Decentralized Single-Timescale Actor Critic on Zero-Sum Two-Player Stochastic Games

Hongyi Guo [1] **Zuyue Fu** [1] **Zhuoran Yang** [2] **Zhaoran Wang** [1]

Abstract

We study the global convergence and global optimality of the actor-critic algorithm applied for the zero-sum two-player stochastic games in a decentralized manner. We focus on the single-timescale setting where the critic is updated by applying the Bellman operator only once and the actor is updated by policy gradient with the information from the critic. Our algorithm is in a decentralized manner, as we assume that each player has no access to the actions of the other one, which, in a way, protects the privacy of both players. Moreover, we consider linear function approximations for both actor and critic, and we prove that the sequence of joint policy generated by our decentralized linear algorithm converges to the minimax equilibrium at a sublinear rate $\mathcal{O}(\sqrt{K})$, where K is the number of iterations. To the best of our knowledge, we establish the global optimality and convergence of our decentralized actor-critic algorithm on zero-sum two-player stochastic games with linear function approximations for the first time.

1. Introduction

Reinforcement Learning (RL) (Sutton & Barto, 2018) has promoted the research of various areas, and has achieved phenomenal empirical successes with the power of deep neural networks (LeCun et al., 2015; Goodfellow et al., 2016). Examples include video games (Mnih et al., 2015; Vinyals et al., 2019), autonomous driving (Bojarski et al., 2016; Codevilla et al., 2018), robotics (Levine & Abbeel, 2014; Akkaya et al., 2019), and artistic creation (Jaques et al., 2016; Huang et al., 2019). RL is typically modeled as Markov decision process (Puterman, 2014, MDP), where an agent aims to learn an optimal policy via interaction with the environment. However, a wild range of real-world complex problems has a multi-agent nature, where more than one decision maker has to interact with each other, which motivates the research on multi-agent RL (MARL) (Busoniu et al., 2008). A wealth of works have been published on MARL, including both theoretical analysis (Wai et al., 2018; Zhang et al., 2018) and empirical training frameworks (Lowe et al., 2017; Rashid et al., 2018), in domains such as team-battle video games (Peng et al., 2017), autonomous driving (Zhou et al., 2020), and trading strategy analysis (Bao & Liu, 2019). See Zhang et al. (2019) for a detailed survey.

MARL is typically modeled as stochastic games (SGs) (Shapley, 1953), which generalizes the MDP framework. In stochastic games, agents (or players) may have different pay-offs (or rewards) that they aim to maximize. We focus on a specific multi-agent setting named zero-sum two-player stochastic games, where two players try to maximize opposite rewards. MARL plays a critical role in solving zero-sum two-player stochastic games, and a wealth of MARL algorithms and theoretical analysis are proposed in this area, e.g. (Pérolat et al., 2015; 2017; 2018; Xie et al., 2020; Bai & Jin, 2020; Wei et al., 2017; Sidford et al., 2020). In contrast to most existing works that assume players can observe each other's action and estimate a global value function, we consider a fully-decentralized setting, where each player has no access to the actions made by the other, even from previous timesteps, and each player maintains its own value function estimator, which protects the privacy of both players.

In our paper, we propose a decentralized single-timescale actor critic algorithm on zero-sum two-player stochastic games. In each iteration, each player learns the best response towards the other player's policy, so that their policies approach the minimax equilibrium at the end of training. We adopt the canonical actor critic (Konda & Tsitsiklis, 2000) scheme, where we let each player maintain its own critic function with the belief that the other player adopts the best response policy. Such decentralization enables our algorithm to handle zero-sum n-player ($n > 2$) stochastic games as well. In addition, we adopt linear function approximations in our algorithm for generalization, so that our

[1]Northwestern University [2]Princeton University. Correspondence to: Hongyi Guo <hongyiguo2025@u.northwestern.edu>, Zuyue Fu <zuyuefu2022@u.northwestern.edu>, Zhuoran Yang <zryang1993@gmail.com>, Zhaoran Wang <zhaoranwang@gmail.com>.

Proceedings of the 38^{th} International Conference on Machine Learning, PMLR 139, 2021.

algorithm is applicable to high-dimensional settings. We also establish a theoretical analysis of our algorithm and prove that the sequence of policies generated by our algorithm converges to the minimax equilibrium at a sublinear rate $\mathcal{O}(\sqrt{K})$, where K is the number of iterations.

Contribution. To summarize, the contributions in this paper are mainly three-folds: (1) We apply actor critic to zero-sum two-player stochastic games in a decentralized manner, which fully preserves the privacy of both players; (2) With the help of linear function approximation, we develop a generalizable decentralized linear actor critic algorithm, and give the closed forms of actor and critic update; (3) Our decentralized linear actor critic is theoretically guaranteed to generate a sequence of policies that converge to the minimax equilibrium at a sublinear rate of $\sqrt{K}$, where K is the number of iterations.

Related work. There is a large body of literature on applying multi-agent reinforcement learning methods to zero-sum two-player stochastic games. In particular, under the tabular settings, Littman & Szepesvári (1996); Littman (2001); Grau-Moya et al. (2018) extend the value iteration and Q-learning algorithms (Watkins & Dayan, 1992) to zero-sum stochastic games, and Pérolat et al. (2015; 2018); Srinivasan et al. (2018) extend actor-critic algorithms (Konda & Tsitsiklis, 2000). Among this line of works, our paper is most related to Pérolat et al. (2015), which proposes a bi-level actor critic algorithm that maintains a global critic function for both players and solves the exact minimax equilibrium with respect to the critic function in each iteration via linear programming. The algorithm they propose in Pérolat et al. (2015) is $\frac{2\gamma\epsilon}{(1-\gamma)^2}$-optimal, with ϵ being the error for the critic estimation, while our decentralized actor critic with linear function approximations achieves a sublinear converging rate $\mathcal{O}(\sqrt{K})$ to the minimax equilibria of two-player zero-sum stochastic games. The works of Lagoudakis & Parr (2012); Pérolat et al. (2016a;b); Yang & Wang (2019) also adopt function approximation techniques and establish finite-time convergence to the minimax equilibria. Different from our work, they consider variants of value-iteration methods, and their results are based on the framework of fitted value-iteration (Munos & Szepesvári, 2008). Motivated by the linear MDP model studied in Yang & Wang (2019), the works of Zanette & Brunskill (2019); Jin et al. (2020); Cai et al. (2020) impose the linear structure assumption on the reward function and the transition kernel of the underlying stochastic games, which is also imposed in our paper. Recently, Xie et al. (2020) studies zero-sum two-player simultaneous-move stochastic games, where they control a single player playing against an arbitrary opponent and aim to minimize the regret. The optimistic least-squares minimax value iteration they propose achieves an $\widetilde{\mathcal{O}}(\sqrt{K})$ upper bound on the duality gap and regret.

Notations. Let $[n] = \{1, \ldots, n\}$ for $n \in \mathbb{N}_+$. Also, we denote by $\mathcal{U}(a, b)$ the uniform distribution with boundaries a and b ($a < b$). For any measure μ, function $f : \mathcal{X} \to \mathbb{R}$, and $1 \le p \le +\infty$, we write $\|f\|_{\mu,p} = (\int_{\mathcal{X}} |f(x)|^p \mathrm{d}\mu)^{1/p}$.

2. Background

In this section, we introduce the background of zero-sum two-player stochastic games, Bellman operators and actor critic methods. A zero-sum two-player stochastic game is a generalization of an MDP to a 2-player setting, and can be modeled as a tuple $(\mathcal{S}, \mathcal{A}^1, \mathcal{A}^2, \mathcal{P}, \zeta, r, \gamma)$, where $\mathcal{S}$ is the state space, $\mathcal{A}^1$ and $\mathcal{A}^2$ are discrete action spaces for players 1 and 2, respectively, $\mathcal{P}\colon \mathcal{S} \times \mathcal{A}^1 \times \mathcal{A}^2 \to [0, 1]$ is the Markov transition kernel, $\zeta\colon \mathcal{S} \to [0, 1]$ is the initial state distribution, $r\colon \mathcal{S} \times \mathcal{A}^1 \times \mathcal{A}^2 \to [-r_{\max}, r_{\max}]$ is the deterministic reward function for both players, and $\gamma \in [0, 1)$ is the discount factor. A policy $\pi^p(a^p \,|\, s)$ measures the probability that player p takes action a^p at state s for $p \in \{1, 2\}$.

