ghadimi and Wang 2018; Chen, Sun, and Yin 2021; Hong et al. 2023), it is worth noting that SFAC is not a special case of federated bilevel learning (FBO). In FBO (Tarzanagh et al. 2022; Huang, Zhang, and Ji 2023; Yang, Xiao, and Ji 2023; Li et al. 2024), the optimization problem is

$$\min_{x\in\mathbb{R}^{d_1}} f(x, y^*(x)) := \frac{1}{m}\sum_{i=1}^{m} f_i(x, y^*(x)) \tag{1a}$$

$$s.t.\ \ y^*(x) = \operatorname*{argmin}_{y\in\mathbb{R}^{d_2}} g(x, y) := \frac{1}{m}\sum_{i=1}^{m} g_i(x, y) \tag{1b}$$

Note that the upper-level optimization in FBO is performed on $f_i(x, y^*(x))$ while the upper-level optimization in FAC is performed on $f_i(x, y_i^*(x))$. Furthermore, $g(x, y)$ has a special definition in FRL and needs to be addressed carefully.

Federated reinforcement learning. FRL differs from distributed reinforcement learning (Zhang et al. 2018; Zhang, Yang, and Başar 2021) in several key ways: 1) agents in FRL interact with heterogeneous environments and follow their respective MDPs; 2) the architecture of agents in FRL is worker-server, with a central server that coordinates with N agents; 3) FRL involves some unique features of FL, including multiple local iterations of agents, heterogeneous and time-varying computation capabilities of agents.

There have been a few recent works on FRL (Zhang et al. 2024; Wang et al. 2023; Jin et al. 2022). (Wang et al. 2023) considered the federated policy evaluation; (Zhang et al. 2024; Jin et al. 2022) analyzed the federated action value iteration. However, none of them has established the convergence of FRL algorithms by considering the collaboration of policy evaluation and policy improvement simultaneously as a two-level collaboration. Furthermore, these three papers aims to solve optimization problem in a constructed virtual environment. Thus, these works for the federation of (action) value functions do not apply to our setting. Specifically, it remains unclear how the federation of critics affect the convergence performance of the optimal policy for the mixture environment considered in this paper. In this paper, we demonstrate that the proposed algorithm can asymptotically produce a near-stationary point of the mixture environment, marking the *first* such result in the existing FRL literature.

Preliminaries

MDP in Heterogeneous Environments

The discounted Markov decision process (MDP) for agent i is defined as $M_i \triangleq \{S, A, P_i, R_i, \gamma\}$, where S denotes the set of states, A represents the set of actions, P_i is the transition kernel at agent i, R_i is the reward function at agent i, and γ is the discount factor. In this paper, while all agents share the same state-action space, their reward functions and transition kernels may vary. Specifically, when agent i takes an action $a \in A$ at state $s \in S$, it transitions to the next state $s' \in S$ based on the probability distribution $P_i(s'\,|s, a\,)$ and receives reward $r_i(s, a, s')$. We consider a stochastic policy π that assigns a probability distribution $\pi(\cdot\,|s\,)$ over the entire action space A for each state $s \in S$.

Given policy π, the state value function and the state-action value function for agent i are defined as follows:

$$V_\pi^i(s) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t^i, a_t^i, s_{t+1}^i)\,\big|s_0^i = s, \pi, P_i\right]$$

$$Q_\pi^i(s,a) = \mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_i(s_t^i, a_t^i, s_{t+1}^i)\,|s_0^i = s, a_0^i = a, \pi, P_i\right]$$

where the expectation $\mathbb{E}$ is taken over all possible trajectories of agent i following policy π. Let b_i represent the initial state distribution of agent i. The discounted state visitation measure induced by policy π is then defined as $\nu_\pi^i(s) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t P_i(s_t^i = s\,\big|s_0^i \sim b_i, \pi,\, P_i)$ and the state-action visitation distribution is defined as $\nu_\pi^i(s,a) = (1-\gamma)\sum_{t=0}^{\infty}\gamma^t P_i(s_t^i = s, a_t^i = a\,\big|s_0^i \sim b_i, \pi,\, P_i)$.

The performance of policy π at agent i can be evaluated by the expected cumulative rewards as $J_i(\pi) = (1-\gamma)\mathbb{E}_{s\sim\eta_i}\left[V_\pi^i(s)\right]$. In the context of single-agent RL, the objective is to find an optimal policy that maximizes $J_i(\pi)$.

Optimal Policy for Mixture Environment

In this paper, we tackle a federated reinforcement learning problem involving N agents who collaboratively work together to find a globally optimal policy without sharing their raw collected samples. Each agent operates within its own unique environment, with each environment defined by its own MDP. Previous studies on FRL (Jin et al. 2022; Wang et al. 2023; Zhang et al. 2024) have focused on optimizing the value function model for a single virtual environment. This virtual environment is constructed as an MDP $\bar{M} \triangleq \{S, A, \bar{P}, \bar{R}, \gamma\}$ by directly averaging the transition kernels and reward functions across all agents' environments, given by $\bar{P} = \frac{1}{N}\sum_{i=1}^N P_i$ and $\bar{R} = \frac{1}{N}\sum_{i=1}^N R_i$.

