Distributed Actor-Critic Algorithms for Multiagent Reinforcement Learning Over Directed Graphs

Dai, Pengcheng; Yu, Wenwu; Wang, He; Baldi, Simone

DOI
10.1109/TNNLS.2021.3139138

Publication date
2023

Document Version
Final published version

Published in
IEEE Transactions on Neural Networks and Learning Systems

Distributed Actor–Critic Algorithms for Multiagent Reinforcement Learning Over Directed Graphs

Pengcheng Dai, *Student Member, IEEE*, Wenwu Yu, *Senior Member, IEEE*, He Wang, and Simone Baldi, *Senior Member, IEEE*

***Abstract*— Actor–critic (AC) cooperative multiagent reinforcement learning (MARL) over directed graphs is studied in this article. The goal of the agents in MARL is to maximize the globally averaged return in a distributed way, i.e., each agent can only exchange information with its neighboring agents. AC methods proposed in the literature require the communication graphs to be undirected and the weight matrices to be doubly stochastic (more precisely, the weight matrices are row stochastic and their expectation are column stochastic). Differently from these methods, we propose a distributed AC algorithm for MARL over directed graph with fixed topology that only requires the weight matrix to be row stochastic. Then, we also study the MARL over directed graphs (possibly not connected) with changing topologies, proposing a different distributed AC algorithm based on the push-sum protocol that only requires the weight matrices to be column stochastic. Convergence of the proposed algorithms is proven for linear function approximation of the action value function. Simulations are presented to demonstrate the effectiveness of the proposed algorithms.**

***Index Terms*— Directed graph, distributed actor–critic (AC) algorithm, multiagent reinforcement learning (MARL), push-sum protocol.**

I. Introduction

REINFORCEMENT learning (RL) is a mathematical framework to describe the problem of a learner to achieve a goal by interacting with an unknown environment [1], [2]. This framework is gaining more and more attention due to its wide applicability in many fields, such as optimal control [3]–[7], board games [8], [9], smart grids [10]–[13], and cyber-physical systems [14]–[16], among others (see also references therein).

Very often, RL involves the participation of many learners, giving rise to multiagent reinforcement learning (MARL) problems [17]. The concept of MARL first appeared in [18], where the author studied a competitive RL setting with two agents, where one agent aims to maximize the long-term return and the other aims to minimize it. As a generalization of [18], Hu and Wellman [20] proposed a Nash Q-learning algorithm for multiagent game where all agents have different reward functions and each agent aims to maximize the local long-term return. The disadvantage of Nash Q-learning is that the number of Q-functions (i.e., action value functions) that each agent needs to calculate is the same as the number of agents; furthermore, the convergence of the algorithm cannot be proved analytically. In place of a competitive setting, Lauer and Riedmiller [19] proposed a cooperative MARL where all agents share a common reward function and the objective of all agents is to maximize the long-term return cooperatively; in the resulting algorithm, each agent only makes use of the local action value function to solve the problem.

With the recent development of the fields of multiagent systems [21]–[27] and distributed optimization [28], [29], many techniques from these two fields have been used in MARL problems to obtain distributed RL algorithms. Compared to a centralized RL algorithm requiring a central controller to collect the global information from every agent, distributed RL algorithms only rely on local exchange of information with neighboring agents. In general, centralized RL exhibits the issue of increasing computational costs, whereas distributed RL manages to distribute computations over the communication network to reduce the cost of computing. As a result, the design of distributed RL algorithms has gained increasing attention. A pioneering work combining average consensus protocol and Q-learning resulted in the so-called distributed $\mathcal{QD}$ algorithm [30], applicable to MARL with finite state and action spaces. However, it is well known in the RL field that the distributed $\mathcal{QD}$ algorithm with finite spaces does not scale as the dimensions of the state and action spaces increase (i.e., the curse of dimensionality issue). In order to improve the scalability of MARL, a new kind of value-based MARL paradigm, called centralized training with decentralized execution (CTDE), was proposed based on value function decomposition [31]–[35]. The CTDE mechanism has recently attracted significant attention whenever agents' policies are trained with access to global information in a centralized way and executed in a decentralized way. Inspired by CTDE, many CTDE-based MARL methods have proposed, such as VDN [31], QMIX [32], QTRAN [33], MAVEN [34], and QPLEX [35], among many others.

