Convergence Rates for Localized Actor-Critic in Networked Markov Potential Games

Abstract

We introduce a class of networked Markov potential games where agents are associated with nodes in a network, each agent has its own local potential function, and the rewards of each agent depend only on the states and actions of agents within a κ-hop neighborhood. In this context, we propose a localized actor-critic policy. The policy is scalable since each agent uses only local information and does not need access to the global state. Further, the policy avoids the curse of dimensionality through the use of function approximation. Our main results provide finite-sample guarantees. We prove an $\mathcal{O}(1/M^{1/4})$ rate of convergence of the averaged Nash regret up to a critic error and a localization error, where M is the total iteration number. This is the first finite-sample bound for multi-agent competitive games that does not depend on the number of agents.

1 INTRODUCTION

Large-scale systems where strategic agents interact with each other have received significant attention recently, motivated by applications in power systems [Shi et al., 2022], EV charging [Lee et al., 2022], board games [Silver et al., 2017], and more. Controlling such systems can be challenging due to the scale of the system, uncertainty about the model, communication constraints, and strategic behavior of the agents. Inspired by the recent success of reinforcement learning (RL), there is an increasing interest in applying RL methods to environments with multi-agent strategic interactions. However, when multi-agent RL (MARL) considers agents with strategic behaviors, analysis of the system behavior becomes even more challenging due to the time-varying nature of the environment faced by each agent, which results from the strategic decisions of the other agents. As a result, the theoretical analysis of strategic MARL is still limited especially when it comes to large-scale systems.

Results on MARL in strategic settings to this point have tended to focus on games with a small number of players, e.g., 2-player zero-sum stochastic games [Littman, 1994], or games with special structure, e.g., Markov potential games (MPGs) [Fox et al., 2022a]. MPGs in particular provide a setting in which the challenges of large-scale systems can be studied and so have drawn attention. The intuition behind an MPG parallels that of classical (one-shot) potential games. Specifically, the existence of a potential function guarantees that agents can converge to a global equilibrium even when using greedy localized updates. MPGs have wide-ranging applications including variants of congestion games [Leonardos et al., 2022, Fox et al., 2022b], medium access control [Macua et al., 2018], and the stochastic lake game [Dechert and O'Donnell, 2006]. However, existing theoretical results for MPGs rely on the assumption that a centralized global state exists and can be observed by each individual agent. Such an assumption rules out applications in many large-scale systems including transportation networks [Zhang and Pavone, 2016] and social networks [Chakrabarti et al., 2008], where the global state space can be exponentially large in the number of agents and/or each agent can only observe its own local state.

A promising approach for the design of scalable, local MARL algorithms in strategic settings is to exploit the networked structure of practical applications to design algorithms with sample complexity that only depends on *local* properties of the network instead of *global* state. This approach has recently been successful in the case of cooperative MARL. For example, Qu et al. [2020] provides a scalable localized algorithm with a sample complexity that does not depend on the number of agents and Lin et al. [2021], Zhang et al. [2022c] use similar ideas. However, to this point, local algorithms that exploit network structure do not exist in the strategic MARL setting. Thus, we ask: *Can we design a scalable, local algorithm with finite-time bounds for networked MARL with strategic agents?*

Submitted to the 39th Conference on Uncertainty in Artificial Intelligence (UAI 2023). **To be used for reviewing only**.

1.1 CONTRIBUTIONS

We address the question above by introducing a class of networked Markov potential games (NMPGs) as the networked counterpart of classical MPGs. Importantly, NMPGs represent a broader class of games than MPGs, and draw focus to algorithm design that uses only local information.

We design a localized actor-critic framework that is a combination of independent policy gradient and localized TD(λ) learning with linear function approximation. Notably, our algorithm is *model-free*, uses only *local information*, and successfully incorporates *function approximation*. This avoids both the need for communication of the global state and the so-called "curse of dimensionality" in MARL.

Our main results provide a finite-sample bound for our proposed algorithm, which has an $\mathcal{O}(M^{-1/4})$ rate of convergence to the averaged Nash regret up to an estimation error of the critic and an approximation error of using local information. To our knowledge, we are the first to provide a localized algorithm in competitive MARL settings with provable performance guarantees that do not depend on the number of agents.

Our results are enabled by novel analysis of the critic in our localized actor-critic framework. In particular, we propose a localized cost evaluation problem, a new MARL setting to investigate the performance of a local algorithm under a fixed policy. As a critical part of the proof, we propose a novel concept called a "sub-chain" that connects local algorithms to their global counterparts, enabling performance bounds via bounds on the gap between the two.

1.2 RELATED WORK

Markov Potential Games. Our work adds to the literature on Markov potential games (MPG) in multi-agent reinforcement learning (MARL). Analytic results for non-cooperative MARL are challenging to obtain because agents learn in a non-stationary environment as other agents update their policies. As a result, analysis has focused on special cases like two-player Markov games [Littman, 1994] and MPGs [Fox et al., 2022a]. The case of MPGs has received considerable attention recently because the potential games are broadly applicable [Leonardos et al., 2022] and the existence of potential functions enables efficient convergence with provable guarantees [Zhang et al., 2022b, Ding et al., 2022, Fox et al., 2022a, Zhang et al., 2022a]. While these papers provide algorithms with provable convergence guarantees, they assume that all agents share a common global state and can observe the global state to decide local actions. An important open question is understanding how to learn in settings where global information is not available. Our work studies the MARL setting where each agent has its own local state and can only decide local actions based on local states.

MARL in Networked Systems. The Markov decision process (MDP) model we study is inspired by a series of works on Networked MARL [Qu et al., 2020, Lin et al., 2021, Zhang et al., 2022c], where RL agents are located on a network $\mathcal{G}$. In such models, the local state transition of an agent is affected by its own local state/action and its direct neighbors' local states. Networked MARL is applicable to a wide range of applications, including communication networks [Vogels et al., 2003], social networks [Chakrabarti et al., 2008], and traffic networks [Zhang and Pavone, 2016]. Compared with general MARL, the additional structure of Networked MARL provides a critical exponential decay property on the local Q functions, which enables the design of scalable decentralized actor-critic algorithms [Qu et al., 2020, Lin et al., 2021]. All prior works on Networked MARL study the case when agents cooperatively maximize the sum of all local rewards. In contrast, our work studies a non-cooperative MPG in which each agent has its own objective.