At the t-th step of the game, players 1 and 2 take actions $a_t^1 \sim \pi^1(\cdot \,|\, s_t)$ and $a_t^2 \sim \pi^2(\cdot \,|\, s_t)$ given the current state s_t, and receive deterministic rewards $r(s_t, a_t^1, a_t^2)$ and $-r(s_t, a_t^1, a_t^2)$, respectively, so that the sum of their rewards is always zero. The policies π^1 and π^2 together induce a stationary state distribution $\nu_{\pi^1,\pi^2}(s)$ and a stationary state-action distribution $\rho_{\pi^1,\pi^2}(s, a^1, a^2) = \nu_{\pi^1,\pi^2}(s) \cdot \pi^1(a^1 \,|\, s) \cdot \pi^2(a^2 \,|\, s)$. For any state-action pair $(s, a^1, a^2) \in \mathcal{S} \times \mathcal{A}^1 \times \mathcal{A}^2$, we define the action-value function Q^{π^1,π^2} as follows,

$$Q^{\pi^1,\pi^2}(s, a^1, a^2) = (1-\gamma) \cdot \mathbb{E}_{\pi^1,\pi^2}\Big[\sum_{t=0}^{\infty} \gamma^t \cdot r(s_t, a_t^1, a_t^2) \,\Big|\, s_0 = s, a_0^1 = a^1, a_0^2 = a^2\Big], \tag{2.1}$$

and the total expected reward $J(\pi^1, \pi^2)$ as follows,

$$J(\pi^1, \pi^2) = \mathbb{E}_{s\sim\zeta,\pi^1,\pi^2}\big[Q^{\pi^1,\pi^2}(s, a^1, a^2)\big].$$

Here, the expectation is taken with respect to $s \sim \zeta(\cdot)$, $a^1 \sim \pi^1(\cdot \,|\, s)$, and $a^2 \sim \pi^2(\cdot \,|\, s)$. For zero-sum two-player stochastic games, player 1 aims to maximize $J(\pi^1, \pi^2)$, while player 2 aims to minimize $J(\pi^1, \pi^2)$. In other words, we aim to solve the following optimization problem,

$$\max_{\pi^1} \min_{\pi^2} J(\pi^1, \pi^2). \tag{2.2}$$

2.1. Bellman Operators

In this section, we introduce some Bellman operators, which simplifies our notations in the sequel. Let $v : \mathcal{S} \to \mathbb{R}$, $q^1 : (\mathcal{S} \times \mathcal{A}^1) \to \mathbb{R}$, $q^2 : (\mathcal{S} \times \mathcal{A}^2) \to \mathbb{R}$, and $q : (\mathcal{S} \times$

$\mathcal{A}^1 \times \mathcal{A}^2) \to \mathbb{R}$ be any functions. We define the following transition operators,

$$
\begin{aligned}
&[\mathbb{P}v](s,a^1,a^2)\\
&\quad = \mathbb{E}\big[v(s_1) \,\big|\, s_0 = s, a_0 = a^1, a_2 = a^2\big],\\
&[\mathbb{P}_1^{\pi^1} q^1](s,a^1,a^2)\\
&\quad = \mathbb{E}_{\pi^1}\big[q^1(s_1,a_1^1) \,\big|\, s_0 = s, a_0 = a^1, a_2 = a^2\big],\\
&[\mathbb{P}_2^{\pi^2} a^2](s,a^1,a^2)\\
&\quad = \mathbb{E}_{\pi^2}\big[q^2(s_1,a_1^2) \,\big|\, s_0 = s, a_0 = a^1, a_2 = a^2\big],\\
&[\mathbb{P}^{\pi^1,\pi^2} q](s,a^1,a^2)\\
&\quad = \mathbb{E}_{\pi^1,\pi^2}\big[q(s_1,a_1^1,a_1^2) \,\big|\, s_0 = s, a_0 = a^1, a_2 = a^2\big].
\end{aligned}
$$

We also define the following Bellman operators,

$$
\begin{aligned}
&[\mathbb{T}_1^{\pi^1} q^1](s,a^1,a^2)\\
&\quad = (1-\gamma)\cdot r(s,a^1,a^2) + \gamma\cdot[\mathbb{P}_1^{\pi^1} q^1](s,a^1,a^2),
\end{aligned}
\tag{2.3}
$$

$$
\begin{aligned}
&[\mathbb{T}_2^{\pi^2} q^2](s,a^1,a^2)\\
&\quad = (1-\gamma)\cdot r(s,a^1,a^2) + \gamma\cdot[\mathbb{P}_2^{\pi^2} q^2](s,a^1,a^2),
\end{aligned}
\tag{2.4}
$$

$$
\begin{aligned}
&[\mathbb{T}^{\pi^1,\pi^2} q](s,a^1,a^2)\\
&\quad = (1-\gamma)\cdot r(s,a^1,a^2) + \gamma\cdot[\mathbb{P}^{\pi^1,\pi^2} q](s,a^1,a^2),
\end{aligned}
\tag{2.5}
$$

$$
\begin{aligned}
&[\widehat{\mathbb{T}}^{\pi^1,\pi^2} q^1](s,a^1)\\
&\quad = \mathbb{E}_{a^2\sim\pi^2(\cdot\,|\,s)}\big[(1-\gamma)\cdot r(s,a^1,a^2)\\
&\qquad + \gamma\cdot\mathbb{P}^{\pi^1,\pi^2} q^1(s,a^1,a^2)\big],
\end{aligned}
\tag{2.6}
$$

$$
\begin{aligned}
&[\widetilde{\mathbb{T}}^{\pi^1,\pi^2} q^2](s,a^2)\\
&\quad = \mathbb{E}_{a^1\sim\pi^1(\cdot\,|\,s)}\big[(1-\gamma)\cdot r(s,a^1,a^2)\\
&\qquad + \gamma\cdot\mathbb{P}^{\pi^1,\pi^2} q^2(s,a^1,a^2)\big].
\end{aligned}
\tag{2.7}
$$

By the definition of $\mathbb{T}^{\pi^1,\pi^2}$ in (2.5), Q^{π^1,π^2} is the unique fixed point of $\mathbb{T}^{\pi^1,\pi^2}$. To simplify the notation, we define

$$
\mathbb{P}^\ell = \underbrace{\mathbb{P}\mathbb{P}\ldots\mathbb{P}}_{\ell}.
$$

Such a notation is also adopted for other operators such as $\mathbb{P}^{\pi^1,\pi^2}$ and $\mathbb{T}^{\pi^1,\pi^2}$.

2.2. Minimax Equilibrium

We define the optimal value of the zero-sum two-player stochastic games as

$$
Q^{\pi_*^1,\pi_*^2} = \max_{\pi^1}\min_{\pi^2} Q^{\pi^1,\pi^2} = \min_{\pi^2}\max_{\pi^1} Q^{\pi^1,\pi^2}. \tag{2.8}
$$

Intuitively, stochastic games can be cast as a matrix game in terms of the visitation measure (Altman, 1999), which allows us to combine with Von Neumann's minimax theorem (Von Neumann & Morgenstern, 1947; Patek, 1997) to get (2.8). To simplify the notation, we denote by $Q^* = Q^{\pi_*^1,\pi_*^2}$. Note that (2.8) also defines optimal policies π_*^1 and π_*^2 at the minimax equilibrium. In the setting of zero-sum two-player stochastic games, the notion of minimax equilibrium is equivalent to that of Nash equilibrium (Conitzer, 2016), so that no player gets higher expected rewards by deviating from the policies π_*^1 and π_*^2 at the equilibrium.