However, such an "averaged" environment may not correspond to any agent's specific environment. Consequently, from the perspective of an individual agent, the objective function may not effectively incentivize participation in the federation. Motivated by this observation, we propose considering a *mixture* environment in this paper, defined as an environment *randomly drawn* from the set of agents' heterogeneous environments. This mixture environment differs from the virtual environment described in (Jin et al. 2022; Wang et al. 2023; Zhang et al. 2024). Specifically, the virtual environment defines an MDP where, after an agent selects an environment and conducts a transition, transitioning to the successive state still demands a new selection of an environment. In contrast, in the mixture environment, once an environment is chosen with a certain probability, the state transitions follow the corresponding MDP of that environment. Compared to the virtual environment, the mixture environment is more practical.

Therefore, the goal of agents in FRL is to cooperatively find an optimal policy θ^* that maximizes the total cumulative discounted rewards,

$$\theta^* = \arg\max_\theta J(\theta) = \arg\max_\theta \frac{1}{N}\sum_{i=1}^N J_i(\theta), \qquad (2)$$

where $J_i(\theta)$ is a non-concave objective function. We use π_θ to denote the policy parameterized by $\theta \in \mathbb{R}^d$. With no ambiguity, we use θ to denote the policy π_θ.

Treating each environment as an independent RL problem implies that each agent aims to maximize their own $J_i(\theta)$. However, our goal is to develop a single policy that optimally balances performance across all environments, while preserving the unique characteristics of each agent's environment. This objective aligns with the principles of federated supervised learning.

Actor-Critic with Function Approximation

In this subsection, we will provide a detailed introduction to AC with function approximation.

Critic update. In AC, policy evaluation (performed by critics) and policy improvement (performed by actors) are alternately applied to achieve the optimal policy. A common approach in policy evaluation is to use linear function approximation. Let $\{\Phi_k\}_{k=1}^d$ be a set of d linearly independent basis vectors in $\mathbb{R}^n$. The true value function V_π is then approximated as $V_\pi(s) \approx V_\omega(s) = \phi(s)^{\mathrm{T}}\omega$, where $\phi(s) \in \mathbb{R}^d$ is a fixed feature vector for state s and $\omega \in \mathbb{R}^d$ is the unknown model to be learned. For convenience, we drop the subscript (i) since each agent follows TD learning to update

its local value function. For an observed tuple (s, r, s'), the TD operator is defined as follows, for any $V \in \mathbb{R}^{|S|}$,

$$(T_\pi V)(s) = R_\pi(s) + \gamma \sum_{s'} P_\pi(s, s')V(s'). \quad (3)$$

Then the value function satisfies the Bellman equation $T_\pi V_\pi = V_\pi$.

The task of finding ω^* is usually addressed by TD learning. The loss function is defined as the squared Bellman error (Bhandari, Russo, and Singal 2018). At time t, an agent observes sample $O_t = (s_t, r_t, s_{t+1})$ and the negative stochastic semi-gradient of the Bellman error evaluated at the current parameter ω_t is expressed as

$$\hat{g}(\omega_t) = (r_t + \gamma\phi(s_{t+1})^{\mathrm{T}}\omega_t - \phi(s_t)^{\mathrm{T}}\omega_t)\phi(s_t). \quad (4)$$

Then the estimated model at time $t + 1$ can be updated by the gradient descent method (Bhandari, Russo, and Singal 2018) with step size β as

$$\omega_{t+1} = \omega_t + \beta\hat{g}(\omega_t). \quad (5)$$

Actor update. Policy improvement using policy function approximation is typically performed through the policy gradient theorem. For agent i, the gradient $\nabla J_i(\theta)$ is derived by the policy gradient method (Sutton et al. 1999) as follows:

$$\nabla J_i(\theta) = \mathbb{E}_{v_\theta^i}\left[Q_\theta^i(s, a)\psi_\theta(s, a)\right] = \mathbb{E}_{v_\theta^i}\left[A_\theta^i(s, a)\psi_\theta(s, a)\right]$$

where $\psi_\theta(s, a) = \nabla_\theta \log \pi_\theta(a\,|\,s)$ denotes the score function and v_θ^i denotes the state visitation measure under policy π_θ at agent i. Advantage function $A_\theta^i(s, a)$ is approximated by temporal difference error $\delta_{\omega_\theta}^i(s, a, s') = r_i(s, a, s') + \gamma V_{\omega_\theta}^i(s') - V_{\omega_\theta}^i(s)$ where $V_{\omega_\theta}^i$ is agent i's critic estimate for policy π_θ. The expected policy gradient of agent i at round k using a mini-batch can then be estimated as

$$\hat{h}_k^i(\omega_\theta, \theta) = \frac{1}{M}\sum_{m=1}^{M} \delta_{\omega_\theta}^i(s_{k,m}^i, a_{k,m}^i, s_{k,m+1}^i)\psi_\theta(s_{k,m}^i, a_{k,m}^i)$$

Then agent i updates its local policy by the gradient ascent method with step size α as

$$\theta_{k+1}^i = \theta_k^i + \alpha\hat{h}_k^i(\omega_\theta, \theta) \quad (6)$$

Single-Loop Federated Actor-Critic

In this section, we describe the single-loop federated reinforcement learning method in heterogeneous environments where the critics perform federated TD learning for policy evaluation and the actors perform federated policy update for policy improvement.