Finite-Time Analysis of TD Learning Variants. Temporal difference (TD) learning and its variants are widely used for RL policy evaluation, which plays a critical role in most policy gradient algorithms. The asymptotic analysis of TD learning dates back to Tsitsiklis and Van Roy [1996], Jaakkola et al. [1994], while finite-time convergence bounds have received attention in the last decade [Tsitsiklis and Van Roy, 1997]. In TD learning, function approximation is a useful technique to reduce the dimension of learning parameters at the cost of incurring an approximation error that depends on the function class. Recently, many breakthroughs are made on finite-time error bounds for TD learning with function approximation [Bhandari et al., 2018, Srikant and Ying, 2019]. Meanwhile, in multi-agent settings, localized TD learning is crucial for limiting communication and the need for global information [Lin et al., 2021]. Our work contributes to the literature analyzing TD learning by providing a novel finite-time error bound for localized TD learning with function approximation.

2 PROBLEM DESCRIPTION

We study MARL in the context of networked multi-agent Markov games. Specifically, we consider a setting with n agents that are associated with an undirected graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, \ldots, n\}$ is the set of nodes and $\mathcal{E} \subseteq \mathcal{N} \times \mathcal{N}$ is the set of edges. We denote by $\text{dist}(i, j)$ the graph distance between agents i and j.

The local state space and local action space of agent i are denoted by $\mathcal{S}_i$ and $\mathcal{A}_i$, respectively, which are both finite sets. The global state is denoted as $s = (s_1, \ldots, s_n) \in \mathcal{S} := \prod_{i=1}^n S_i$ and the global action is defined similarly. For any set $I \subseteq \mathcal{N}$, we use s_I to denote the states of the agents in I

and use $\mathcal{S}_I := \prod_{i\in I} \mathcal{S}_i$ to denote the state space of agents in I. We define a_I and $\mathcal{A}_I$ as the action and action space of the agents in I and $\mu \in \Delta(\mathcal{S})$ as the initial state distribution.

At time $t \geq 0$, given current state $s(t)$ and action $a(t)$, for each agent $i \in \mathcal{N}$, its successor state $s_i(t+1)$ is independently generated according to the following transition probability, which is only dependent on its neighbors' states and its own action: $\mathcal{P}(s(t+1) \mid s(t), a(t)) = \prod_{i=1}^{n} \mathcal{P}_i(s_i(t+1) \mid s_{N_i}(t), a_i(t))$. where $N_i = \{i\} \cup \{j \in \mathcal{N} \mid (i,j) \in \mathcal{E}\}$ denotes the neighborhood of i (including i). In addition, given an integer $\kappa \geq 0$, we use N_i^κ to denote the κ-hop neighborhood of i, i.e., $N_i^\kappa = \{i\} \cup \{j \in \mathcal{N} \mid \text{dist}(i,j) \leq \kappa\}$, and use $-N_i^\kappa = \mathcal{N}/N_i^\kappa$ to denote the set of agents that are not in N_i^κ. We use $U_i^\kappa = N_i^\kappa/\{i\}$ to denote the agents in the κ-hop neighborhood of i, excluding i itself.

Finally, each agent $i \in \mathcal{N}$ is associated with a deterministic reward function $r_i : \mathcal{S} \times \mathcal{A} \in [0,1]$. The interval $[0,1]$ is chosen without loss of generality over the set of bounded reward functions. In general, agent i's reward depends on the global state and the global action. Due to the network structure, we assume that there exists a non-negative integer κ_r such that the reward function of each agent depends only on the states and the actions of other agents within its κ_r-hop neighborhood, i.e., $r_i(s,a) = r_i(s_{N_i^{\kappa_r}}, a_{N_i^{\kappa_r}})$ for all i. Intuitively, we expect the dependence between two agents to be weaker as their graph distance grows.

Policy. Each agent $i \in \mathcal{N}$ is associated with a localized policy $\xi_i : \mathcal{S}_i \mapsto \Delta(\mathcal{A}_i)$. Given a subset $I \subseteq \mathcal{N}$, we define $\xi_I : \mathcal{S}_I \mapsto \Delta(\mathcal{A}_I)$ as the joint policy of agents in I. Note that $\xi_I(a_I \mid s_I) = \prod_{i\in I} \xi_i(a_i \mid s_i)$. We use Ξ_i to denote agent i's local policy space, and Ξ_I to denote the joint policy space of agents in I. When $I = \mathcal{N}$, we omit the subscript and just write ξ for $\xi_\mathcal{N}$ (and Ξ for $\Xi_\mathcal{N}$). Throughout, we also use $\xi = (\xi_1, \xi_2, \cdots, \xi_n)$ to represent ξ and its local policy components. We frequently work with softmax policies, which are defined as

$$\xi_i^{\theta_i}(a_i|s_i) = \frac{\exp(\theta_{i,s_i,a_i})}{\sum_{a_i'\in\mathcal{A}_i} \exp(\theta_{i,s_i,a_i'})}, \ \forall\, i, s_i, a_i, \quad (1)$$

where $\xi_i^{\theta_i}$ stands for agent i's local policy parametrized by the weight vector $\theta_i \in \mathbb{R}^{|\mathcal{S}_i||\mathcal{A}_i|}$. We denote $\theta = (\theta_1, \theta_2, \cdots, \theta_n)$ as the parameter of a global policy ξ^θ.

When the policy uses softmax parameterization, we may abuse the policy parameter θ to represent the policy ξ for simplicity. For example, we may write $J_i(\theta)$ for $J_i(\xi^\theta)$. The dimension of θ is $\sum_{i\in\mathcal{N}} |\mathcal{S}_i| \times |\mathcal{A}_i|$.

Value Function. Given a policy ξ, we define the state-action value function (aka. the Q-function) of agent i as

$$Q_i^\xi(s,a) = \mathbb{E}_\xi\left[\sum_{t=0}^{\infty} \gamma^t r_i(s_{N_i^{\kappa_r}}(t), a_{N_i^{\kappa_r}}(t)) \middle| s(0)=s, a(0)=a\right]$$

for all (s,a), where $\mathbb{E}_\xi[\,\cdot\,]$ is taken with respect to the randomness in the (stochastic) policy ξ and the transition dynamics. With Q_i^ξ defined above, the averaged Q-function and the value function of agent i are defined as

$$\overline{Q}_i^\xi(s,a_i) = \mathbb{E}_{a_{-i}\sim\xi_{-i}(\cdot|s_{-i})}[Q_i^\xi(s,a_i,a_{-i})], \ \forall\, (s,a_i),$$
$$V_i^\xi(s) = \mathbb{E}_{a_i\sim\xi_i(\cdot|s_i)}[\overline{Q}_i^\xi(s,a_i)], \ \forall\, s,$$

where we use s_{-i} (respectively, a_{-i}) to denote the joint state (respectively, action) of the agents in $\mathcal{N}/\{i\}$. Given a fixed initial state distribution μ, we define $J_i(\xi) = \mathbb{E}_{s\sim\mu}[V_i^\xi(s)]$. When using a softmax policy with parameter θ, we abuse the notation and denote $J_i(\theta)$ for $J_i(\xi^\theta)$.