2.3. Actor Critic

One way of solving the objective in (2.2) is via directly applying the actor critic method (Konda & Tsitsiklis, 2000) in a centralized way. The actor critic method is composed of iterations of actor update and critic update. In critic update, a policy evaluation algorithm such as temporal-difference (Tesauro, 1995) is adopted to estimate the action-value function, while in actor update, a policy improvement algorithm such as Schulman et al. (2015; 2017) is invoked to refine the policy with the information provided by the critic. To adopt actor critic in the setting of zero-sum two-player stochastic games, a natural way is to maintain an estimator of Q^{π^1,π^2} defined in (2.1), and improve the policy of each player with the information provided by that estimator, while treating the other player's policy as a fixed component of the underlying environment. This technique requires access to both players' actions when maintaining the global critic (the estimator of Q^{π^1,π^2}). Such centralized global critic leaks the decision-making strategy of both players. Also, algorithms with centralized global critic are hard to be extended to zero-sum n-player stochastic games, since the magnitude of the domain space of $Q^{\pi^1,\pi^2,\ldots,\pi^n}$ grows exponentially with n growing linearly. Those defects of centralized actor critic motivate us to consider applying actor critic in a fully decentralized manner, as the algorithm presented in the following section.

3. Algorithm

In this section, we introduce the details of our decentralized actor critic algorithm on zero-sum two-player stochastic games. The key of developing our decentralized algorithm is to maintain two separated estimators of the state-action value functions, one for each player. We denote by $\widehat{Q}^{\pi^1}(s,a^1)$ the estimator of $\min_{\pi^2}\mathbb{E}_{a^2\sim\pi^2(\cdot\,|\,s)}[Q^{\pi^1,\pi^2}(s,a^1,a^2)]$ for player 1, and by $\widehat{Q}^{\pi^2}(s,a^2)$ the estimator of $\max_{\pi^1}\mathbb{E}_{a^1\sim\pi^1(\cdot\,|\,s)}[Q^{\pi^1,\pi^2}(s,a^1,a^2)]$ for player 2. To simplify the notation, for the policies π_k^1 and π_k^2 in the k-th iteration, we denote the stationary state distribution $\nu_{\pi_k^1,\pi_k^2}$ and state-action distribution $\rho_{\pi_k^1,\pi_k^2}$ by ν_k and ρ_k, respectively. We present our decentralized actor critic algorithm for zero-sum two-player stochastic games in Algorithm 1

Algorithm 1 Decentralized Actor Critic on Zero-Sum Two-Player Stochastic Games

Require: Initial policies π_0^1, π_0^2, initial estimators $\widehat{Q}^{\pi_0^1}, \widehat{Q}^{\pi_0^2}$, the number of iterations K, the number of iterations T for the subroutine, regularization parameter β, and TD parameter m.

1: **for** $k \leftarrow 0, 1, 2, \ldots, K-1$ **do**

2: Actor update of player 1: $\pi_{k+1}^1 \leftarrow \operatorname{argmax}_{\pi^1} \mathbb{E}_{\nu_k}[\langle \widehat{Q}^{\pi_k^1}(s,\cdot), \pi^1(\cdot \mid s)\rangle - \beta \cdot \mathrm{KL}(\pi^1(\cdot \mid s) \,\|\, \pi_k^1(\cdot \mid s))]$.

3: Find the best responding policy π_{k+1}^2 via $(\pi_{k+1}^2, \widehat{Q}^{\pi_{k+1}^2}) \leftarrow$ *BestResponse*$(\pi_{k+1}^1, \pi_k^2, \widehat{Q}^{\pi_k^2}, T, \beta, m)$.

4: Critic update of player 1: $\widehat{Q}^{\pi_{k+1}^1} \leftarrow (\widehat{\mathbb{T}}^{\pi_{k+1}^1, \pi_{k+1}^2})^m \widehat{Q}^{\pi_k^1}$.

5: **end for**

Ensure: Policy sequences $\{\pi_k^1\}_{k\in[K]}$ and $\{\pi_k^2\}_{k\in[K]}$.

Algorithm 2 The Subroutine $BestResponse$

Require: Current policy π^1 of player 1, initial policy π_0^2, initial estimator $\widehat{Q}^{\pi_0^2}$, the number of iterations T, regularization parameter β, and TD parameter m.

1: **for** $t \leftarrow 0, 1, 2, \ldots, T-1$ **do**

2: Actor update of player 2: $\pi_{t+1}^2 \leftarrow \operatorname{argmax}_{\pi^2} \mathbb{E}_{\nu_k}[\langle \widehat{Q}^{\pi_t^2}(s,\cdot), \pi^2(\cdot \mid s)\rangle - \beta \cdot \mathrm{KL}(\pi^2(\cdot \mid s) \,\|\, \pi_t^2(\cdot \mid s))]$.

3: Critic update of player 2: $\widehat{Q}^{\pi_{t+1}^2} \leftarrow (\widetilde{\mathbb{T}}^{\pi_{k+1}^1, \pi_{t+1}^2})^m \widehat{Q}^{\pi_t^2}$.

4: **end for**

Ensure: π_T^2 and $\widehat{Q}^{\pi_T^2}$.

and its subroutine in Algorithm 2.

Our algorithm follows the idea of treating the other player as part of the environment when performing actor updates and critic updates on each player. In every iteration, we first perform actor update on player 1. See line 2 of Algorithm 1, where $\mathrm{KL}(\cdot \,\|\, \cdot)$ computes the Kullback–Leibler divergence. We adopt PPO algorithm (Schulman et al., 2017) to calculate the optimal policy with respect to $\widehat{Q}^{\pi_k^1}$ and meanwhile constrain the KL divergence between π_k^1 and π_{k+1}^1. The KL constraint controls the learning step and prevents the algorithm converging too fast to a suboptimal solution. In line 3 of Algorithm 1, a subroutine $BestResponse$ given in Algorithm 2 is performed to compute the best response of player 2 towards π_{k+1}^1. Specifically, in subroutine $BestResponse$, we run the actor and critic update of player 2 for T iterations. The actor update in line 2 of Algorithm 2 adopts the same PPO algorithm as in line 2 of Algorithm 1. In the critic update in line 3 of Algorithm 2, we apply the operator $\widetilde{\mathbb{T}}^{\pi_{k+1}^1, \pi_{t+1}^2}$ m times on $\widehat{Q}^{\pi_t^2}$, where $\widetilde{\mathbb{T}}^{\pi_{k+1}^1, \pi_{t+1}^2}$ is defined in (2.7). After T iterations, the policy π_T^2 obtained well approximates the best response of player 2 towards π_{k+1}^1, i.e., π_T^2 well approximates $\operatorname{argmin}_{\pi^2} Q^{\pi_{k+1}^1, \pi^2}$. Then, we perform critic update for player 1 in line 4 of Algorithm 1, where we apply the operator $\widehat{\mathbb{T}}^{\pi_{k+1}^1, \pi_{k+1}^2}$ m times on $\widehat{Q}^{\pi_k^1}$, with π_{k+1}^2 being the policy generated by the subroutine and $\widehat{\mathbb{T}}^{\pi_{k+1}^1, \pi_{k+1}^2}$ defined in (2.6). Thus, the estimator $\widehat{Q}^{\pi_{k+1}^1}$ approximates $\mathbb{E}_{\pi_{k+1}^1, \pi_{k+1}^2}[Q^{\pi_{k+1}^1, \pi_{k+1}^2}]$, which is approximately $\min_{\pi^2} \mathbb{E}_{\pi_{k+1}^1, \pi^2}[Q^{\pi_{k+1}^1, \pi^2}]$ by the construction of π_{k+1}^2.