First, we design a single-loop process for FAC as shown in Algorithm 1. Namely, the outer loop consists of actor's updates for policy π to optimize the global policy, followed by an entire inner loop of critics' updates.

Inner optimizer: FedC. We first define the global optimal value function in the mixture environment. In the process of policy evaluation for agent i, the local loss function F_i is

Algorithm 1: Single-Loop Federated Actor Critic (SFAC)

Input: number of rounds K, stepsize α_k, β_k, initial actor parameter θ_0
for $k = 1$ to K **do**
  $\omega_{k,0} = \omega_t$
  for $t = 0$ to $T - 1$ **do**
    $\omega_{k,t+1} = \textbf{FedC}(\theta_t, \omega_{k,t}, \beta_k)$
  end for
  $\omega_{k+1} = \omega_{k,T}$
  $\theta_{k+1} = \textbf{FedA}(\theta_k, \omega_{k+1}, \alpha_k)$
end for
Output: $\theta_{\hat{K}}$ with $\hat{K}$ chosen uniformly from $\{1, \ldots, K\}$

usually defined as expected Bellman error squared (Bhandari, Russo, and Singal 2018; Srikant and Ying 2019). Accordingly, the optimization problem for federated value evaluation can be formulated as

$$\min_{\omega\in\mathbb{R}^d}\left[F(\omega) = \frac{1}{N}\sum_{i=1}^{N} F_i(\omega)\right] \quad (7)$$

where $F_i(\omega) = \mathbb{E}_{O_k\sim D_i}\left[\frac{1}{2}(r_k + \gamma V_\omega(s_{k+1}) - V_\omega(s_k))^2\right]$ is the local objective function of i-th agent, i.e., the expected squared Bellman error with respect to the model parameter ω and D_i is the stationary distribution of the associated state transition Markov chain in i-th environment.

The global objective function of our federated TD learning problem is defined as the *average MSPBE* across all agents, where the MSPBE serves as a measure of the error in the value function model for each agent's respective environment. To minimize the average MSPBE, we devise the FedC algorithm which updates the value function model via federated TD learning. FedC aims to iteratively estimate the gradient of the global objective function (i.e., the average MSPBE) as

$$g(\omega) = \frac{1}{N}\sum_{i=1}^{N} g_i(\omega) \quad (8)$$

where $g_i(\omega)$ is the expected gradient of agent i's local objective function (i.e., the MSPBE for agent i's environment). The optimal value function model ω^* that minimizes the average MSPBE of agents satisfies $g(\omega^*) = 0$. Note that the gradient in TD learning is different from that of the standard gradient descent, as $g_i(\omega)$ or $g(\omega)$ is not the gradient of any *fixed* objective function.

For FedC, in communication round t of critics, each agent performs υ_i local iterations to approximate the value function of the given policy using its local observations as Algorithm 2 shows. υ_i may vary across agents since agents have *heterogeneous* computation capabilities. Then agent i sends the local value model ω_{t,υ_i}^i to the global critic server. In round $t + 1$, the global critic server aggregates local critics' models as

$$\theta_{t+1} = \Pi_{2,\mathcal{H}}\left(\theta_t + \alpha\left(\frac{1}{N}\sum_{i=1}^{N}\upsilon_i\right)\cdot\frac{1}{N}\sum_{i=1}^{N} d_t^i\right) \quad (9)$$

Algorithm 2: Federated Critic (FedC)

Input: $\omega_{k,t}$, stepsize β_k
$\omega^i_{k,t} = \omega_{k,t}$ for agent $i \in \mathcal{N}$
for $i \in \mathcal{N}$ **in parallel do**
  for $\upsilon = 0$ to $\upsilon_i - 1$ **do**
    Observe tuple $O^i_{k,t,\upsilon} = (s^i_{k,t,\upsilon}, r^i_{k,t,\upsilon}, s^i_{k,t,\upsilon+1})$ and calculate the gradient by (4)
    Update the local model $\omega^i_{k,t,\upsilon+1} = \omega^i_{k,t,\upsilon} + \beta_i \hat{g}_i(\omega^i_{k,t,\upsilon})$
  end for
end for
Server computes the global model by (9)
Output: $\omega_{k,t+1}$

where υ_i is the number of local updates at agent i and d^i_t is the normalized gradient for agent i in the t-th round as $d^i_t = \frac{1}{\upsilon_i} \sum_{k=0}^{\upsilon_i - 1} \hat{g}_i(\omega^i_{t,\upsilon})$. Here we consider agents have heterogeneous number of local updates while the number of local updates are identical and fixed in the previous work (Wang et al. 2023; Zhang et al. 2024; Jin et al. 2022). Note that the cumulative local gradients are normalized by averaging, and this is necessary when dealing with heterogeneous number of local updates. Besides, we use $\Pi_{2,\mathcal{H}}(\cdot)$ to denote the standard Euclidean projection on to a convex compact subset $\mathcal{H} \subset \mathbb{R}^d$ that is assumed to contain ω^*. Such a projection step is commonly adopted in RL (Bhandari, Russo, and Singal 2018; Doan, Maguluri, and Romberg 2019; Wang et al. 2023).