Manuscript received 11 January 2021; revised 19 July 2021; accepted 17 December 2021. Date of publication 11 January 2022; date of current version 6 October 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 61673107, Grant 62073076, and Grant 62073074; in part by the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence under Grant BM2017002; in part by the Double Innovation Plan under Grant 4207012004; and in part by the Special Funding for Overseas Talents under Grant 6207011901. *(Corresponding author: Wenwu Yu.)*

Pengcheng Dai and He Wang are with the School of Mathematics, Southeast University, Nanjing 210096, China (e-mail: Jldaipc@163.com; wanghe@seu.edu.cn).

Wenwu Yu is with the School of Mathematics and the School of Automation, Southeast University, Nanjing 210096, China (e-mail: wwyu@seu.edu.cn).

Simone Baldi is with the School of Mathematics, Southeast University, Nanjing 210096, China, and also with the Delft Center for Systems and Control, Delft University of Technology, 2628 CD Delft, The Netherlands (e-mail: s.baldi@tudelft.nl).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2021.3139138.

Digital Object Identifier 10.1109/TNNLS.2021.3139138

Different from the above value-based methods, policy gradient methods promise to address the stability and convergence in MARL. For large state and action spaces, a classical policy gradient method uses both value function approximation (i.e., critic) and policy parameters update (i.e., actor), which gives rise to the so-called actor–critic (AC) algorithms [2]. Accordingly, combining the advantages of distributed computation and AC, some distributed versions of AC algorithms have been proposed. Zhang *et al.* [36] proposed a fully decentralized AC algorithm where the collective goal of all agents is to maximize the globally averaged return over time-varying undirected graphs. To the best of our knowledge, this was the first provably convergent AC-based distributed MARL algorithm. Different from [36], Zhang *et al.* [37] proposed a distributed MARL algorithm using the expected policy gradient. Both the works [36] and [37] are on-policy algorithms, meaning that each agent learns only about the policy it is executing. Different from the on-policy algorithms, in off-policy algorithms, each agent can learn about a policy different from the one it is executing [38]. A distributed off-policy AC algorithm based on multiagent off-policy gradient theorem was proposed in [39]. Despite the recent advances in the field, common assumptions to these recent AC methods are that the communication graphs between agents are undirected and that the weight matrices associated with the graphs are doubly stochastic (more precisely, the weight matrices are row stochastic and their expectations are column stochastic [36], [37], [39]). An exception in this sense is [40], which considers MARL over directed graphs with column stochastic weight matrices.

In view of the recent results in MARL, it is a relevant and largely unsolved problem to design distributed AC methods with analytic convergence guarantees over more general graphs and weight matrices. In this work, we provide a solution to this problem for directed graphs with both fixed and time-varying topologies. The main contributions of this work are as follows.

1) Different from the state of the art [36], [37], [39], we are relaxing the conditions of undirected graphs and doubly stochastic weight matrices (more precisely, the weight matrices are row stochastic and their expectations are column stochastic). Two distributed AC algorithms are proposed to handle MARL over directed graphs with fixed topology and row stochastic weight matrix or directed graphs with changing topologies and column stochastic weight matrices.
2) In the case of MARL over directed graphs with fixed topology, considering that the row stochastic weight matrix of fixed directed graph may not have a left eigenvector $(1/N)\mathbf{1}^\top$, we design a new temporal difference (TD) error to ensure consensus over the normalized left Perron eigenvector of the weight matrix.
3) Different from the AC method in [40], which handles the MARL over directed graphs with column stochastic weight matrices, we propose a new distributed AC algorithm with the push-sum protocol that can handle directed graphs with time-varying topologies and column stochastic weight matrices. Interestingly, compared to [40], the time-varying topologies we consider can even give rise to disconnected graphs over finite-time intervals.