Our algorithm is single-timescale and decentralized. Each time the actor of player 1 is updated, the actor and critic of player 2 are updated T times to well-approximate the best response. In other words, player 1 and player 2 are on different levels. However, following from the discussion in §2.2, we get the same minimax equilibrium by switching the roles of player 1 and player 2. Also, from Algorithm 1 and 2, the number of actor updates are consistent with that of critic updates for both players, which indicates that our algorithm is single-timescale. Finally, in our algorithm, each player maintains its own critic, and when updating the critic with the Bellman operator defined in (2.6) and (2.7), the action of the other player is actually marginalized out. Thus, the whole algorithm words in a decentralized manner.

3.1. Linear Approximation

In this section, we consider linear approximation for both policy and action-value function of each player. Here and in the sequel, we use $p \in \{1, 2\}$ to identify each player. We focus on the family of energy-based policies as follows,

$$\pi_{\theta_k^p}^p(a^p \mid s) = \frac{\exp\big(\tau_k^{-1} \cdot \phi^p(s, a^p)^\top \theta_k^p\big)}{\sum_{a' \in \mathcal{A}^p} \exp\big(\tau_k^{-1} \cdot \phi^p(s, a')^\top \theta_k^p\big)}. \quad (3.1)$$

Here, τ_k is the temperature parameter, $\theta_k^p \in \mathbb{R}^d$ is the parameter of π^p in the k-th iteration, and $\phi^p \in \mathbb{R}^d$ is the feature vector for player p. The parameters θ^p and τ are initialized as follows,

$$\theta_0^p \sim \mathcal{U}(-1, 1), \quad \tau_0 \leftarrow \infty. \quad (3.2)$$

Here, $\tau_0 \leftarrow \infty$ indicates that the initial policy is uniform, which is commonly adopted in the literature. We assume

the feature vector ϕ^p in (3.1) is always available for player p, and although the action spaces $\mathcal{A}^1$ and $\mathcal{A}^2$ may differ, we assume the feature vectors for both players are of the same dimension. To simplify the notation, we denote $\pi^p_{\theta^p_k}$ by π^p_k, for $p \in \{1, 2\}$. The linear approximation of the action-value function for player p is given by

$$\widehat{Q}^{\pi^p_k}_{\omega^p_k}(s, a^p) = \phi^p(s, a^p)^\top \omega^p_k. \tag{3.3}$$

Here, $\omega^p_k \in \mathbb{R}^d$ is the parameter for the estimator $\widehat{Q}^{\pi^p_k}$, and ϕ^p is the same feature vector in (3.1). To simplify the notation, we denote $\widehat{Q}^{\pi^p_k}_{\omega^p_k}$ by $\widehat{Q}^{\pi^p_k}$. The parameter ω^p is initialized as follows,

$$\omega^p_0 \sim \mathcal{U}(-R/\sqrt{d}, R/\sqrt{d}). \tag{3.4}$$

In (3.4), we let the parameter ω^p lie in a centered ball in $\mathbb{R}^d$ with radius R, where d is the dimension of the feature vector ϕ^p. With the approximated action-value function, we develop the closed forms of actor and critic update for both players as follows.

Actor update for player p. The following lemma gives the closed forms of the updates in line 2 of Algorithm 1 and line 2 of Algorithm 2.

Lemma 3.1. With the energy-based policy π^p_k given in (3.1) and the linear action-value function $\widehat{Q}^{\pi^p_k}$ given in (3.3), we let $\theta^p_{k+1} = \operatorname{argmax}_\theta \mathbb{E}_{\nu_k}[\langle \widehat{Q}^{\pi^p_k}(s, \cdot), \pi^p(\cdot \,|\, s)\rangle - \beta \cdot \mathrm{KL}(\pi^p(\cdot \,|\, s) \,\|\, \pi^p_k(\cdot \,|\, s))]$. Then, θ^p_{k+1} has the following closed form

$$\theta^p_{k+1} = \tau_{k+1} \cdot (\beta^{-1}\omega^p_k + \tau_k^{-1}\theta^p_k). \tag{3.5}$$

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

By Lemma 3.1 and setting $\tau_{k+1} = (\tau_k^{-1} + \eta)^{-1}$ for constant $\eta > 0$, we obtain an exact greedy policy with respect to the approximated state-action value function $\widehat{Q}^{\pi_{k+1}}$ under the KL constraint. In what follows, we introduce the closed form of critic update for both players under our linear approximation.

Critic update for player 1. To simplify our analysis, we assume $m = 1$ in Algorithm 1 and 2, hereafter. Thus, the critic update for player 1 in line 4 of Algorithm 1 corresponds to the following formula

$$\widetilde{\omega}^1_{k+1} = \operatorname*{argmin}_{\omega^1} \mathbb{E}_{\rho_{k+1}}\Big[\big(\widehat{Q}^{\pi^1_{k+1}}(s, a^1) - \mathbb{T}_1^{\pi^1_{k+1}} \widehat{Q}^{\pi^1_k}(s, a^1)\big)^2\Big]. \tag{3.6}$$

The solution to the minimization problem in (3.6) is the minimum mean square error (MMSE) estimator of $\mathbb{T}^{\pi^1_{k+1}} \widehat{Q}^{\pi^1_k}$, which has the following closed form,

$$\widetilde{\omega}^1_{k+1} = \Big(\mathbb{E}_{\rho_{k+1}}\big[\phi^1(s, a^1)\phi^1(s, a^1)^\top\big]\Big)^{-1} \cdot \mathbb{E}_{\rho_{k+1}}\big[\mathbb{T}^{\pi^1_{k+1}} \widehat{Q}^{\pi^1_k}(s, a^1) \cdot \phi^1(s, a^1)\big]. \tag{3.7}$$

Then, we use data to approximate the expectation over the stationary state-action distribution ρ_{k+1}. Specifically, we sample $\{(s_{i,0}, a^1_{i,0}, a^2_{i,0})\}_{i\in[N]}$ and $\{(s_{i,1}, a^1_{i,1}, a^2_{i,1}, r_{i,1}, s_{i,2}, a^1_{i,2}, a^2_{i,2})\}_{i\in[N]}$ where $(s_{i,0}, a^1_{i,0}, a^2_{i,0}) \overset{\text{i.i.d}}{\sim} \rho_{k+1}$, $(s_{i,1}, a^1_{i,1}, a^2_{i,1}) \overset{\text{i.i.d}}{\sim} \rho_{k+1}$, $r_{i,1} = r(s_{i,1}, a^1_{i,1}, a^2_{i,1})$, $s_{i,2} \sim \mathcal{P}(\cdot \,|\, s_{i,1}, a^1_{i,1}, a^2_{i,1})$, $a^1_{i,2} \sim \pi^1_{k+1}(\cdot \,|\, s_{i,2})$, $a^2_{i,2} \sim \pi^2_{k+1}(\cdot \,|\, s_{i,2})$, and N is the sample size. Then, the parameter ω^1_{k+1} for $\widehat{Q}^{\pi^1_{k+1}}$ that we obtain is given by

$$\omega^1_{k+1} = \Pi_R\Big\{\big[\sum_{i=1}^N \phi^1(s_{i,0}, a^1_{i,0})\phi^1(s_{i,0}, a^1_{i,0})^\top\big]^{-1} \cdot \sum_{i=1}^N \big((1-\gamma)\cdot r_{i,1} + \gamma \cdot \widehat{Q}^{\pi^1_k}(s_{i,2}, a^1_{i,2})\big)\phi^1(s_{i,1}, a^1_{i,1})\Big\}. \tag{3.8}$$

Here, Π_R is the projection operator which projects the parameter into the centered ball in $\mathbb{R}^d$ with radius R.