Finally, we define the advantage function of agent i as $A_i^\xi(s,a) = Q_i^\xi(s,a) - V_i^\xi(s)$, and the averaged advantage function of agent i as $\overline{A}_i^\xi(s,a_i) = \overline{Q}_i^\xi(s,a_i) - V_i^\xi(s)$.

Discounted State Visitation Distribution. Given a policy ξ and an initial state $s(0)$, we define the *discounted state visitation distribution* as $d_{s(0)}^\xi(s) = (1-\gamma)\sum_{t=0}^{\infty} \gamma^t \Pr^\xi[s(t) = s \mid s(0)]$ for all $s \in \mathcal{S}$, where $\Pr^\xi[s(t) = s \mid s(0)]$ denotes the probability that $s(t) = s$ given that the initial state is $s(0)$ and the global policy is ξ. We use $d_{\mu'}^\xi(s) := \mathbb{E}_{s(0)\sim\mu'}[d_{s(0)}^\xi(s)]$ to represent the discounted state visitation distribution when the initial state distribution is μ'. In particular, when the initial state distribution is μ, we omit the subscript and write $d^\xi(s)$.

3 NETWORKED MARKOV POTENTIAL GAMES

Our focus is a class of networked multi-agent MDPs that we name Networked Markov Potential Games (NMPGs). This class generalizes the standard of Markov Potential Games.

Definition 3.1. *A networked multi-agent MDP is called a κ_G-NMPG if there exists a set of local potential functions $\{\Phi_i\}_{i\in\mathcal{N}}$, where $\Phi_i : \Xi \to \mathbb{R}$ for all $i \in \mathcal{N}$, such that the following equality holds for any $i \in \mathcal{N}$, $j \in N_i^{\kappa_G}$, and $\xi_j, \xi_j' \in \Xi_j$:*

$$J_j(\xi_j', \xi_{-j}) - J_j(\xi_j, \xi_{-j}) = \Phi_i(\xi_j', \xi_{-j}) - \Phi_i(\xi_j, \xi_{-j}). \quad (2)$$

Due to the boundedness of the reward function and Eq. (2), we can show that the local potential functions are uniformly bounded from above and below, i.e., there exist $\Phi_{\min}, \Phi_{\max} > 0$ such that $\Phi_i(\xi) \in [\Phi_{\min}, \Phi_{\max}]$ for all $i \in \mathcal{N}$ and $\xi \in \Xi$. See Appendix for the proof.

Recall that in the definition of a standard MPG, there exists a (global) potential function Φ such that Eq. (2) holds with Φ_i being replaced by Φ. Therefore, an MPG is always an NMPG (by choosing $\Phi_i = \Phi$ for all i), and hence NMPG represents a broader class of games. A concrete example of an NMPG is presented in Section 3.1.

Our goal is to design policies that learn Nash equilibria of NMPGs. Thus, we next introduce the concepts of Nash equilibrium, Nash equilibrium gap, and averaged Nash regret.

Definition 3.2 (Nash Equilibrium)**.** *A global policy $\xi = (\xi_i, \xi_{-i})$ is a Nash equilibrium (NE) if and only if*

$$J_i(\xi_i, \xi_{-i}) \geq J_i(\xi_i', \xi_{-i}), \forall\, \xi_i' \in \Xi_i, \text{ and } i \in \mathcal{N}.$$

To measure the performance of a policy by its "distance" to a Nash equilibrium, we use the Nash equilibrium gap.

Definition 3.3. *Given a global policy ξ, agent i's Nash equilibrium gap and the global Nash equilibrium gap are*

$$\textit{NE-Gap}_i(\xi) := \max_{\xi_i'} J_i(\xi_i', \xi_{-i}) - J_i(\xi_i, \xi_{-i}),$$
$$\textit{NE-Gap}(\xi) := \max_{i \in \mathcal{N}} \textit{NE-Gap}_i(\xi).$$

With NE-Gap$(\cdot)$ defined above, given $\epsilon > 0$, we say that a policy ξ is an ϵ-approximate Nash equilibrium if NE-Gap$(\xi) \leq \epsilon$. When using a softmax policy with parameter θ, we abuse the notation to denote NE-Gap$_i(\theta)$ for NE-Gap$_i(\xi^\theta)$ and also NE-Gap(θ) for NE-Gap(ξ^θ).

While Definition 3.3 enables us to measure the performance of a single policy, in MARL, most algorithms iterate over a sequence of policies. To measure the performance of a sequence of policies, we use the averaged Nash regret, which is defined in the following.

Definition 3.4. *Given a sequence of M policies $\{\xi(0), \ldots, \xi(M-1)\}$, the averaged Nash regret of agent i and the global averaged Nash regret are defined as*

$$\textit{Avg-Nash-Regret}_i(M) = \frac{1}{M} \sum_{m=0}^{M-1} \textit{NE-Gap}_i(\xi(m)),$$
$$\textit{Avg-Nash-Regret}(M) = \max_{i \in \mathcal{N}} \textit{Avg-Nash-Regret}_i(M).$$

Note that a similar concept called "Nash Regret" was introduced in Ding et al. [2022], and is defined as

$$\text{Nash-Regret}(M) = \frac{1}{M} \sum_{m=0}^{M-1} \max_{i \in \mathcal{N}} \text{NE-Gap}_i(\xi(m)). \quad (3)$$

By using Jensen's inequality and the fact that the maximum of a set of positive real numbers is less than the sum, we easily have Avg-Nash-Regret$(M) = \Theta(\text{Nash-Regret}(M))$. See Appendix for the proof. As a result, Avg-Nash-Regret(M) and Nash-Regret(M) have the same rate of convergence (up to a multiplicative constant that depends on the number of agents).

3.1 EXAMPLE

To illustrate the model, we present an extension of classical congestion games Roughgarden and Tardos [2004] and distributed welfare games [Marden and Wierman, 2013]. In this example, n agents are located on a traffic network $\mathcal{T} = (\mathcal{V}, \zeta)$, where $\mathcal{V}$ denotes the set of nodes and ζ denotes the set of directed edges with self-loops.[1] The objective of each agent i is to commute from its start node h_i to its destination d_i. Specifically, the local state $s_i(t)$ of agent i at time t is its current location (a node $v \in \mathcal{V}$). By choosing a directed edge $(v, u) \in \zeta$ as its local action $a_i(t)$ at time t, agent i will transit to state $s_i(t+1)$ at time $t+1$. Without the loss of generality, we assume an agent will stay at the same node after it arrives at its destination. The reward of agent i is defined as

$$r_i(t) = \begin{cases} 0 & \text{if } s_i(t) = d_i, \\ -\epsilon & \text{else if } s_i(t+1) = s_i(t), \\ -\epsilon - N(a_i(t), t) & \text{else,} \end{cases}$$

where $\epsilon > 0$ is a constant and $N(e, t)$ denote the number of agents that chooses edge e at time t. The reward is designed so that the agent incurs a time cost of ϵ for every step spent on its trip and a congestion cost of $N(a_i(t), t)$ depending on the traffic on the edge it travels through. The congestion cost is avoided if the agent chooses to wait at its current location ($s_i(t+1) = s_i(t)$). Each agent wants to maximize its expected discounted cumulative reward $\mathbb{E}\left[\sum_{t=0}^{\infty} \gamma^t r_i(t)\right]$.