The rest of this article is organized as follows. Section II introduces some preliminary notions of graph theory and single-agent RL. The cooperative MARL problem is formulated in Section III. In Section IV, two distributed AC algorithms for MARL over directed graphs with fixed and changing topologies are proposed. Section V analyzes the convergence of the two proposed distributed AC algorithms with linear action value function approximation. Simulation results to demonstrate the effectiveness of the proposed algorithms are shown in Section VI. Conclusion and future problems are discussed in Section VII.

Throughout this article, the notations $\mathbb{R}^N$, $\mathbb{R}^{N\times M}$, I, $A^\top$, and $\|\cdot\|$ are standard. The vector with each element being 1 is denoted by $\mathbf{1}$. e_i is a vector whose ith element is 1 and others are 0. For notational simplicity, we use $\lim_t$, $\sup_t$, and $\sum_t$ to represent $\lim_{t\to\infty}$, $\sup_{t\to\infty}$, and $\sum_{t\geq 0}$, respectively. For a finite set $\mathcal{S}$, we use $|\mathcal{S}|$ to denote the cardinality of $\mathcal{S}$. $\otimes$ and $\prod$ are the Kronecker product and Cartesian product, respectively. $\mathbb{I}_{\{\cdot\}}$ is the indicator function.

II. Preliminaries

A. Preliminaries on Graph Theory

Denote a directed communication graph over N agents as $\mathcal{G} = (\mathcal{N}, \mathcal{E})$, where $\mathcal{N} = \{1, \ldots, N\}$ is the set of nodes and $\mathcal{E}$ is the set of edges. A node $i \in \mathcal{N}$ represents the agent with label i. A directed edge $e_{ij} = (j, i) \in \mathcal{E}$ represents that agent i can receive and use the information from agent j. Denote $\mathcal{N}_i^{\text{in}} = \{j | e_{ij} \in \mathcal{E}\}$ and $\mathcal{N}_i^{\text{out}} = \{j | e_{ji} \in \mathcal{E}\}$ as the in-neighborhoods and out-neighborhoods of agent i, respectively. A directed path from agent i_1 to agent i_k can be represented as a sequence of edges: $(i_1, i_2) \to (i_2, i_3) \to \cdots \to (i_{k-1}, i_k)$. A directed graph $\mathcal{G}$ is strongly connected if there exists at least one directed path from agent j to agent i for all $i, j \in \mathcal{N}$. A weight matrix $C = [c_{ij}]_{N\times N}$ associated with graph $\mathcal{G}$ gives weight to every edge (including self-edges) and satisfies $c_{ii} > 0$ for any i, $c_{ij} > 0$ for $(j, i) \in \mathcal{E}$ and $c_{ij} = 0$ otherwise. The weight matrix $C = [c_{ij}]_{N\times N}$ is row stochastic if $\sum_{j\in\mathcal{N}} c_{ij} = 1$ for all $i \in \mathcal{N}$ and column stochastic if $\sum_{i\in\mathcal{N}} c_{ij} = 1$ for all $j \in \mathcal{N}$. The weight matrix $C = [c_{ij}]_{N\times N}$ is said to be double stochastic if it is both row stochastic and column stochastic. Let us also define directed graphs with possibly changing topologies as $\mathcal{G}_t = (\mathcal{N}, \mathcal{E}_t)$, where $\mathcal{N} = \{1, \ldots, N\}$ is the set of agents and $\mathcal{E}_t$ is the (time-varying) set of edges at time t. The sequence of communication graphs $\{\mathcal{G}_t\}$ is said to be uniformly strongly connected if there exists an integer $B > 0$ such that the graph with agents set $\mathcal{N}$ and edge set $\mathcal{E}_B(k) = \bigcup_{t=kB}^{(k+1)B-1} \mathcal{E}_t$ is strongly connected for every $k \geq 0$.