Critic update for player 2. We omit the derivation of the closed form of critic update for player 2, since it follows the same idea as that for player 1. In the end, the critic update for player 2 in line 3 of Algorithm 2 takes the following form,

$$\omega^2_{t+1} = \Pi_R\Big\{\big[\sum_{j=1}^N \phi^2(s_{j,0}, a^2_{j,0})\phi^2(s_{j,0}, a^2_{j,0})^\top\big]^{-1} \cdot \sum_{j=1}^N \big((1-\gamma)\cdot r_{j,1} + \gamma \cdot \widehat{Q}^{\pi^2_t}(s_{j,2}, a^2_{j,2})\big)\phi^2(s_{j,1}, a^2_{j,1})\Big\}. \tag{3.9}$$

Here, ω^2_{t+1} is the parameter for the estimator $\widehat{Q}^{\pi^2_{t+1}}$, Π_R is the same projection with that in (3.8), N is the sample size, and we sample $\{(s_{j,0}, a^1_{j,0}, a^2_{j,0})\}_{j\in[N]}$ and $\{(s_{j,1}, a^1_{j,1}, a^2_{j,1}, r_{j,1}, s_{j,2}, a^1_{j,2}, a^2_{j,2})\}_{j\in[N]}$ where $(s_{j,0}, a^1_{j,0}, a^2_{j,0}) \overset{\text{i.i.d}}{\sim} \rho_{\pi^1, \pi^2_{t+1}}$, $(s_{j,1}, a^1_{j,1}, a^2_{j,1}) \overset{\text{i.i.d}}{\sim} \rho_{\pi^1, \pi^2_{t+1}}$, $r_{j,1} = r(s_{j,1}, a^1_{j,1}, a^2_{j,1})$, $s_{j,2} \sim \mathcal{P}(\cdot \,|\, s_{j,1}, a^1_{j,1}, a^2_{j,1})$, $a^1_{j,2} \sim \pi^1(\cdot \,|\, s_{j,2})$, $a^2_{j,2} \sim \pi^2_{t+1}(\cdot \,|\, s_{j,2})$.

We refer our linear approximation of Algorithm 1 as decentralized linear actor critic algorithm and conclude it in Algorithm 3. In what follows, we give our theoretical analysis of our decentralized linear actor critic algorithm.

Algorithm 3 Decentralized Linear Actor Critic on Zero-Sum Two-Player Stochastic Games

Require: The number of iterations K, the number of iterations T for the subroutine, regularization parameter β, learning parameter η, and TD parameter m.

Initialize the parameters of π_0^1 and π_0^2 by $\theta_0^1 \sim \mathcal{U}(-1,1)$ and $\theta_0^2 \sim \mathcal{U}(-1,1)$, respectively.
Initialize the parameters of $\widehat{Q}^{\pi_0^1}$ and $\widehat{Q}^{\pi_0^2}$ by $\omega_0^1 \sim \mathcal{U}(-R/\sqrt{d}, R/\sqrt{d})$ and $\omega_0^2 \sim \mathcal{U}(-R/\sqrt{d}, R/\sqrt{d})$, respectively.
Set $\tau_0 \leftarrow \infty$.
for $k \leftarrow 0, 1, 2, \ldots, K-1$ **do**
  Update θ_k^1 to get θ_{k+1}^1 following (3.5) with p substituted by 1.
  Set $\theta_{k,0}^2 \leftarrow \theta_k^2$ and $\omega_{k,0}^2 \leftarrow \omega_k^2$.
  for $t = 0, 1, 2, \ldots, T-1$ **do**
    Update $\theta_{k,t}^2$ to get $\theta_{k,t+1}^2$ following (3.5) with p substituted by 2.
    Update $\omega_{k,t}^2$ to get $\omega_{k,t+1}^2$ following (3.9) with π_{t+1}^2 substituted by $\pi_{k,t+1}^2$.
  end for
  $\pi_{k+1}^2 \leftarrow \pi_{k,T}^2$.
  Update ω_k^1 to get ω_{k+1}^1 following (3.8).
  $\tau_{k+1} \leftarrow (\tau_k^{-1} + \eta)^{-1}$.
end for

Ensure: Policy sequences $\{\pi_k^1\}_{k=0}^K$ and $\{\pi_k^2\}_{k=0}^K$.

4. Theoretical Results

In this section, we introduce our theoretical results for our decentralized linear actor critic algorithm presented in Algorithm 3. We make the following assumptions to help us establish the theoretical results.

Assumption 4.1 (Concentration Coefficient)**.** We assume there exists a state-action distribution ρ such that for an arbitrary sequence of policies $\{\pi_\ell^1, \pi_\ell^2\}_{\ell\in[k]}$, the k-step future state-action distribution $\rho\mathbb{P}^{\pi_1^1,\pi_1^2}\cdots\mathbb{P}^{\pi_k^1,\pi_k^2}$ is absolutely continuous with respect to ρ. Also, it holds for such ρ that

$$C_\rho = (1-\gamma)^2 \sum_{k=1}^{\infty} k\gamma^k \cdot c(k) < \infty,$$
$$\text{where} \quad c(k) = \sup_{\{\pi_\ell\}_{\ell\in[k]}} \left\| \frac{\mathrm{d}(\rho\mathbb{P}^{\pi_1^1,\pi_1^2}\cdots\mathbb{P}^{\pi_\ell^1,\pi_\ell^2})}{\mathrm{d}\rho} \right\|_{\rho,\infty}.$$

In Assumption 4.1, C_ρ is known as the discounted-average concentrability coefficient of the future-state-action distributions, which measures the stochastic stability properties of the stochastic game. Such assumption is commonly imposed in the literature (Munos, 2005; Munos & Szepesvári, 2008; Scherrer, 2013; Scherrer et al., 2015; Farahmand et al., 2016).

Assumption 4.2 (Zero Approximation Error)**.** It holds for any energy-based policies π^1 and π^2 taking the form of (3.1) and any linear estimator $\widehat{Q}^{\pi^p}$ taking the form of (3.3) that

$$\inf_{\omega\in\mathcal{B}(0,R)} \mathbb{E}_{\rho_{\pi^1,\pi^2}}\Big[\big(\mathbb{T}_p^{\pi^p}\widehat{Q}^{\pi^p}(s,a^1,a^2) - \phi^p(s,a^p)^\top\omega\big)^2\Big] = 0.$$

Assumption 4.2 imposes a linear structure on the underlying MDP, namely linear MDP (Yang & Wang, 2019). Specifically speaking, it assumes that the Bellman operator of any joint policy maps a linear value function to a linear function. Such an assumption is commonly adopted in the theoretical analysis of RL (Lagoudakis & Parr, 2012; Pérolat et al., 2016a;b; Yang & Wang, 2019). When Assumption 4.2 does not hold, we just need to add an extra estimation error term in our results.

Assumption 4.3 (Well-conditioned Feature)**.** We assume $\|\phi^p(s,a^p)\|_2 \le 1$ for any $s \in \mathcal{S}, a^p \in \mathcal{A}^p, p \in \{1,2\}$ and that the minimum singular value of the matrix $\mathbb{E}_{\rho_k}[\phi^p(s,a^p)\phi^p(s,a^p)^\top]$ is uniformly lower bounded by $\lambda_* > 0$ for any $k \in [K]$.

Assumption 4.3 ensures that the minimization problem in (3.6) has a unique minimizer, and that $\|\mathbb{E}_{\rho_k}[\phi^p(s,a^p)\phi^p(s,a^p)^\top]^{-1}\|_2 \le 1/\lambda_*$. Similar assumptions are commonly imposed in the literature (Bhandari et al., 2018; Zou et al., 2019; Wu et al., 2020). In tabular settings, such an assumption is just stating that all state-action pairs can be reached.

We emphasize that even with those assumptions, the theoretical analysis is still nontrivial due to the decentralization, the nonconvexity for each agent, and their interaction in a minimax manner.