Outer optimizer: FedA. For Markovian sampling, we maintain separate Markov chains for actors and critics. For critics, samples are generated following the transition kernel P_i, while the actor's chain can be viewed as evolving under transition kernel $\hat{P}_i = \gamma P_i + (1-\gamma)\eta_i$. This separate sampling protocol for the actor and critic is essential; otherwise, using the same samples for both will introduce a non-diminishing bias (Shen et al. 2023). The global actor server aggregates local actors' policies as

$$\theta_{k+1} = \theta_k + \alpha_k \cdot \frac{1}{N} \sum_{i=1}^{N} \hat{h}^i_k(\omega_k, \theta_k) \tag{10}$$

where $\omega_k = \omega_{k,T}$; T is the number of communication rounds for FedC when evaluating policy π_{θ_k}.

Algorithm summary. In each outer loop $k \in \{1, \ldots, K\}$, the actor server first broadcasts the global policy π_θ to all agents. In inner-loop communication round t, each critic $i \in \{1, \ldots, N\}$ independently performs υ_i local iterations to approximate the value function of π_θ in their respective environments, as shown in Algorithm 2. Specifically, following policy π_θ, agent i observes the tuple $O^i_{k,t,\upsilon} = (s^i_{k,t,\upsilon}, r^i_{k,t,\upsilon}, s^i_{k,t,\upsilon+1})$ at local iteration υ of round t which is generated by its own MDP characterized by $\{S, A, P_i, R_i, \gamma\}$. Using observation $O^i_{k,t,\upsilon}$, agent i computes the stochastic gradient and update its local model. At the end of each round, agents send the gradients directly to the critic server which then aggregates the gradients, updates the global critic model and starts round $t+1$ of federation. After T rounds, the global critic parameter is sent to each actor. Then actors approximates the advantage function $A^i_{\pi_\theta}$ by the temporal difference error. The policy gradient can then be estimated as $\hat{h}^i_t(\omega_\theta, \theta) = \delta^i_{\omega_\theta}(s, a, s')\psi_\theta(s, a)$ as Algorithm 3 shows. Moreover, we employ Markovian mini-batch sampling to estimate the policy gradient, which helps reduce the variance in policy gradient estimation. Then the server aggregates local policy updates as a global policy and broadcasts it to all critics.

Algorithm 3: Federated Actor (FedA)

Input: $\omega_{k,T}$, stepsize α_k,
for $i \in \mathcal{N}$ **in parallel do**
  for $m = 0$ to $M - 1$ **do**
    Observe tuple $O^i_{k,m} = (s^i_{k,m}, a^i_{k,m}, r^i_{k,m}, s^i_{k,m+1})$ and estimate the advantage function
  end for
  Update the local policy by (6)
end for
Server computes the global policy by (10)
Output: θ_{k+1}

Convergence Analysis for SFAC

In this section, we provide the convergence results of SFAC. To begin with, we make the following assumptions, which are commonly imposed in reinforcement learning and federated reinforcement learning (Fan et al. 2021; Wang et al. 2023; Khodadadian et al. 2022; Zeng et al. 2021; Xu, Wang, and Liang 2020).

Assumption 1. (Bounded Gradient Heterogeneity) *For any set of weights satisfying convex combination, i.e.,* $\{p_i \geq 0\}_{i=1}^N$ *and* $\sum_{i=1}^N p_i = 1$*, there exist constants* $\chi^2 \geq 1$*,* $\kappa^2 \geq 0$ *such that* $\sum_i p_i \|g_i(\omega)\|_2^2 \leq \chi^2 \|\sum_i p_i g_i(\omega)\|_2^2 + \kappa^2$*. If agents are in identical environments, then* $\chi^2 = 1$*,* $\kappa^2 = 0$*.*

Assumption 1 is commonly used in the federated learning literature to capture the dissimilarities of local objectives (Yang, Xiao, and Ji 2023; Wang et al. 2020).

Assumption 2. *For any given state-action pair* (s, a) *and two policies* θ*,* $\theta' \in \mathbb{R}^d$*, the following inequalities always hold:* $i) \|\psi_\theta(s,a) - \psi_{\theta'}(s,a)\|_2 \leq L_\psi \|\theta - \theta'\|_2$*;* $ii) \|\psi_\theta(s,a)\|_2 \leq C_\psi$*;* $iii) |\pi_\theta(a|s) - \pi_{\theta'}(a|s)| \leq L_\pi \|\theta - \theta'\|_2$*.*

Assumption 2 holds for various commonly used policy classes(Konda and Borkar 1999; Doya 2000).