B. Preliminaries on Single-Agent RL

RL formalizes the problem where agent aims to maximize a return by interacting with an unknown environment [2]. The RL problem typically relies on the Markov decision process

(MDP), which can be represented as a tuple $(\mathcal{S}, \mathcal{A}, P, R)$, where $\mathcal{S}$ and $\mathcal{A}$ are state space and action space, respectively, $P(s'|s,a):\mathcal{S}\times\mathcal{A}\times\mathcal{S}\to[0,1]$ is a state transition probability function, and $R(s,a):\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ is the expected reward function. Assume that at time step t, the state is s_t and agent executes the action a_t, and then, agent will get the instantaneous reward r_{t+1}, which takes a random variable with expected value $R(s_t,a_t)$, i.e., $R(s_t,a_t)=\mathbb{E}[r_{t+1}|s_t,a_t]$. Define the policy $\pi(s,a):\mathcal{S}\times\mathcal{A}\to[0,1]$ as a probability distribution over actions for a state. A policy π can be executed in the environment and produce sample sequence $\{s_0,a_0,r_1,s_1,a_1,r_2,\ldots\}$. When the RL problem involves large state and action spaces, the policy is typically parameterized as $\pi_\theta(s,a)$, where θ is the policy parameter. In particular, policy parameterization can take many forms, for example, according to an exponential soft-max distribution [2]

$$\pi_\theta(s,a)=\frac{\exp(f_\pi(s,a,\theta))}{\sum_{b\in\mathcal{A}}\exp(f_\pi(s,b,\theta))}$$

where $f_\pi(s,a,\theta)=\phi_\pi(s,a)^\top\theta$ and $\phi_\pi(s,a)$ is the feature vector of (s,a).

In continuing tasks, the goal of the agent is to maximize the expected time-average reward $J(\theta)$ that is represented as follows:

$$J(\theta)=\lim_{T\to\infty}\frac{1}{T}\sum_{t=0}^{T-1}\mathbb{E}\big[r_{t+1}\big]=\sum_{s\in\mathcal{S}}d_\theta(s)\sum_{a\in\mathcal{A}}\pi_\theta(s,a)R(s,a)$$

where $d_\theta(s)=\lim_{t\to\infty}\mathbb{P}(s_t=s|\pi_\theta)$ is the stationary distribution of state s under policy π_θ and satisfies $d_\theta(s')=\sum_{s\in\mathcal{S}}d_\theta(s)\sum_{a\in\mathcal{A}}\pi_\theta(s,a)P(s'|s,a)$ for all $s'\in\mathcal{S}$. Under a given policy π_θ, the quantitative evaluation of a state–action pair (s,a) (i.e., action value function) is denoted as

$$Q_{\pi_\theta}(s,a)=\sum_t\mathbb{E}\big[r_{t+1}-J(\theta)|s_0=s,a_0=a,\pi_\theta\big]$$

where $Q_{\pi_\theta}(s,a)$ is also typically parameterized as $Q(s,a;w)$ with parameter w (i.e., action value function approximation). In general, the action value function approximation can take many forms, for example, linear function approximation, i.e., $Q(s,a;w)=\phi(s,a)^\top w$, where $\phi(s,a)$ is the feature vector of state–action pair (s,a). Let us now recall the standard AC algorithm [36] based on action value function approximation at time step t

$$\begin{cases}\mu_{t+1}=(1-\beta_{w,t})\mu_t+\beta_{w,t}r_{t+1} & \text{(1a)}\\ w_{t+1}=w_t+\beta_{w,t}\delta_t\nabla_w Q(s_t,a_t;w_t) & \text{(1b)}\\ \theta_{t+1}=\theta_t+\beta_{\theta,t}A_t\psi_t & \text{(1c)}\end{cases}$$

where $\beta_{w,t},\beta_{\theta,t}>0$ are stepsizes, μ_t is the estimation of $J(\theta)$, $\delta_t=r_{t+1}-\mu_t+Q(s_{t+1},a_{t+1};w_t)-Q(s_t,a_t;w_t)$ is the TD error, $A_t=Q(s_t,a_t;w_t)-\sum_{a\in\mathcal{A}}\pi_{\theta_t}(s_t,a)Q(s_t,a;w_t)$, and $\psi_t=\nabla_\theta\log\pi_{\theta_t}(s_t,a_t)$.