4.1. Main Theorem

In this section, we establish the upper bound of the total optimality gap of our decentralized linear actor critic algorithm. Specifically speaking, we are interested in upper bounding $\mathcal{L}(K) = \mathbb{E}_\rho[\sum_{k=0}^K l_k]$, where ρ is a state-action distribu-

tion satisfying Assumption 4.1 and $l_k = Q^* - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}$. Here, recall that Q^* is the optimal value function defined in (2.8), and we denote by $\widetilde{\pi}^2_k$ the best response of player 2 towards π^1_k, i.e., $Q^{\pi^1_k,\widetilde{\pi}^2_k} = \min_{\pi^2} Q^{\pi^1_k,\pi^2}$. We denote by ϵ_k the following critic update error for $k \in \{0,1,\ldots,K-1\}$,

$$\epsilon_k(s,a^1,a^2) = [\mathbb{T}_1^{\pi^1_{k+1}}\widehat{Q}^{\pi^1_k} - \widehat{Q}^{\pi^1_{k+1}}](s,a^1,a^2), \quad (4.1)$$

where $\widetilde{\pi}^2_{k+1}$ is the best response towards π^1_{k+1}. We now give the following proposition which upper bounds ϵ_k.

Proposition 4.4. Let $T = m = 1$ and suppose Assumption 4.2 and 4.3 hold. Then, it holds for any given $k \in \{0,1,\ldots,K-1\}$ with probability at least $1-2\delta$ that

$$\mathbb{E}_\rho\big[|\epsilon_k|\big] \le \frac{16(r_{\max}+R)}{\sqrt{N}\lambda_*}\cdot\log\left(\frac{2d}{\delta}\right),$$

where the expectation is taken with respect to $(s,a^1,a^2)\sim\rho$, the uncertainty comes from ω_k, $r_{\max}$ is the maximum reward magnitude, R is the domain radius of all $\{\omega_k\}_{k\in[K]}$, N is the sample size, λ_* is given in Assumption 4.3, and d is the feature dimension.

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

Proposition 4.4 establishes the upper bound of ϵ_k in one iteration. We denote by ϵ_Q the following union bound,

$$\epsilon_Q = \max_{k\in\{0,1,\ldots,K-1\}} \mathbb{E}_\rho[|\epsilon_k|], \quad (4.2)$$

and use the following corollary of Proposition 4.4 to upper bound ϵ_Q.

Corollary 4.5. Under the same conditions of Proposition 4.4, it holds with probability at least $1-\delta$ that

$$\epsilon_Q \le \frac{16(r_{\max}+R)}{\sqrt{N}\lambda_*}\cdot\log\left(\frac{4Kd}{\delta}\right),$$

where the uncertainty comes from $\{\omega_k\}_{k\in[K]}$.

Finally, we give the following theorem which establishes the optimality gap of our Algorithm 1.

Theorem 4.6. Let K be a sufficiently large number, ρ be a state-action distribution satisfying Assumption 4.1, $\beta=\sqrt{K}$, $N=\Omega(KC_\rho^2\log^2(Kd)/\lambda_*^2)$, $\delta\in(0,1)$, $T=1$, $m=1$, $\eta=1$, and the sequence of policy parameters $\{\theta_k\}_{k=0}^K$ be generated by running Algorithm 3 for K iterations. Under Assumptions 4.1,4.2 and 4.3, it holds with probability at least $1-\delta$ that

$$\mathcal{L}(K) = \mathbb{E}_\rho[\sum_{k=0}^K Q^* - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}] \le \mathcal{O}\left(\frac{\sqrt{K}\log(|\mathcal{A}^1|/\delta)}{(1-\gamma)^2}\right),$$

where the expectation is taken with respect to $(s,a^1,a^2)\sim\rho$ and the uncertainty comes from $\{\omega_k\}_{k\in[K]}$.

We sketch the proof in §5. Theorem 4.6 establishes an $\mathcal{O}(\sqrt{K})$ suboptimality of our decentralized linear actor critic algorithm, where K is the total number of iterations. To better understand the theorem, note that in single-agent setting and with access to the true action-value function Q^π, the nature policy gradient method achieves the same $\mathcal{O}(\sqrt{K})$ regret (Liu et al., 2019; Agarwal et al., 2019; Cai et al., 2019). We use $2KN$ samples during the first k iterations, as we need N data points for Player 1 and $TN=N$ data points for Player 2. The sample complexity becomes $\widetilde{\mathcal{O}}(K^2)$ by our choice of parameters where $\widetilde{\mathcal{O}}$ hides constants and logarithms. Moreover, by the $1/\sqrt{K}$ rate of convergence, the sample complexity to attain a Nash equilibrium is $\widetilde{\mathcal{O}}(1/\epsilon^4)$. Note that we employ an on-policy algorithm that uses a fresh batch of data points at each iteration. If we employ an off-policy algorithm instead that reuses the same batch, the sample complexity becomes $\widetilde{\mathcal{O}}(K)$ by our choice of parameters, which translates to $\widetilde{\mathcal{O}}(1/\epsilon^2)$. Such a sample complexity matches that of Pérolat et al. (2015); Bai & Jin (2020); Xie et al. (2020); Bai et al. (2020) in the tabular case. However, our setting allows for function approximation and decentralized execution, which are common in practice.

5. Proof Sketch

In this section, we sketch the proof of Theorem 4.6. For notational convenience, we let $\widehat{Q}^{\pi^1}(s,a^1,a^2) = \widehat{Q}^{\pi^1}(s,a^1)$, which leads to $\mathbb{T}^{\pi^1,\pi^2}\widehat{Q}^{\pi^1} = \mathbb{T}_1^{\pi^1}\widehat{Q}^{\pi^1}$ for any π^2, where $\mathbb{T}^{\pi^1,\pi^2}$ is defined in (2.5) and $\mathbb{T}_1^{\pi^1}$ is defined in (2.3). Since the value Q^* defined in (2.8) is the invariant point of $\mathbb{T}^{\pi^1_*,\pi^2_*}$, it holds that $Q^* = \mathbb{T}^{\pi^1_*,\pi^2_*}Q^*$. Thus, we can write

$$\begin{aligned}
l_k &= Q^* - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}} \\
&= \mathbb{T}^{\pi^1_*,\pi^2_*}Q^* - \mathbb{T}^{\pi^1_*,\pi^2_*}\widehat{Q}^{\pi^1_k} + \mathbb{T}^{\pi^1_*,\pi^2_*}\widehat{Q}^{\pi^1_k} - \mathbb{T}_1^{\pi^1_{k+1}}\widehat{Q}^{\pi^1_k} \\
&\quad + \mathbb{T}_1^{\pi^1_{k+1}}\widehat{Q}^{\pi^1_k} - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}} \\
&= \mathbb{T}^{\pi^1_*,\pi^2_*}Q^* - \mathbb{T}^{\pi^1_*,\pi^2_*}\widehat{Q}^{\pi^1_k} + \mathbb{T}_1^{\pi^1_*}\widehat{Q}^{\pi^1_k} - \mathbb{T}_1^{\pi^1_{k+1}}\widehat{Q}^{\pi^1_k} \\
&\quad + \mathbb{T}_1^{\pi^1_{k+1}}\widehat{Q}^{\pi^1_k} - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}} \\
&= \underbrace{\gamma\mathbb{P}^{\pi^1_*,\pi^2_*}(Q^* - \widehat{Q}^{\pi^1_k})}_{\triangleq g_k} + \underbrace{\gamma(\mathbb{P}_1^{\pi^1_*} - \mathbb{P}_1^{\pi^1_{k+1}})\widehat{Q}^{\pi^1_k}}_{\triangleq f_k} \\
&\quad + \underbrace{\mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} - Q^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}}_{\triangleq h_k}. \quad (5.1)
\end{aligned}$$