To see that this congestion game fits in our NMPG framework, consider a communication network $\mathcal{G}$ such that that agents i and j are neighbors if and only if

$$\sum_{t=0}^{\infty} \mathbb{P}(s_i(t) = s_j(t), s_i(t) \neq d_i, s_j(t) \neq d_j) > 0.$$

Under this communication network, the transition kernel is completely local because the next state of any agent i is decided completely locally and the local reward of agent i is a function that depends on the 1-hop local states and actions $(s_{N_i^1}, a_{N_i^1})$. We provide more discussion of this example, and numerical examples using it in the Appendix.

4 ALGORITHM DESIGN

We now present a novel algorithm for solving NMPGs. Our approach uses a combination of independent policy gradient (IPG) with localized TD-learning to form a localized actor-critic framework.

4.1 ACTOR: INDEPENDENT POLICY GRADIENT

Suppose that the agents have complete knowledge about the underlying model (e.g. reward function and transition dynamics). Then a popular approach for solving MPGs is

[1] Note that the traffic network and the communication network $\mathcal{G}$ may be different.

to use IPG, which is presented in Algorithm 1 [Leonardos et al., 2022, Zhang et al., 2022b, Ding et al., 2022, Fox et al., 2022b, Zhang et al., 2022a].

Algorithm 1 Independent Policy Gradient

1: **Input:** step size $\beta > 0$, $\theta_i(0) = 0, \forall\, i \in \mathcal{N}$.
2: **for** $m = 0, 1, 2, \cdots, M-1$ **do**
3: $\quad \theta_i(m+1) = \theta_i(m) + \beta \nabla_{\theta_i} J_i(\theta(m))$ for all $i \in \mathcal{N}$
4: **end for**

In each round of Algorithm 1, each agent simultaneously updates its policy by implementing gradient ascent (in the policy space) with respect to their own objective function (see Line 3). Notably, to carry out Algorthm 1, each agent only needs to know its own policy.

While Algorithm 1 is promising, it is not a model-free algorithm as computing the gradient requires knowledge of the underlying MDP model. This motivates the design of a critic to help estimate the gradient.

4.2 CRITIC: LOCALIZED TD(λ) WITH LINEAR FUNCTION APPROXIMATION

To motivate the algorithm design of the critic, we first present an explicit expression of the policy gradient of agent i [Sutton et al., 1999]:

$$\nabla_{\theta_i} J_i(\theta) = \sum_{t=0}^{\infty} \gamma^t \mathbb{E}_{\xi^\theta} \big[\nabla_{\theta_i} \log \xi_i^{\theta_i}(a_i(t)|s_i(t)) \times \overline{Q}_i^{\theta}(s(t), a_i(t)) \big]. \tag{4}$$

Similar versions of policy gradient theorems under different multi-agent settings were previously introduced in Zhang et al. [2022a], Mao et al. [2022]. For completeness, we present a proof of (4) in the Appendix.

In view of Eq. (4), to estimate $\nabla_{\theta_i} J_i(\theta)$, the key is to construct an estimate of the averaged Q-function $\overline{Q}_i^{\theta}$. However, directly estimating the averaged Q-function of agent i requires information about the global state and action, incurring long-distance communication. To localize the algorithm, we introduce the κ-truncated averaged Q-function.

Truncated Averaged Q-functions. Given a non-negative integer κ, agent i's κ-truncation operator $T_\kappa^i : \mathbb{R}^{|\mathcal{S}||\mathcal{A}_i|} \mapsto \mathbb{R}^{|\mathcal{S}_{N_i^\kappa}||\mathcal{A}_i|}$ is defined as

$$T_\kappa^i \overline{Q}_i^{\theta}(s_{N_i^\kappa}, a_i) = \sum_{s_{-N_i^\kappa}} u_i(s_{-N_i^\kappa}) \overline{Q}_i^{\theta}(s_{N_i^\kappa}, s_{-N_i^\kappa}, a_i) \tag{5}$$

for all $(s_{N_i^\kappa}, a_i)$, where where $\{u_i(s_{-N_i^\kappa})\}$ are non-negative weights that sum up to 1.

Note that when $\kappa \geq \max_{i,j} \mathrm{dist}(i,j)$, there is essentially no truncation, and hence we have $T_\kappa^i \overline{Q}_i^{\theta} = \overline{Q}_i^{\theta}$. When $\kappa < \max_{i,j} \mathrm{dist}(i,j)$, we have the following *exponential-decay property*, the proof of which is presented in the Appendix.

Lemma 4.1. *It holds that*

$$\max_{i,s,a_i} \left| T_\kappa^i \overline{Q}_i^{\theta}(s_{N_i^\kappa}, a_i) - \overline{Q}_i^{\theta}(s, a_i) \right| \leq \frac{2}{1-\gamma} \min\left(\gamma^{\kappa - \kappa_r + 1}, 1\right).$$

Lemma 4.1 suggests that the κ-truncated averaged Q-function is a good approximation of the averaged Q-function. It is worth noting that the use of truncated Q-function and the exponential-decay property have been widely exploited in the cooperative MARL literature for communication and dimension reduction in recent years [Qu et al., 2020, Gu et al., 2022, Lin et al., 2021]. Here, we show how to use such an approach in a noncooperative setting for the first time.

Linear Function Approximation Architecture. While using the κ-truncated Q-functions enables us to overcome the computational bottleneck as the number of agents increases, there is still the challenge due to the curse of dimensionality. To further reduce the parameter dimension, we use linear function approximation. To be specific, for each $i \in \mathcal{N}$, let $\phi_i : \mathcal{S}_{N_i^{\kappa_c}} \times \mathcal{A}_i \to \mathbb{R}^{d_i}$ be a feature mapping of agent i. Then, with weight vector $w_i \in \mathbb{R}^{d_i}$, we consider approximating the κ-truncated Q-functions using

$$\hat{Q}_i(s_{N_i^{\kappa_c}}, a_i, w_i) = \langle \phi_i(s_{N_i^{\kappa_c}}, a_i), w_i \rangle, \ \forall\, (s_{N_i^\kappa}, a_i).$$

Since the state-action space is finite, we assume without loss of generality that $\max_{i, s_{N_i^\kappa}, a_i} \|\phi_i(s_{N_i^{\kappa_c}}, a_i)\| \leq 1$.