III. Problem Setup

The MARL over directed graphs can be described as the networked multiagent MDP, which is characterized by a tuple $(\mathcal{S},\{\mathcal{A}^i\}_{i\in\mathcal{N}},P,\{R^i\}_{i\in\mathcal{N}},\{\mathcal{G}_t\}_{t\geq0})$, where $\mathcal{N}=\{1,\ldots,N\}$ is the set of agents, $\mathcal{S}$ is the state space shared by all the agents, and $\mathcal{A}^i$ is the local action space of agent i. Denote the joint action of all agents as $a=(a^1,\ldots,a^N)\in\mathcal{A}$, where $a^i\in\mathcal{A}^i$ and $\mathcal{A}=\prod_{i=1}^N\mathcal{A}^i$ is the joint action space of all agents. $P(s'|s,a):\mathcal{S}\times\mathcal{A}\times\mathcal{S}\to[0,1]$ is the state transition probability function and $R^i(s,a):\mathcal{S}\times\mathcal{A}\to\mathbb{R}$ is the local expected reward function of agent i. Assume that at time step t, the global state is s_t, and agents execute the joint action $a_t=(a_t^1,\ldots,a_t^N)$; each agent $i\in\mathcal{N}$ will get the local instantaneous reward r_{t+1}^i, which takes a random variable with expected value $R^i(s_t,a_t)$, i.e., $R^i(s_t,a_t)=\mathbb{E}[r_{t+1}^i|s_t,a_t]$. $\mathcal{G}_t=(\mathcal{N},\mathcal{E}_t)$ represents a possibly time-varying directed graphs that describe the information interaction between agents at time step t. Denote $\pi^i(s,a^i):\mathcal{S}\times\mathcal{A}^i\to[0,1]$ as the local policy of agent i and $\pi(s,a)=\prod_{i\in\mathcal{N}}\pi^i(s,a^i):\mathcal{S}\times\mathcal{A}\to[0,1]$ as the joint policy of all agents. Consequently, the joint action $a_t=(a_t^1,\ldots,a_t^N)$ at time step t can be produced according to the joint policy $\pi(s_t,\cdot)$. In this article, we consider the same assumption as in [36], [37], and [39] that the state s and the joint action a are globally observable, whereas the reward r_t^i is observed only locally by agent i.

We focus on the MARL with large state space $\mathcal{S}$ and action space $\{\mathcal{A}^i\}_{i\in\mathcal{N}}$ in continuing task. Denote the parameterized local policy of agent i as $\pi_{\theta^i}^i(s,a^i)$, where $\theta^i\in\Theta^i$ is the policy parameter and $\Theta^i\subseteq\mathbb{R}^{m_i}$ is a compact set. The parameterized joint policy is denoted as $\pi_\theta(s,a)=\prod_{i\in\mathcal{N}}\pi_{\theta^i}^i(s,a^i)$, where $\theta=((\theta^1)^\top,\ldots,(\theta^N)^\top)^\top\in\Theta$ and $\Theta=\prod_{i\in\mathcal{N}}\Theta^i$. In order to understand how the multiagent MDP evolves, assume that at time step t, the global state is $s_t\in\mathcal{S}$ and each agent i executes the local action a_t^i according to a local policy $\pi_{\theta_t^i}^i(s_t,\cdot)$. After the joint action $a_t=(a_t^1,\ldots,a_t^N)$ is executed, each agent i receives the instantaneous reward r_{t+1}^i; meanwhile, the multiagent MDP shifts to a new state s_{t+1} with probability $P(s_{t+1}|s_t,a_t)$. Also, the whole process will continue to develop. For notational convenience, we denote $\pi_\theta=\prod_{i\in\mathcal{N}}\pi_{\theta^i}^i$. Before moving on, the following standard regularity assumption is made for the multiagent MDP and the policy parameterization.