Assumption 3. (Ergodicity) *For each agent* $i \in [N]$*, the Markov chain induced by policy* π_θ*, corresponding to the state transition matrix* $P^i_{\pi_\theta}$*, is aperiodic and irreducible. Then the geometric mixing property of the associated Markov chains is*

$$\sup \left\| P^i_{\pi_\theta}(s^i_t \in \cdot \,|\, s^i_0) - D^i_{\pi_\theta}(\cdot) \right\|_{TV} \leq \eta_i \rho^t_i \tag{11}$$

where $D^i_{\pi_\theta}(\cdot)$ *is the stationary distribution of MDP* i *following policy* π_θ*;* $\eta_i > 0$ *and* $\rho_i \in [0, 1]$ *for all* $i \in \mathcal{N}$*.*

Assumption 3 is a standard assumption (Bhandari, Russo, and Singal 2018; Xu, Wang, and Liang 2020; Khodadadian et al. 2022; Zhang et al. 2024) which holds for any uniformly ergodic Markov chain with a general state space.

Now we are ready to present the main results of SFAC. First, we characterize the performance of the aggregated critics' model. The global objective function of our federated TD learning problem is the *average MSPBE* of all agents for their respective environments, as the MSPBE quantifies the error of a value function model for an environment.

Proposition 1. For policy π_{θ_k}, T represents the number of communication rounds for critics' federation. Consider FedC shown in Algorithm 2, assuming Assumptions 1 and 3 hold, we have:

$$\mathbb{E}\left\|\omega_{k,T}-\omega^*_{\theta_k}\right\|_2^2 \leq \left(1-\frac{\beta_k\overline{\upsilon}\lambda}{4}\right)\left\|\omega_{k,T-1}-\omega^*_{\theta_k}\right\|_2^2 + C_1\beta_k^4 + C_2\beta_k^3 + C_3\frac{\beta_k^2}{N} + C_4\beta_k^2$$

where λ, C_1, C_2, C_3 and C_4 are positive, problem-dependent constants, and the detailed proof can be found in the extended version (Zhu and Gong 2024). Note that when the heterogeneity level $\kappa^2 = 0$, C_4 will be zero.

Remark 1. Proposition 1 provides a convergence error bound for $\left\|\omega_{k,T}-\omega^*_{\theta_k}\right\|_2^2$, where $\omega_{k,T}$ is the global critic parameter. From Proposition 1, we can observe that the second and the third terms represent higher-order terms of step size β_k, which are negligible, compared to other terms. The fourth term captures the effect of noise, with its variance reduced by a factor of N (linear speedup) due to collaboration among the agents. The fifth term describes the environmental heterogeneity.

Then we present the convergence results of SFAC as follows.

Theorem 1. Consider SFAC, assuming Assumptions 1 to 3 hold, if we select $\alpha_k = O\left(\sqrt{\frac{N}{K}}\right)$, $\beta_k = O\left(\sqrt{\frac{N}{K}}\right)$, the output of Algorithm 1 satisfies:

$$\frac{1}{K}\sum_{k=1}^{K}\mathbb{E}\left\|\nabla J(\theta_k)\right\|_2^2 \leq \frac{2V_1}{\alpha_k K} + \frac{40H^2\left(1+(\eta-1)\rho\right)}{M(1-\rho)} + \frac{40\kappa^2}{c^2} + 5\xi_{critic} + 2TC(\chi)C(\alpha_k)\left(C_1\beta_k^2 + C_2\beta_k + \frac{C_3}{N} + C_4\right)\frac{\beta_k^2}{\alpha_k}$$

where we define the approximation error introduced by critics as $\xi_{critic} = \max_{i,\theta}\mathbb{E}_{v^i_\theta}\left|V^i_{\pi_\theta}(s) - V^i_{\omega^{i*}_\theta}(s)\right|^2$.

Remark 2. Theorem 1 provides a bound on the convergence error of SFAC. The first term comes from the non-convex setting. The second term is the variance term which is controlled by the sample sizes of actors. The third term is determined by the heterogeneity level of environments. The last term shows how critics' error impact on the convergence bound. When the heterogeneity level equals to zero, the major term in the error bound $\mathcal{O}\left(\frac{1}{\sqrt{NK}}\right)$ shows a linear speedup.

Proof Sketch of SFAC

In this subsection, we present the key steps of the proof, and highlight the key technical differences in the convergence analysis of SFAC, compared with the previous works.