Let $\tilde{\phi}_i(s,a) = \phi_i(s_{N_i^{\kappa_c}}, a_i)$ for any $i \in \mathcal{N}$, $s \in \mathcal{S}$, and $a \in \mathcal{A}$, and let $\Omega_i \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}_i| \times d_i}$ be the feature matrix of agent i defined as:

$$\Omega_i := \begin{bmatrix} — & \tilde{\phi}_i^\top(s^1, a^1) & — \\ \vdots & \vdots & \vdots \\ — & \tilde{\phi}_i^\top(s^{|\mathcal{S}|}, a^{|\mathcal{A}|}) & —. \end{bmatrix} \tag{6}$$

Without loss of generality, we assume the feature matrix Ω_i has linearly independent columns for all $i \in \mathcal{N}$.

We propose a novel policy evaluation algorithm called localized TD(λ) with linear function approximation, which is presented in the Appendix. The algorithm can be viewed as an extension of the classical TD(λ) with linear function approximation [Tsitsiklis and Van Roy, 1996] to the case where we estimate the κ_c-truncated Q-functions.

Importantly, we use ϵ-exploration policies to ensure exploration in localized TD(λ). Specifically, suppose the critic algorithm has an input policy ξ (or ξ^θ when using softmax parametrization). The algorithm uses an ϵ-exploration version of $\hat{\xi}$ (or $\hat{\xi}^\theta$) to collect samples, where $\hat{\xi} \in \Xi^\epsilon$ (or $\hat{\xi}^\theta \in \Xi^\epsilon$) is nothing but a policy that is a convex combination of a uniformly random policy and ξ (or ξ^θ), with parameter ϵ.

Algorithm 2 Localized Actor-Critic

1: **Input**: Non-negative integers H, M, $\kappa_c \geq \kappa_r$, and a positive real number $\epsilon > 0$, initializations $\theta_i(0) = 0$ for all i, and $\Delta_{i,0}(m) = 0$ for all i and m.
2: **for** $m = 0, 1, 2, \cdots, M-1$ **do**
3: All agents simultaneously execute localized TD(λ) with linear function approximation to estimate their κ_c-truncated averaged Q-function $T^i_{\kappa_c}\overline{Q}^{\theta(m)}_i$, $i \in \mathcal{N}$, and output weight vectors $\{w^m_i\}_{i\in\mathcal{N}}$. ▷ CRITIC UPDATE
4: **for** $t = 0, 1, \cdots, H-1$ **do**
5: Each agent i samples $a^m_i(t) \sim \xi^{\theta_i(m)}_i(\cdot \mid s^m_i(t))$ simultaneously, and observes the successor state $s^m_{N^{\kappa_c}_i}(t+1)$.
6: $\Delta_{i,t+1}(m) = \Delta_{i,t}(m) + \gamma^t \nabla_{\theta_i} \log \xi^{\theta_i(m)}_i(a^m_i(t)|s^m_i(t))\langle \phi_i(s^m_{N^{\kappa_c}_i}(t), a^m_i(t)), w^m_i\rangle$
7: **end for**
8: $\theta_i(m+1) = \theta_i(m) + \beta\Delta_{i,H}(m)$ ▷ ACTOR UPDATE
9: **end for**

4.3 LOCALIZED ACTOR-CRITIC

Combining IPG with localized TD(λ), we arrive at a localized actor-critic algorithm for solving NMPGs, which is presented in Algorithm 2.

The algorithm consists of three major steps. First, in Line 3, each agent calls localized TD(λ) with linear function approximation for policy evaluation and outputs a weight vector w^m_i for all $i \in \mathcal{N}$. Then, in Lines 4 – 7, each agent uses the averaged Q-function estimate to iteratively construct an estimate of the independent policy gradient. Finally, in Line 8, using the estimated gradient, each agent implements an approximate version of IPG presented in Algorithm 1.

Compared with Algorithm 1, Algorithm 2 has the following strengths: (1) the algorithm is model-free, (2) due to the use of truncated Q-functions, each agent only requires information from its κ_c-hop neighborhood to carry out the algorithm, which eliminates long-distance communication along the network, and (3) the algorithm, to some extent, overcomes the curse of dimensionality due to the use of linear function approximation.

5 ALGORITHM ANALYSIS

We now present the main results of the paper. We formally state our assumptions in Section 5.1 and then present convergence bounds for Algorithms 1 and 2 in Section 5.2. A proof sketch of our main theorems is given in Section 5.3.

5.1 ASSUMPTIONS

We make the following assumptions in our main results. The first assumption captures the impact of network structure.

Assumption 5.1. *There exists a decreasing function $\nu : \mathbb{N} \to \mathbb{R}^+$ such that the following holds for all $\kappa \in \mathbb{N}$:*

$$\left|\Phi_i(\theta_{N^\kappa_i}, \theta'_{-N^\kappa_i}) - \Phi_i(\theta_{N^\kappa_i}, \theta_{-N^\kappa_i})\right| \leq \nu(\kappa) \max_{j\in -N^\kappa_i} \left\|\theta'_j - \theta_j\right\|, \tag{7}$$

where $\Phi_i(\theta)$ is the short-hand notation for $\Phi_i(\xi^\theta)$.

Assumption 5.1 captures the idea that, for each agent, its potential function is less impacted by the agents far away. In the extreme case where κ exceeds the diameter $\max_{i,j} \text{dist}(i,j)$ of the network, we have $\nu(\kappa) = 0$.

Our next two assumptions ensure exploration in our policies. Denote $d^\theta = d^{\xi^\theta} \in \Delta(\mathcal{S})$ as the discounted state visitation distribution under a softmax policy ξ^θ.

Assumption 5.2. $\inf_\theta \min_{s\in\mathcal{S}} d^\theta(s) > 0$.

Assumption 5.2 states that every state can be visited with positive probability under any policy, which easily holds when the initial state distribution $\mu(\cdot)$ is supported on the entire state-space. This assumption is standard and has been used in, e.g., [Zhang et al., 2022a, Agarwal et al., 2021, Mei et al., 2020]). Under Assumption 5.2, we define $D = 1/\inf_\theta \min_{s\in\mathcal{S}} d^\theta(s)$, which is finite.