Since the value Q^* defined in (2.8) is the fixed point of

$\mathbb{T}^{\pi^1_*,\pi^2_*}$, it holds that $Q^* = \mathbb{T}^{\pi^1_*,\pi^2_*}Q^*$. Thus, we can write

$$\begin{aligned}
g_{k+1} &= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}(Q^* - \widehat{Q}^{\pi^1_{k+1}}) \\
&= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}\big[\mathbb{T}^{\pi^1_*,\pi^2_*}Q^* - \mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} \\
&\quad + \mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} - \widehat{Q}^{\pi^1_{k+1}}\big] \\
&= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}\big[\mathbb{T}^{\pi^1_*,\pi^2_*}Q^* - \mathbb{T}^{\pi^1_*,\pi^2_*}\widehat{Q}^{\pi^1_k} + \mathbb{T}^{\pi^1_*,\pi^2_*}\widehat{Q}^{\pi^1_k} \\
&\quad - \mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} + \mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} - \widehat{Q}^{\pi^1_{k+1}}\big] \\
&= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}\big[\gamma\mathbb{P}^{\pi^1_*,\pi^2_*}(Q^* - \widehat{Q}^{\pi^1_k}) + \gamma(\mathbb{P}_1^{\pi^1_*} - \mathbb{P}_1^{\pi^1_{k+1}})\widehat{Q}^{\pi^1_k} \\
&\quad + \mathbb{T}^{\pi^1_{k+1},\widetilde{\pi}^2_{k+1}}\widehat{Q}^{\pi^1_k} - \widehat{Q}^{\pi^1_{k+1}}\big] \\
&= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}(g_k + f_k + \epsilon_k), \qquad (5.2)
\end{aligned}$$

where f_k is defined in (5.1) and ϵ_k is defined in (4.1). By applying (5.2) k times, we obtain that

$$\begin{aligned}
g_k &= \gamma\mathbb{P}^{\pi^1_*,\pi^2_*}(Q^* - \widehat{Q}^{\pi^1_k}) \\
&= (\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^k g_0 + \sum_{t=1}^{k}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^t(f_{k-t} + \epsilon_{k-t}),
\end{aligned} \qquad (5.3)$$

By the triangle inequality, it holds that

$$|g_0| = |Q^* - \widehat{Q}^{\pi^1_0}| \le r_{\max} + \|\phi^1\|_2 \cdot \|\omega^1_0\|_2 \le r_{\max} + R, \qquad (5.4)$$

where $r_{\max}$ is the maximum magnitude of the rewards, R is the radius given in (3.4), and the last inequality follows from $\|\omega^1_0\|_2 \le R$ and $\|\phi^1\|_2 \le 1$ which comes from Assumption 4.3. The following lemma establishes the upper bound for $\sum_{k=1}^{K} f_k$.

Lemma 5.1. Under Assumption 4.3, it holds for any $(s, a^1, a^2) \in \mathcal{S} \times \mathcal{A} \times \mathcal{A}$ that

$$\sum_{k=0}^{K} f_k(s, a^1, a^2) \le \beta \log(|\mathcal{A}^1|), \qquad (5.5)$$

where β is the regularization parameter in line 2 of Algorithm 1, and $\mathcal{A}^1$ is the action space of player 1.

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

Recall that ρ is a given state-action distribution that satisfies Assumption 4.1. Upon telescoping with respect to k and taking expectation with respect to $(s, a^1, a^2) \sim \rho$, we obtain that

$$\begin{aligned}
&\mathbb{E}_\rho\big[\sum_{k=0}^{K}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^k g_0\big] \\
&\quad \le \mathbb{E}_\rho\big[\sum_{k=0}^{K}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^k |g_0|\big] \le \frac{r_{\max} + R}{1-\gamma}, \qquad (5.6)
\end{aligned}$$

$$\begin{aligned}
&\mathbb{E}_\rho\big[\sum_{k=0}^{K}\sum_{t=1}^{k}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^t f_{k-t}\big] \\
&\quad = \mathbb{E}_\rho\big[\sum_{t=1}^{K}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^t \sum_{k=0}^{K-t} f_k\big] \le \frac{\beta\log(|\mathcal{A}^1|)}{1-\gamma}, \qquad (5.7)
\end{aligned}$$

and

$$\begin{aligned}
&\mathbb{E}_\rho\big[\sum_{k=0}^{K}\sum_{t=1}^{k}(\gamma\mathbb{P}^{\pi^1_*,\pi^2_*})^t \epsilon_{k-t}\big] \\
&\quad \le \sum_{k=1}^{K}\sum_{t=1}^{k}\gamma^t c(t)\mathbb{E}_\rho\big[\epsilon_{k-t}\big] \le \frac{KC_\rho\epsilon_Q}{(1-\gamma)^2}. \qquad (5.8)
\end{aligned}$$

Here, the second inequality in (5.6) follows from (5.4), the inequality in (5.7) follows from (5.5), the first inequality in (5.8) follows from Assumption 4.1, and the second inequality in (5.8) follows from Corollary 4.5. Combining (5.3), (5.6), (5.7), and (5.8) together, we obtain that

$$\mathbb{E}_\rho[\sum_{k=0}^{K}\sum_{t=1}^{k} g_k] \le \frac{r_{\max} + R + \beta\log(|\mathcal{A}^1|) + KC_\rho\epsilon_Q}{(1-\gamma)^2}. \qquad (5.9)$$

Then, the following lemma upper bounds $\mathbb{E}_\rho[\sum_{k=0}^{K} h_k]$ for which the proof is deferred to the appendix.

Lemma 5.2. Under Assumption 4.1 and 4.3, it holds that

$$\mathbb{E}_\rho[\sum_{k=0}^{K} h_k(s, a^1, a^2)] \le \frac{r_{\max} + R + KC_\rho\epsilon_Q}{(1-\gamma)^2}, \qquad (5.10)$$

where h_k is defined in (5.1).

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

Then, combining (5.1), (5.5), (5.9), and (5.10) together, we obtain that

$$\begin{aligned}
\mathbb{E}_\rho[\sum_{k=0}^{K} l_k] &= \mathbb{E}_\rho[\sum_{k=0}^{K} f_k + g_k + h_k] \\
&\le \frac{2\big(KC_\rho\epsilon_Q + \beta\log(|\mathcal{A}^1|) + r_{\max} + R\big)}{(1-\gamma)^2}.
\end{aligned}$$

Thus, following from Corollary 4.5, by choosing $\beta = \sqrt{K}$ and $N = \Omega(KC_\rho^2\log^2(Kd)/\lambda_*^2)$, it holds with probability at least $1-\delta$ that

$$\begin{aligned}
\mathbb{E}_\rho[\sum_{k=0}^{K} l_k] &\le \frac{32KC_\rho(r_{\max} + R)}{(1-\gamma)^2 \cdot \sqrt{N}\lambda_*} \cdot \log\left(\frac{4Kd}{\delta}\right) \\
&\quad + \frac{2\beta}{(1-\gamma)^2} \cdot \log(|\mathcal{A}^1|) + \frac{2(r_{\max} + R)}{(1-\gamma)^2} \\
&\le \mathcal{O}\left(\frac{\sqrt{K}\log(|\mathcal{A}^1|/\delta)}{(1-\gamma)^2}\right),
\end{aligned}$$

which concludes the proof of Theorem 4.6.

Acknowledgement

Zhaoran Wang acknowledges National Science Foundation (Awards 2048075, 2008827, 2015568, 1934931), Simons Institute (Theory of Reinforcement Learning), Amazon, J.P. Morgan, and Two Sigma for their supports. Zhuoran Yang acknowledges Simons Institute (Theory of Reinforcement Learning).

References

Agarwal, A., Kakade, S. M., Lee, J. D., and Mahajan, G. Optimality and approximation with policy gradient methods in Markov decision processes. *arXiv preprint arXiv:1908.00261*, 2019.

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

Altman, E. *Constrained Markov decision processes*, volume 7. CRC Press, 1999.

Bai, Y. and Jin, C. Provable self-play algorithms for competitive reinforcement learning. In *International Conference on Machine Learning*, pp. 551–560. PMLR, 2020.

Bai, Y., Jin, C., and Yu, T. Near-optimal reinforcement learning with self-play. *arXiv preprint arXiv:2006.12007*, 2020.

Bao, W. and Liu, X.-y. Multi-agent deep reinforcement learning for liquidation strategy analysis. *arXiv preprint arXiv:1906.11046*, 2019.

Bhandari, J., Russo, D., and Singal, R. A finite time analysis of temporal difference learning with linear function approximation. *arXiv preprint arXiv:1806.02450*, 2018.