In the convergence analysis, in order to study $\mathbb{E}\left\|\nabla J\left(\theta_k\right)\right\|_2^2$, we focus on analyzing the total variance term $\frac{1}{N}\sum_{i=1}^{N}\hat{h}^i_k(\omega_{k+1},\theta_k) - \nabla J(\theta_k)$ in (12) by five terms. The 1st term is an error introduced by the inaccurate estimations of the lower level. This term was directly bounded to zero under both the double-loop setting and the two-timescale setting due to their particular algorithm design, to enable a decoupled analysis. We make them as an extension of this paper later. The 2nd term "local variance" is a noise term introduced by Markovian sampling. This bias reduces to 0 under IID sampling after taking the expectation. The 3rd term is due to environment heterogeneity, which causes an non-vanishing term. The 4th term tracks the difference between the drifting critic targets, which is crucial to eliminate the error of critics. The 5th term is from the approximation error of the critics.

$$\begin{aligned}
&\left\|\frac{1}{N}\sum_{i=1}^{N}\hat{h}^i_k(\omega_{k+1},\theta_k) - \nabla J(\theta_k)\right\|_2^2 \\
&= \Bigg\|\frac{1}{N}\sum_{i=1}^{N}\underbrace{\hat{h}^i_k(\omega_{k+1},\theta_k) - \hat{h}^i_k(\omega^*_{k+1},\theta_k)}_{\text{error of lower level}} \\
&+ \underbrace{\hat{h}^i_k(\omega^*_{k+1},\theta_k) - h^i_k(\omega^*_{k+1},\theta_k)}_{\text{local variance}} + \underbrace{h^i_k(\omega^*_{k+1},\theta_k) - h^i_k(\omega^{i*}_{k+1},\theta_k)}_{\text{gradient heterogeneity}} \\
&+ \underbrace{h^i_k(\omega^{i*}_{k+1},\theta_k) - h^i_k(\omega^{i*}_k,\theta_k)}_{\text{smoothness}} + \underbrace{h^i_k(\omega^{i*}_k,\theta_k) - \nabla J_i(\theta_k)}_{\text{approximation error}}\Bigg\|_2^2
\end{aligned} \tag{12}$$

Error of lower level. Next, we present key steps and highlight the technical differences of Proposition 1, which provides bounds on the critic error. Similar to the convergence analysis of federated temporal difference learning (Khodadadian et al. 2022; Wang et al. 2023; Zhang et al. 2024), the contraction property of the Bellman equation is utilized to produce a descent direction for the critic error. The informal decomposition can be expressed as:

$\mathbb{E}\left\|\omega_{t+1}-\omega^*\right\| \leq$ recursion + descent direction + client drift + gradient variance + gradient norm.

Note that we consider policy evaluation for policy π_{θ_k} and ω^* is the optimal value model for the mixture environment. In order to analyze $\mathbb{E}\left\|\omega_{k,t+1}-\omega^*\right\|$, we first bound an inner product term and it can be decomposed into three terms as shown in (13). As the objective of FedC is to minimize the

average MSPBE, term B can be *cancelled* (after the double summation before the inner product) due to the definiiton of average MSPBE. In contrast, in (Wang et al. 2023; Zhang et al. 2024), term B cannot be cancelled and becomes a *non-vanishing bias* in the convergence error. Then, (13) contributes to the descent direction term and client drift term of the above decomposition. Furthermore, from Lemma 1, the descent direction provides the essential negative term so that the contraction property can be guaranteed.

$$\begin{aligned}
&\frac{1}{N}\sum_i \frac{1}{\upsilon_i}\sum_{\upsilon=0}^{\upsilon_i-1}\mathbb{E}\left\langle g_i(\omega^i_{k,t,\upsilon}), \omega_{k,t}-\omega^*\right\rangle \\
&= \frac{1}{N}\sum_i \frac{1}{\upsilon_i}\sum_{k=0}^{\upsilon_i-1}\mathbb{E}\left\langle \omega_{k,t}-\omega^*\right., \\
&\left.\underbrace{g_i(\omega^i_{k,t,\upsilon})-g_i(\omega_{k,t})}_{\text{client drift}}+\underbrace{g_i(\omega_{k,t})-g(\omega_{k,t})}_{B}+\underbrace{g(\omega_{k,t})}_{\text{decent direction}}\right\rangle
\end{aligned} \tag{13}$$

The gradient variance term and gradient norm term are commonly appear in FRL analyses. Specifically, the gradient variance term is controlled by the mixing property of the Markov chains (Levin and Peres 2017); the gradient norm term can be decomposed by the client drift and gradient variance terms, presenting *linear speedup* effect as shown in Proposition 1.

Smoothness. Due to the Lipschitz continuity of $\omega^{i*}(\theta)$ as established in Lemma 5, the difference in the expected gradients at ω^{i*}_{k+1} and ω^{i*}_k can be related to the difference between θ_{k+1} and θ_k. Since the two successive policies can be controlled by α_k, the fourth term can also be controlled by α_k. This is how the single-loop architecture works where the biased term in the critics can be eliminated in the actors.

Comparisons with double-loop AC. The proof addresses the policy drift issue in the critic update by considering the "lower-level error" in (12). When the policy changes, the critic's parameters are initialized from the parameters of the previous policy. However, the double-loop variant leverages this additional property to eliminate the term, enabling a decoupled analysis.