Assumption 5.3. *There exists a joint policy $\xi \in \Xi$ such that the Markov chain $\{s(t)\}$ induced by ξ is irreducible and aperiodic.*

Under Assumption 5.3, [Zhang et al., 2022c, Lemma 4] implies a uniform exploration property for the Markov chain $\{(s(t), a(t))\}$ induced by any policy with entries bounded away from zero, which includes ϵ-exploration policy. Therefore, for any $\hat{\xi} \in \Xi^\epsilon$, the Markov chain $\{(s(t), a(t))\}$ induced by $\hat{\xi}$ has a unique stationary distribution, denoted by $\overline{\pi}^{\hat{\xi}} \in \Delta(\mathcal{S}\times\mathcal{A})$, which satisfies $\pi_{\min} := \inf_{\hat{\xi}\in\Xi^\epsilon} \min_{i\in\mathcal{N}} \min_{s_{N^{\kappa_c}_i}, a_i} \overline{\pi}^{\hat{\xi}}(s_{N^{\kappa_c}_i}, a_i) > 0$. This is important for our analysis of the critic.

To state our final assumption, let $D^{\hat{\xi}} \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|\times|\mathcal{S}||\mathcal{A}|}$ be a diagonal matrix with diagonal entries $\{\overline{\pi}^{\hat{\xi}}(s,a)\}_{(s,a)\in\mathcal{S}\times\mathcal{A}}$. Since $D^{\hat{\xi}}$ has strictly positive diagonal entries, and the feature matrix Ω_i has linearly independent columns for all i,

we have $\underline{\lambda} := \min_{i\in\mathcal{N}} \inf_{\hat{\xi}\in\Xi^\epsilon} \lambda_{\min}(\Omega_i D^{\hat{\xi}} \Omega_i) > 0$, where $\lambda_{\min}(\cdot)$ returns the smallest eigenvalue of a positive definite matrix. For any $i \in \mathcal{N}$ and $\theta \in \mathbb{R}^{|\mathcal{S}||\mathcal{A}|}$, let

$$c_i(\theta) := \min_s \sum_{a_i^* \in \arg\max_{a_i} \overline{Q}_i^\theta(s,a_i)} \xi_i^{\theta_i}(a_i^*|s_i).$$

Assumption 5.4. $c := \inf_\theta \min_{i\in\mathcal{N}} c_i(\theta) > 0$.

The inequality stated in Assumption 5.4 is called a non-uniform Łojasiewicz inequality, e.g., see [Zhang et al., 2022a, Mei et al., 2020], and is standard. It is used to connect the NE-Gap with the gradient of the objective function through gradient domination. For IPG (cf. Algorithm 1), this assumption is automatically satisfied [Zhang et al., 2022a]. However, for Algorithm 2, due to the more challenging model-free setup, $\inf_\theta \min_{i\in\mathcal{N}} c_i(\theta)$ is not necessarily strictly positive, which motivates Assumption 5.4. An approach for removing Assumption 5.4 is to regularize the problem (e.g., using log-barrier regularization like in Zhang et al. [2022b]), which prevents the policy generated by IPG from being deterministic, albeit at a cost of introducing an asymptotic bias due to regularization.

5.2 RESULTS

We are now ready to present our main results, which are finite-time/sample bounds. We first present the averaged Nash-regret bound of the IPG algorithm (cf. Algorithm 1) as a warm-up, then we present the finite-sample bound of Algorithm 2, which involves a critic error. Finally, we present a concise bound of critic error when using our localized TD(λ) with linear function approximation.

Given an integer κ, let $n(\kappa) := \max_{i\in\mathcal{N}} |N_i^\kappa|$ be the size of the largest κ-hop neighborhood.

Theorem 5.5. *Consider $\{\theta_i(m)\}_{0\le m\le M-1}$ generated by Algorithm 1. Suppose that Assumptions 5.1, 5.2, and 5.4 are satisfied, and $\beta = \frac{(1-\gamma)^3}{6n(\kappa_G)}$, where β is the step size. Then,*

$$\begin{aligned}&\textit{Avg-Nash-Regret}(M)\\ &\le \mathcal{O}\left(\frac{D}{c}\sqrt{\frac{\max_{j\in\mathcal{N}}|\mathcal{A}_j| n(\kappa_G)(\Phi_{\max}-\Phi_{\min})}{(1-\gamma)^3 M}}\right)\\ &\quad + \mathcal{O}\left(\frac{D\sqrt{\max_{j\in\mathcal{N}}|\mathcal{A}_j|\nu(\kappa_G)}}{c(1-\gamma)}\right). \end{aligned} \tag{8}$$

The first term on the right hand side of (8) goes to zero at a rate of $\mathcal{O}(M^{-1/2})$, which matches with the existing convergence rate of IPG for solving MPGs [Zhang et al., 2022a]. Note that, unlike in existing results, the total number of agents $|\mathcal{N}|$ does not appear in the bound. Instead, we have $n(\kappa_G)$, which captures the impact of network structure. The second term on the right hand side of (8) arises because of the relaxation from MPG to MLPG (see Definition 3.1), which decreases with κ_G, and vanishes when $\kappa_G \ge \max_{i,j} \text{dist}(i,j)$.

We now move on to our result for Algorithm 2.

Theorem 5.6. *Consider $\{\theta_i(m)\}_{0\le m\le M-1}$ generated by Algorithm 2. Suppose that Assumptions 5.1-5.4 are satisfied, and $\beta = \frac{(1-\gamma)^3}{24n(\kappa_G)\sqrt{M}}$. Then,*

$$\begin{aligned}&\mathbb{E}\left[\textit{Avg-Nash-Regret}(M)\right]\\ &\le \frac{\sqrt{\max_{j\in\mathcal{N}}|\mathcal{A}_j|}D}{c}\Bigg\{\mathcal{O}\left(\frac{\sqrt{n(\kappa_G)(\Phi_{\max}-\Phi_{\min})}}{(1-\gamma)^{1.5}M^{1/4}}\right)\\ &\quad + \mathcal{O}\left(\frac{\sqrt{\nu(\kappa_G)}}{1-\gamma}\right) + \mathcal{O}\left(\frac{\sqrt{n(\kappa_G)}[1+(1-\gamma)\epsilon_{\textit{critic}}]}{(1-\gamma)^2 M^{1/4}}\right)\\ &\quad + \mathcal{O}\left(\frac{\sqrt{n(\kappa_G)}\epsilon_{\textit{critic}}^{1/2}}{(1-\gamma)^{1.5}}\right) + \mathcal{O}\left(\frac{\sqrt{n(\kappa_G)}\gamma^{H/2}}{(1-\gamma)^2}\right)\Bigg\}, \end{aligned} \tag{9}$$

where $\epsilon_{\textit{critic}}$ stands for the critic error in policy evaluation. See Appendix for the explicit definition and Theorem 5.7 for an upper bound.

The first two terms on the right hand side of (9) are analogous to the two terms on the right hand side of IPG error bounds presented in Theorem 5.5. The last three terms are approximation errors for the policy gradient, which involve the error in the critic and the error in using a finite sum (Algorithm 2 Lines 4 – 7) to approximate an infinite sum (cf. Eq. (4)).