Assumption 1 [36], [37], [39]: For any $\theta^i\in\Theta^i$, the policy function satisfies $\pi_{\theta^i}^i(s,a^i)>0$ for any $i\in\mathcal{N}$, $s\in\mathcal{S}$, and $a^i\in\mathcal{A}^i$. The policy $\pi_{\theta^i}^i(s,a^i)$ is continuously differentiable with respect to θ^i for all $i\in\mathcal{N}$. In addition, for any $\theta\in\Theta$, let $P^\theta\in\mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|\times|\mathcal{S}|\cdot|\mathcal{A}|}$ be the transition matrix of the state–action pair in the Markov chain induced by the joint policy π_θ, which satisfies

$$P^\theta\big(s',a'|s,a\big)=P\big(s'|s,a\big)\pi_\theta\big(s',a'\big) \tag{2}$$

for all $(s,a),(s',a')\in(\mathcal{S},\mathcal{A})$. As in [36], we assume that the Markov chain $\{(s_t,a_t)\}_{t\geq0}$ is irreducible and aperiodic under any π_θ with the stationary distribution denoted by $d_\theta(s)=\lim_{t\to\infty}\mathbb{P}(s_t=s|\pi_\theta)$.

The objective of the all agents is to collaboratively find a joint policy π_θ to maximize the globally averaged long-term return, i.e.,

$$\max_\theta J(\theta)=\lim_{T\to\infty}\frac{1}{T}\mathbb{E}\left(\sum_{t=0}^{T-1}\frac{1}{N}\sum_{i\in\mathcal{N}}r_{t+1}^i\right)$$

$$= \sum_{s\in\mathcal{S}} d_\theta(s) \sum_{a\in\mathcal{A}} \pi_\theta(s,a)\bar{R}(s,a) \tag{3}$$

where $\bar{R}(s,a) = (1/N)\sum_{i=1}^{N} R^i(s,a)$. Denote $\bar{r}_{t+1} = (1/N)\sum_{i=1}^{N} r_{t+1}^i$ as the averaged instantaneous reward generated at time step t, and the global expected action value function $Q_\theta(s,a)$ associated with state–action pair (s,a) under policy π_θ is as follows:

$$Q_\theta(s,a) = \sum_t \mathbb{E}\big[\bar{r}_{t+1} - J(\theta)|s_0 = s, a_0 = a, \pi_\theta\big]. \tag{4}$$

Meanwhile, the state value function $V_\theta(s)$ with state s satisfies $V_\theta(s) = \sum_{a\in\mathcal{A}} \pi_\theta(s,a) Q_\theta(s,a)$. Moreover, the advantage function of state–action pair (s,a) can be defined as $A_\theta(s,a) = Q_\theta(s,a) - V_\theta(s)$.

Lemma 1 [36]: For any $\theta \in \Theta$ and $i \in \mathcal{N}$, we define the local advantage function of agent i as follows:

$$A_\theta^i(s,a) = Q_\theta(s,a) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta^i}^i\big(s,a^i\big) Q_\theta\big(s,a^i,a^{-i}\big) \tag{5}$$

where a^{-i} is the action of agents except for i. The gradient of $J(\theta)$ in (3) with respect to θ^i is given by

$$\nabla_{\theta^i} J(\theta) = \sum_{s\in\mathcal{S}} d_\theta(s) \sum_{a\in\mathcal{A}} \pi_\theta(s,a)\big[A_\theta^i(s,a)\psi_{\theta^i}^i\big] \tag{6}$$

where $\psi_{\theta^i}^i = \nabla_{\theta^i} \log \pi_{\theta^i}^i(s,a^i)$.