Experiments

We test the SFAC algorithm in the Lunar Lander environment provided by OpenAI Gym and the codes were running using T4 Tensor Core GPUs. We evaluate SFAC against `A3C` (Shen et al. 2023).

We use multilayer perceptrons for the critic model and the actor model. Specifically, the critic model uses a linear combination of the basis functions, composed of Relu functions of the weighted states, and the actor model uses a three-layer neural networks with Relu as the activation functions in the hidden layer and the softmax function in the output layer. The stepsizes are set to $\alpha_k = 10^{-4}\times 0.99^{-k}$ and $\beta_k = 10^{-4}\times 0.99^{-k}$. The discount factor is 0.99. To measure the average performance, we collect the return for each round and use the average of the latest 10 rounds for all agents as the average return during training. In each inner round, the number of local updating steps is set to $T = 10$. The sample size of actors is set to 20. Note that the term 'rounds' on the x-axis represents the number of outer rounds.

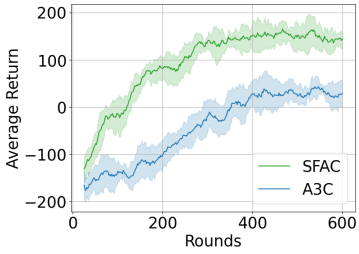
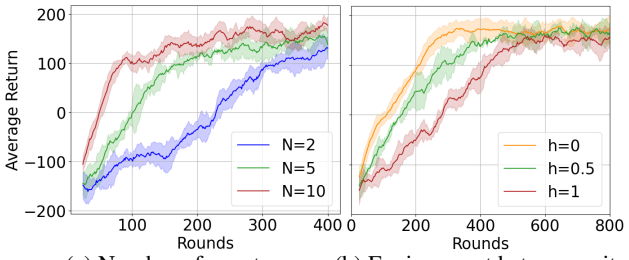


Figure 1: SFAC Performance in Comparison to A3C

Comparison with baseline. We illustrate the results of the comparison between SFAC and A3C as shown in Figure 1. In terms of average return, SFAC manages to obtain higher objective values. Moreover, two-level federation accelerates the training process which matches our theoretical results.



(a) Number of agents (b) Environment heterogeneity

Figure 2: Performance of SFAC

Linear speedup. To verify the advantages due to the federation, we conduct the experiments on the impact of the number of agents of SFAC as Figure 2(a) shows. With a certain level of environment heterogeneity, increasing the number of participated agents accelerate the training. This corroborates our theoretical insights and verifies the practical performance benefit offered by the participation of more agents.

Environment heterogeneity. To check the impact of environment heterogeneity, we construct tasks of SFAC with various h, which controls how different the state transitions are. Figure. 2(b) shows when we increases h, the performance will decreases, which validates the theoretical results.

Conclusion

In this paper, we have studied SFAC with heterogeneous environments and developed a two-level collaboration algorithm where the critics perform federated TD learning for policy evaluation, while the actors perform federated policy update for policy improvement, aiming to achieve the optimal global policy across all environments. Furthermore, we have analyzed that SFAC can asymptotically produce a near stationary point with a linear speedup, which is the first result in existing works on FRL with actor-critic algorithms considering heterogeneous environments.

Acknowledgments

This work was supported by U.S. NSF grants CNS-2145031 and CNS-2206977.

References

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

Chen, T.; Sun, Y.; and Yin, W. 2021. Closing the gap: Tighter analysis of alternating stochastic gradient methods for bilevel problems. *Advances in Neural Information Processing Systems (NeurIPS)*, 34: 25294–25307.

Chen, X.; and Zhao, L. 2023. Finite-time analysis of single-timescale actor-critic. *Advances in Neural Information Processing Systems (NeurIPS)*, 36: 7017–7049.

Doan, T.; Maguluri, S.; and Romberg, J. 2019. Finite-time analysis of distributed TD (0) with linear function approximation on multi-agent reinforcement learning. In *International Conference on Machine Learning (ICML)*, 1626–1635.

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

Fan, X.; Ma, Y.; Dai, Z.; Jing, W.; Tan, C.; and Low, B. K. H. 2021. Fault-tolerant federated reinforcement learning with theoretical guarantee. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1007–1021.

Ghadimi, S.; and Wang, M. 2018. Approximation methods for bilevel programming. *arXiv preprint arXiv:1802.02246*.

Hong, M.; Wai, H.-T.; Wang, Z.; and Yang, Z. 2023. A two-timescale stochastic algorithm framework for bilevel optimization: Complexity analysis and application to actor-critic. *SIAM Journal on Optimization*, 33(1): 147–180.

Huang, M.; Zhang, D.; and Ji, K. 2023. Achieving linear speedup in non-iid federated bilevel learning. In *International Conference on Machine Learning (ICML)*, 14039–14059.

Jin, H.; Peng, Y.; Yang, W.; Wang, S.; and Zhang, Z. 2022. Federated reinforcement learning with environment heterogeneity. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 18–37.

Khodadadian, S.; Sharma, P.; Joshi, G.; and Maguluri, S. T. 2022. Federated reinforcement learning: Linear speedup under markovian sampling. In *International Conference on Machine Learning (ICML)*, 10997–11057.