Due to the randomness in both the actor and the critic, unlike in Theorem 5.5, we need to use a smaller step size to control the stochastic error. As a result, we have an $\mathcal{O}(M^{-1/4})$ rate of convergence compared to $\mathcal{O}(M^{-1/2})$ in Theorem 5.5. This phenomenon is well-observed in stochastic approximation literature. For example, implementing gradient descent to minimize a smooth and strongly convex objective achieves geometric convergence, but stochastic gradient descent in the same setting has an optimal rate of $\mathcal{O}(1/M)$.

In single-agent RL, popular algorithms such as Q-learning and natural actor-critic are known to achieve $\mathcal{O}(1/\sqrt{M})$ rate of convergence [Qu and Wierman, 2020, Lan, 2022]. While we study the more challenging setting of using localized algorithms to solve MARL problems, it is an interesting direction to investigate whether there is a fundamental gap.

As a complement to Theorem 5.6, we next provide the critic error bound when using localized TD(λ) with linear function approximation.

Theorem 5.7. *Consider $\{w_i(K)\}_{i\in\mathcal{N}}$ (where K is the total iteration number) generated by localized TD(λ) with linear function approximation. Suppose that Assumption 5.3 is satisfied. Then, with appropriately chosen step size α and λ (see Appendix for the explicit requirements) and large*

enough K, we have

$$\epsilon_{critic} \leq \mathcal{O}(1-(1-\gamma)\underline{\lambda}\alpha)^{\frac{K}{2}} + \mathcal{O}\left[\frac{\alpha\log(1/\alpha)}{(1-\gamma)\underline{\lambda}}\right]^{1/2} + \mathcal{O}\left(\frac{\epsilon_{app}}{\pi_{\min}(1-\gamma)}\right) + \mathcal{O}\left(\frac{\gamma^{\kappa_c-\kappa_r}}{1-\gamma}\right) + \mathcal{O}\left(\frac{n\epsilon}{(1-\gamma)^2}\right), \tag{10}$$

where ϵ_{app} stands for the function approximation error. See Appendix for the explicit definition.

The first two terms on the right hand side of (10) represent the convergence bias (which has geometric convergence rate) and the variance (which decreases with the step size α), and their behaviors agree with existing results on stochastic approximation [Srikant and Ying, 2019, Chen et al., 2022]. The third term arises from using linear function approximation and vanishes in the tabular setting where we use a complete basis. The fourth term represents the error between the averaged Q-function and the κ_c-truncated averaged Q-function, which is introduced to overcome the scalability issue when the number of agents increases. Note that the fourth term decays exponentially with the choice of κ_c, and vanishes when κ_c is greater than the diameter (i.e., $\max_{i,j} \text{dist}(i,j)$) of the network. The last term arises because of using ϵ-exploration behavior policies to ensure sufficient exploration (see the end of Section 4.2).

5.3 PROOF SKETCH

Analysis of the Actor. At a high level, we use a Lyapunov approach to analyze the policy update, where the potential function is a natural choice of the Lyapunov function. The key is to bound $\Phi_i(\theta(m+1)) - \Phi_i(\theta(m))$, $i \in \mathcal{N}$, in each iteration using the gradient of objective function $J_i(\cdot)$, which is related to NE-Gap of agent i through the non-uniform Łojasiewicz inequality [Zhang et al., 2022a] Mei et al. [2020]. To exploit the network structure and to remove the raw dependence on the total number of agents in the NMPG setting, instead of directly bounding $\Phi_i(\theta(m+1)) - \Phi_i(\theta(m))$, we perform the following decomposition:

$$\begin{aligned}
&\Phi_i(\theta(m+1)) - \Phi_i(\theta(m)) \\
&= \underbrace{\left[\Phi_i(\theta_{N_i^{\kappa_G}}(m+1), \theta_{-N_i^{\kappa_G}}(m)) - \Phi_i(\theta(m)\right]}_{(a)} \\
&\quad + \underbrace{\left[\Phi_i(\theta(m+1)) - \Phi_i(\theta_{N_i^{\kappa_G}}(m+1), \theta_{-N_i^{\kappa_G}}(m))\right]}_{(b)}.
\end{aligned}$$

The term (a) captures the policy change of the agents inside the κ_G-hop neighborhood of agent i, and the first step of bounding it is to use the smoothness property of the potential function, which is similar to that of Zhang et al. [2022a]. However, unlike existing analysis of IPG, we also need to bound the error in approximating the gradient, which can be decomposed into three error terms:

- e_1: error due to estimation of the averaged Q-function, which is exactly the critic error;
- e_2: error due to the randomness in the trajectory sampling (see Algorithm 2 Lines 4 – 7), which has mean zero;
- e_3: error resulted from truncating the sample trajectory at horizon H (see Algorithm 2 Lines 4 – 7), which decays exponentially with H.

Term (b) results from the policy change of agents outside the κ_G-hop neighborhood of agent i, and is a decreasing function of κ_G (cf. Assumption 5.1).

Analysis of the Critic. The critic is designed to perform policy evaluation of a softmax policy ξ^θ using localized TD(λ) with linear function approximation. Similar to Chen et al. [2022], Srikant and Ying [2019], we formulate localized TD(λ) as a stochastic approximation algorithm and again use a Lyapunov approach to establish the finite-sample bound of the difference between $w_i(K)$ and w_i^*, where w_i^* is the solution to a properly defined projected Bellman equation associated with agent i.

The challenge lies in bounding the difference between the Q-function associated with the weight vector w_i^* (denoted by $Q(w_i^*)$) and the true averaged Q-function $\overline{Q}_i^\theta$ of policy ξ^θ, which can be decomposed into a function approximation error, an error due to using ϵ-exploration policy, and an error due to truncating the averaged Q-function at its κ_c-hop neighborhood. To bound these errors, we develop a novel approach involving the construction of a "sub-chain", which is an auxiliary Markov chain with state space $\mathcal{S}_{N_i^{\kappa_c}} \times \mathcal{A}_i$. More details are provided in the Appendix.

6 CONCLUSION

We study MARL in the context of MPGs and introduce a networked structure that allows agents to learn equilibria using online local information. In particular, we develop a localized actor-critic framework for minimizing the averaged Nash regret of NMPGs. Importantly, the algorithm is scalable, i.e., it does not suffer from computational bottlenecks as the number of agents increases. Nor does it suffer from the curse of dimensionality due to the use of function approximation. We provide finite-sample convergence bounds to theoretically support our proposed algorithm and conduct numerical simulations to demonstrate its empirical effectiveness. An immediate future direction is to investigate whether there is a fundamental gap in the convergence rates between localized MARL algorithms and single-agent RL algorithms in strategic settings. It is also interesting to see if localized algorithms (with provable guarantees) can be designed to solve other classes of games beyond NMPGs.