Consider that the action value function $Q_\theta(s,a)$ can be parameterized in a centralized way as $Q(s,a;w^{\text{cen}})$ with parameters w^{cen}. Motivated by the standard AC algorithm (1), the parameters w^{cen} and θ can be updated at time step t as follows:

$$\begin{cases} \mu_{t+1}^{\text{cen}} = (1-\beta_{w,t})\mu_t^{\text{cen}} + \beta_{w,t}\bar{r}_{t+1} & \text{(7a)} \\ w_{t+1}^{\text{cen}} = w_t^{\text{cen}} + \beta_{w,t}\delta_t^{\text{cen}} \nabla_w Q\big(s_t,a_t;w_t^{\text{cen}}\big) & \text{(7b)} \\ \theta_{t+1}^i = \Gamma^i\big(\theta_t^i + \beta_{\theta,t} A_t^{\text{cen},i}\psi_t^i\big) & \text{(7c)} \end{cases}$$

where $\beta_{w,t}, \beta_{\theta,t} > 0$ are stepsizes, $\delta_t^{\text{cen}} = \bar{r}_{t+1} - \mu_t^{\text{cen}} + Q(s_{t+1},a_{t+1};w_t^{\text{cen}}) - Q(s_t,a_t;w_t^{\text{cen}})$, $A_t^{\text{cen},i} = Q(s_t,a_t;w_t^{\text{cen}}) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta_t^i}^i(s_t,a^i) Q(s_t,a^i,a_t^{-i};w_t^{\text{cen}})$, and $\psi_t^i = \nabla_{\theta^i} \log \pi_{\theta_t^i}^i(s_t,a_t^i)$. Furthermore, $\Gamma^i(\cdot) : \mathbb{R}^{m_i} \to \Theta^i \subseteq \mathbb{R}^{m_i}$ is the local projection operator of agent i that projects $\theta_t^i + \beta_{\theta,t} A_t^{\text{cen},i}\psi_t^i$ onto a compact set Θ^i.

IV. Distributed AC Algorithms

In this section, we propose two distributed AC algorithms for MARL over directed graphs with fixed and changing topologies. Different from the assumptions in [36], [37], and [39], the first distributed AC algorithm we propose only requires a row stochastic weight matrix. The second distributed AC algorithm we propose uses the framework of push-sum protocol and only requires column stochastic weight matrices [42].

A. Distributed AC Algorithm for MARL Over Directed Graph With Fixed Topology

Assumption 2: The fixed graph $\mathcal{G}$ is strongly connected and the weight matrix $C = [c_{ij}]_{N\times N}$ is row stochastic.

Each agent i maintains its own parameter w^i and uses $Q(s,a;w^i)$ as a local estimation of $Q_\theta(s,a)$. Moreover, each agent i can collect and use the parameters of its in-neighborhoods through the distributed information interaction. The distributed AC algorithm (hereafter referred to as Algorithm 1) for MARL over directed graph with fixed topology is designed as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{w,t})\mu_t^i + \beta_{w,t} r_{t+1}^i & \text{(8a)} \\ p_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij} p_t^j & \text{(8b)} \\ \tilde{w}_{t+1}^i = w_t^i + \beta_{w,t}\delta_t^i \nabla_w Q_t\big(w_t^i\big) & \text{(8c)} \\ w_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}\tilde{w}_{t+1}^i & \text{(8d)} \\ \theta_{t+1}^i = \Gamma^i\big(\theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i\big) & \text{(8e)} \end{cases}$$