Kiran, B. R.; Sobh, I.; Talpaert, V.; Mannion, P.; Al Sallab, A. A.; Yogamani, S.; and Pérez, P. 2021. Deep reinforcement learning for autonomous driving: A survey. *IEEE Transactions on Intelligent Transportation Systems*, 23(6): 4909–4926.

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

Kumar, H.; Koppel, A.; and Ribeiro, A. 2023. On the sample complexity of actor-critic method for reinforcement learning with function approximation. *Machine Learning*, 112(7): 2433–2467.

Levin, D. A.; and Peres, Y. 2017. *Markov chains and mixing times*, volume 107. American Mathematical Soc.

Li, D.; Zhu, Y.; Gong, X.; Mao, S.; and Zhou, Y. 2024. Anarchic Federated Bilevel Optimization. In *2024 22nd International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt)*, 353–360.

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

Shalev-Shwartz, S.; Shammah, S.; and Shashua, A. 2016. Safe, multi-agent, reinforcement learning for autonomous driving. *arXiv preprint arXiv:1610.03295*.

Shen, H.; Zhang, K.; Hong, M.; and Chen, T. 2023. Towards understanding asynchronous advantage actor-critic: Convergence and linear speedup. *IEEE Transactions on Signal Processing*, 71: 2579–2594.

Srikant, R.; and Ying, L. 2019. Finite-time error bounds for linear stochastic approximation and td learning. In *Conference on Learning Theory (COLT)*, 2803–2830.

Sutton, R. S.; McAllester, D. A.; Singh, S. P.; and Mansour, Y. 1999. Policy gradient methods for reinforcement learning with function approximation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 1057–1063.

Tarzanagh, D. A.; Li, M.; Thrampoulidis, C.; and Oymak, S. 2022. Fednest: Federated bilevel, minimax, and compositional optimization. In *International Conference on Machine Learning (ICML)*, 21146–21179.

Wang, H.; Mitra, A.; Hassani, H.; Pappas, G. J.; and Anderson, J. 2023. Federated temporal difference learning with linear function approximation under environmental heterogeneity. *arXiv preprint arXiv:2302.02212*.

Wang, J.; Liu, Q.; Liang, H.; Joshi, G.; and Poor, H. V. 2020. Tackling the objective inconsistency problem in heterogeneous federated optimization. *Advances in neural information processing systems (NeurIPS)*, 33: 7611–7623.

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*.

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

Xie, Z.; and Song, S. 2023. FedKL: Tackling data heterogeneity in federated reinforcement learning by penalizing KL divergence. *IEEE Journal on Selected Areas in Communications*, 41(4): 1227–1242.

Xu, T.; Wang, Z.; and Liang, Y. 2020. Improving sample complexity bounds for (natural) actor-critic algorithms. *Advances in Neural Information Processing Systems (NeurIPS)*, 33: 4358–4369.

Yang, Y.; Xiao, P.; and Ji, K. 2023. Simfbo: Towards simple, flexible and communication-efficient federated bilevel learning. *Advances in Neural Information Processing Systems (NeurIPS)*, 36.

Ye, H.; Li, G. Y.; and Juang, B.-H. F. 2019. Deep reinforcement learning based resource allocation for V2V communications. *IEEE Transactions on Vehicular Technology*, 68(4): 3163–3173.

Yu, S.; Chen, X.; Zhou, Z.; Gong, X.; and Wu, D. 2020. When deep reinforcement learning meets federated learning: Intelligent multitimescale resource management for multiaccess edge computing in 5G ultradense network. *IEEE Internet of Things Journal*, 8(4): 2238–2251.

Zeng, S.; Anwar, M. A.; Doan, T. T.; Raychowdhury, A.; and Romberg, J. 2021. A decentralized policy gradient approach to multi-task reinforcement learning. In *Uncertainty in Artificial Intelligence (UAI)*, 1002–1012.

Zhang, C.; Wang, H.; Mitra, A.; and Anderson, J. 2024. Federated temporal difference learning with linear function approximation under environmental heterogeneity. In *International Conference on Learning Representations (ICLR)*.

Zhang, K.; Yang, Z.; and Başar, T. 2021. Decentralized multi-agent reinforcement learning with networked agents: Recent advances. *Frontiers of Information Technology & Electronic Engineering*, 22(6): 802–814.

Zhang, K.; Yang, Z.; Liu, H.; Zhang, T.; and Basar, T. 2018. Fully decentralized multi-agent reinforcement learning with networked agents. In *International Conference on Machine Learning (ICML)*, 5872–5881.

Zhu, Y.; and Gong, X. 2023. Distributed policy gradient with heterogeneous computations for federated reinforcement learning. In *2023 57th Annual Conference on Information Sciences and Systems (CISS)*, 1–6.

Zhu, Y.; and Gong, X. 2024. Sinlge-Loop Federated Actor-Critic across Heterogeneous Environments. *arXiv preprint arXiv:2412.14555*.