References

Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *The Journal of Machine Learning Research*, 22(1):4431–4506, 2021.

Jalaj Bhandari, Daniel Russo, and Raghav Singal. A Finite Time Analysis of Temporal Difference Learning With Linear Function Approximation. In *Conference On Learning Theory*, pages 1691–1692, 2018.

Justin A Boyan. Least-Squares Temporal Difference Learning. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pages 49–56, 1999.

Deepayan Chakrabarti, Yang Wang, Chenxi Wang, Jurij Leskovec, and Christos Faloutsos. Epidemic thresholds in real networks. *ACM Transactions on Information and System Security (TISSEC)*, 10(4):1, 2008.

Zaiwei Chen, Sheng Zhang, Thinh T Doan, John-Paul Clarke, and Siva Theja Maguluri. Finite-sample analysis of nonlinear stochastic approximation with applications in reinforcement learning. *Automatica*, 146:110623, 2022.

W Davis Dechert and SI O'Donnell. The stochastic lake game: A numerical solution. *Journal of Economic Dynamics and Control*, 30(9-10):1569–1587, 2006.

Dongsheng Ding, Chen-Yu Wei, Kaiqing Zhang, and Mihailo Jovanovic. Independent policy gradient for large-scale markov potential games: Sharper rates, function approximation, and game-agnostic convergence. In *International Conference on Machine Learning*, pages 5166–5220. PMLR, 2022.

Roy Fox, Stephen M Mcaleer, Will Overman, and Ioannis Panageas. Independent natural policy gradient always converges in markov potential games. In *International Conference on Artificial Intelligence and Statistics*, pages 4414–4425. PMLR, 2022a.

Roy Fox, Stephen M Mcaleer, Will Overman, and Ioannis Panageas. Independent natural policy gradient always converges in Markov potential games. In *International Conference on Artificial Intelligence and Statistics*, pages 4414–4425. PMLR, 2022b.

Haotian Gu, Xin Guo, Xiaoli Wei, and Renyuan Xu. Mean-Field Multi-Agent Reinforcement Learning: A Decentralized Network Approach. *Available at SSRN 3900139*, 2022.

Tommi Jaakkola, Michael I Jordan, and Satinder P Singh. Convergence of stochastic iterative dynamic programming algorithms. In *Advances in neural information processing systems*, pages 703–710, 1994.

Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, pages 1–48, 2022.

Zachary J Lee, Tongxin Li, Steven H Low, and Sunash B Sharma. Systems and methods for adaptive ev charging, July 5 2022. US Patent 11,376,981.

Stefanos Leonardos, Will Overman, Ioannis Panageas, and Georgios Piliouras. Global convergence of multi-agent policy gradient in markov potential games. In *International Conference on Learning Representations*, 2022.

David A Levin and Yuval Peres. *Markov chains and mixing times*, volume 107. American Mathematical Soc., 2017.

Yiheng Lin, Guannan Qu, Longbo Huang, and Adam Wierman. Multi-agent reinforcement learning in stochastic networked systems. *Advances in Neural Information Processing Systems*, 34:7825–7837, 2021.

Michael L Littman. Markov games as a framework for multi-agent reinforcement learning. In *Machine learning proceedings 1994*, pages 157–163. Elsevier, 1994.

Sergio Valcarcel Macua, Javier Zazo, and Santiago Zazo. Learning parametric closed-loop policies for markov potential games. In *International Conference on Learning Representations*, 2018.

Weichao Mao, Lin Yang, Kaiqing Zhang, and Tamer Basar. On improving model-free algorithms for decentralized multi-agent reinforcement learning. In *International Conference on Machine Learning*, pages 15007–15049. PMLR, 2022.

Jason R Marden and Adam Wierman. Distributed welfare games. *Operations Research*, 61(1):155–168, 2013.

Jincheng Mei, Chenjun Xiao, Csaba Szepesvari, and Dale Schuurmans. On the global convergence rates of softmax policy gradient methods. In *International Conference on Machine Learning*, pages 6820–6829. PMLR, 2020.

Guannan Qu and Adam Wierman. Finite-time analysis of asynchronous stochastic approximation and Q-learning. In *Conference on Learning Theory*, pages 3185–3205. PMLR, 2020.

Guannan Qu, Adam Wierman, and Na Li. Scalable reinforcement learning of localized policies for multi-agent networked systems. In *Learning for Dynamics and Control*, pages 256–266. PMLR, 2020.

Tim Roughgarden and Éva Tardos. Bounding the inefficiency of equilibria in nonatomic congestion games. *Games and economic behavior*, 47(2):389–403, 2004.

Yuanyuan Shi, Guannan Qu, Steven Low, Anima Anandkumar, and Adam Wierman. Stability constrained reinforcement learning for real-time voltage control. In *2022 American Control Conference (ACC)*, pages 2715–2721. IEEE, 2022.

David Silver, Julian Schrittwieser, Karen Simonyan, Ioannis Antonoglou, Aja Huang, Arthur Guez, Thomas Hubert, Lucas Baker, Matthew Lai, Adrian Bolton, et al. Mastering the game of go without human knowledge. *nature*, 550(7676):354–359, 2017.

R. Srikant and Lei Ying. Finite-time error bounds for linear stochastic approximation and td learning. In *COLT*, 2019.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

John Tsitsiklis and Benjamin Van Roy. Analysis of temporal-diffference learning with function approximation. *Advances in neural information processing systems*, 9, 1996.

John N Tsitsiklis and Benjamin Van Roy. An Analysis of Temporal-Difference Learning with Function Approximation. *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, 42(5), 1997.

Werner Vogels, Robbert van Renesse, and Ken Birman. The power of epidemics: Robust communication for large-scale distributed systems. *SIGCOMM Comput. Commun. Rev.*, 33(1):131–135, January 2003. ISSN 0146-4833. doi: 10.1145/774763.774784.

Rick Zhang and Marco Pavone. Control of robotic mobility-on-demand systems: a queueing-theoretical perspective. *The International Journal of Robotics Research*, 35(1-3): 186–203, 2016.

Runyu Zhang, Jincheng Mei, Bo Dai, Dale Schuurmans, and Na Li. On the global convergence rates of decentralized softmax gradient play in markov potential games. In *Advances in Neural Information Processing Systems*, 2022a.

Runyu Zhang, Zhaolin Ren, and Na Li. Gradient play in stochastic games: Stationary points and local geometry. In *25th International Symposium on Mathematical Theory of Networks and Systems (MTNS 2022)*, 2022b.

Yizhou Zhang, Guannan Qu, Pan Xu, Yiheng Lin, Zaiwei Chen, and Adam Wierman. Global Convergence of Localized Policy Iteration in Networked Multi-Agent Reinforcement Learning. *Preprint arXiv:2211.17116*, 2022c.