Bojarski, M., Del Testa, D., Dworakowski, D., Firner, B., Flepp, B., Goyal, P., Jackel, L. D., Monfort, M., Muller, U., Zhang, J., et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Busoniu, L., Babuska, R., and De Schutter, B. A comprehensive survey of multi-agent reinforcement learning. *IEEE Transactions on Systems, Man, And Cybernetics-Part C: Applications and Reviews, 38 (2), 2008*, (2):156–172, 2008.

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

Cai, Q., Yang, Z., Jin, C., and Wang, Z. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pp. 1283–1294. PMLR, 2020.

Codevilla, F., Miiller, M., López, A., Koltun, V., and Dosovitskiy, A. End-to-end driving via conditional behavior cloning. In *2018 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 1–9. IEEE, 2018.

Conitzer, V. On stackelberg mixed strategies. *Synthese*, 193 (3):689–703, 2016.

Farahmand, A.-m., Ghavamzadeh, M., Szepesvári, C., and Mannor, S. Regularized policy iteration with nonparametric function spaces. *Journal of Machine Learning Research*, 17(1):4809–4874, 2016.

Goodfellow, I., Bengio, Y., and Courville, A. *Deep learning*. MIT press, 2016.

Grau-Moya, J., Leibfried, F., and Bou-Ammar, H. Balancing two-player stochastic games with soft q-learning. *arXiv preprint arXiv:1802.03216*, 2018.

Henderson, H. V. and Searle, S. R. On deriving the inverse of a sum of matrices. *Siam Review*, 23(1):53–60, 1981.

Huang, Z., Heng, W., and Zhou, S. Learning to paint with model-based deep reinforcement learning. In *International Conference on Computer Vision*, pp. 8709–8718, 2019.

Jaques, N., Gu, S., Turner, R. E., and Eck, D. Generating music by fine-tuning recurrent neural networks with reinforcement learning. In *Deep Reinforcement Learning Workshop, NIPS*, 2016.

Jin, C., Yang, Z., Wang, Z., and Jordan, M. I. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pp. 2137–2143. PMLR, 2020.

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

Lagoudakis, M. and Parr, R. Value function approximation in zero-sum markov games. *arXiv preprint arXiv:1301.0580*, 2012.

LeCun, Y., Bengio, Y., and Hinton, G. Deep learning. *Nature*, 521(7553):436–444, 2015.

Levine, S. and Abbeel, P. Learning neural network policies with guided policy search under unknown dynamics. In *NIPS*, volume 27, pp. 1071–1079. Citeseer, 2014.

Littman, M. L. Value-function reinforcement learning in Markov games. *Cognitive Systems Research*, 2(1):55–66, 2001.

Littman, M. L. and Szepesvári, C. A generalized reinforcement-learning model: Convergence and applications. In *icml*, volume 96, pp. 310–318, 1996.

Liu, B., Cai, Q., Yang, Z., and Wang, Z. Neural proximal/trust region policy optimization attains globally optimal policy. *arXiv preprint arXiv:1906.10306*, pp. 10564–10575, 2019.

Lowe, R., Wu, Y., Tamar, A., Harb, J., Abbeel, P., and Mordatch, I. Multi-agent actor-critic for mixed cooperative-competitive environments. *arXiv preprint arXiv:1706.02275*, 2017.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015.

Munos, R. Error bounds for approximate value iteration. In *Proceedings of the 20th national conference on Artificial intelligence-Volume 2*, pp. 1006–1011. AAAI Press, 2005.

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

Patek, S. D. *Stochastic and shortest path games: Theory and algorithms*. PhD thesis, Massachusetts Institute of Technology, 1997.

Peng, P., Wen, Y., Yang, Y., Yuan, Q., Tang, Z., Long, H., and Wang, J. Multiagent bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play starcraft combat games. *arXiv preprint arXiv:1703.10069*, 2017.

Pérolat, J., Scherrer, B., Piot, B., and Pietquin, O. Approximate dynamic programming for two-player zero-sum Markov games. In *International Conference on Machine Learning*, 2015.

Pérolat, J., Piot, B., Geist, M., Scherrer, B., and Pietquin, O. Softened approximate policy iteration for Markov games. In *International Conference on Machine Learning*, 2016a.

Pérolat, J., Piot, B., Scherrer, B., and Pietquin, O. On the use of non-stationary strategies for solving two-player zero-sum Markov games. In *aistats*, pp. 893–901, 2016b.

Pérolat, J., Strub, F., Piot, B., and Pietquin, O. Learning nash equilibrium for general-sum markov games from batch data. In *Artificial Intelligence and Statistics*, pp. 232–241. PMLR, 2017.

Pérolat, J., Piot, B., and Pietquin, O. Actor-critic fictitious play in simultaneous move multistage games. In *aistats*, 2018.

Puterman, M. L. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, 2014.

Rashid, T., Samvelyan, M., Schroeder, C., Farquhar, G., Foerster, J., and Whiteson, S. Qmix: Monotonic value function factorisation for deep multi-agent reinforcement learning. In *International Conference on Machine Learning*, pp. 4295–4304. PMLR, 2018.

Scherrer, B. On the performance bounds of some policy search dynamic programming algorithms. *arXiv preprint arXiv:1306.0539*, 2013.

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

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

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

Shapley, L. S. Stochastic games. *Proceedings of the national academy of sciences*, (10):1095–1100, 1953.

Sidford, A., Wang, M., Yang, L., and Ye, Y. Solving discounted stochastic two-player games with near-optimal time and sample complexity. In *International Conference on Artificial Intelligence and Statistics*, pp. 2992–3002. PMLR, 2020.

Srinivasan, S., Lanctot, M., Zambaldi, V., Pérolat, J., Tuyls, K., Munos, R., and Bowling, M. Actor-critic policy optimization in partially observable multiagent environments. In *Advances in Neural Information Processing Systems*, pp. 3426–3439, 2018.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT press, 2018.

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

Vinyals, O., Babuschkin, I., Chung, J., Mathieu, M., Jaderberg, M., Czarnecki, W., Dudzik, A., Huang, A., Georgiev, P., Powell, R., et al. Alphastar: Mastering the Real-Time Strategy Game StarCraft II, 2019.

Von Neumann, J. and Morgenstern, O. *Theory of Games and Economic Behavior*. Princeton University Press, 1947.

Wai, H.-T., Yang, Z., Wang, P. Z., and Hong, M. Multi-agent reinforcement learning via double averaging primal-dual optimization. In *Advances in Neural Information Processing Systems*, pp. 9649–9660, 2018.

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

Wei, C.-Y., Hong, Y.-T., and Lu, C.-J. Online reinforcement learning in stochastic games. In *Advances in Neural Information Processing Systems*, pp. 4987–4997, 2017.

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

Xie, Q., Chen, Y., Wang, Z., and Yang, Z. Learning zero-sum simultaneous-move markov games using function approximation and correlated equilibrium. In *Conference on Learning Theory*, pp. 3674–3682. PMLR, 2020.

Yang, L. F. and Wang, M. Sample-optimal parametric q-learning with linear transition models. *arXiv preprint arXiv:1902.04779*, 2019.

Zanette, A. and Brunskill, E. Tighter problem-dependent regret bounds in reinforcement learning without domain knowledge using value function bounds. In *International Conference on Machine Learning*, pp. 7304–7312. PMLR, 2019.

Zhang, K., Yang, Z., Liu, H., Zhang, T., and Başar, T. Fully decentralized multi-agent reinforcement learning with networked agents. *arXiv preprint arXiv:1802.08757*, 2018.

Zhang, K., Yang, Z., and Başar, T. Multi-agent reinforcement learning: A selective overview of theories and algorithms. *arXiv preprint arXiv:1911.10635*, 2019.

Zhou, M., Luo, J., Villela, J., Yang, Y., Rusu, D., Miao, J., Zhang, W., Alban, M., Fadakar, I., Chen, Z., et al. Smarts: Scalable multi-agent reinforcement learning training school for autonomous driving. *arXiv preprint arXiv:2010.09776*, 2020.

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