where μ_t^i tracks the long-term return of agent i, $\beta_{w,t}, \beta_{\theta,t} > 0$ are the stepsizes, p_t^i is a local estimation of the normalized left Perron eigenvector $p = (p_1,\ldots,p_n)^\top$ of C with $\mathbf{1}^\top p = 1$, and the initial value of p_t^i is set as $p_0^i = e_i \in \mathbb{R}^N$. In this article, for notational convenience, we let $Q_t(w_t^i) = Q(s_t,a_t;w_t^i)$. In particular, δ_t^i is the local TD-error that is defined as $\delta_t^i = (r_{t+1}^i - \mu_t^i)(N\cdot p_{t,i}^i)^{-1} + Q_{t+1}(w_t^i) - Q_t(w_t^i)$, where $p_{t,i}^i$ is the ith element of p_t^i. $\tilde{w}_{t+1}^i$ is the intermediate of w_{t+1}^i and is not involved in the interaction of information. $A_t^i = Q_t(w_t^i) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta_t^i}^i(s_t,a^i) Q(s_t,a^i,a_t^{-i};w_t^i)$ is the advantage function of agent i and $\psi_t^i = \nabla_{\theta^i} \log \pi_{\theta_t^i}^i(s_t,a_t^i)$. The distinguishing features of the proposed algorithm can be remarked as follows.

Remark 1: The distributed AC algorithms in [36], [37], and [39] require the communication graph to be undirected and the weight matrix to be double stochastic matrix (more precisely, the weight matrices are row stochastic and their expectations are column stochastic), which is no longer applicable in directed graph with row stochastic weight matrix since $(1/N)\mathbf{1}^\top$ may not be a left eigenvector for the weight matrix $C = [c_{ij}]_{N\times N}$. $p_{t,i}^i > 0$ used in the TD error $\delta_t^i = (r_{t+1}^i - \mu_t^i)(N\cdot p_{t,i}^i)^{-1} + Q_{t+1}(w_t^i) - Q_t(w_t^i)$ is to guarantee that w_t^i converges to the consensus term $\sum_{i=1}^N p_i w_t^i$. In particular, in the TD error δ_t^i, the scaling coefficient $(N\cdot p_{t,i}^i)^{-1}$ is only set for reward term $(r_{t+1}^i - \mu_t^i)$ without setting the action value function term $Q_{t+1}(w_t^i) - Q_t(w_t^i)$, which is due to the change of reward guiding action value function.

B. Distributed AC Algorithm for MARL Over Directed Graphs With Changing Topologies

Assumption 3: The sequence $\{\mathcal{G}_t\}_{t\geq 0}$ is uniformly strongly connected. Moreover, the column stochastic weight matrix $C(t) = [c_{ij}(t)]_{N\times N}$ for all $t > 0$ is defined by

$$c_{ij}(t) = \begin{cases} 1/d_j(t), & \text{if } i \in \mathcal{N}_j^{\text{out}}(t) \\ 0, & \text{otherwise} \end{cases} \tag{9}$$

where $d_j(t)$ is the number of out-neighborhoods of agent j in $\mathcal{G}_t$.

The distributed AC algorithm (hereafter referred to as Algorithm 2) for MARL over directed graphs with changing

Algorithm 1 Distributed AC Algorithm for Directed Graph With Fixed Topology

Set the initial values of $C = [c_{ij}]_{N\times N}$, μ_0^i, w_0^i, θ_0^i for all $i \in \mathcal{N}$, the initial state s_0, and the stepsizes $\{\beta_{w,t}\}_{t\geq 0}$, $\{\beta_{\theta,t}\}_{t\geq 0}$;
Set $t = 0$, $p_0^i = e_i$ for all $i \in \mathcal{N}$;
Each agent $i \in \mathcal{N}$ observes the global state s_0, executes the local action $a_0^i \sim \pi_{\theta_0^i}^i(s_0, \cdot)$, and observes the joint action $a_0 = (a_0^1, \ldots, a_0^N)$;
repeat
  for $i \in \mathcal{N}$ **do**
    Agent i observes the reward r_{t+1}^i and the global state s_{t+1}, executes the local action $a_{t+1}^i \sim \pi_{\theta_t^i}^i(s_{t+1}, \cdot)$, and observes the joint action $a_{t+1} = (a_{t+1}^1, \ldots, a_{t+1}^N)$;
    $\mu_{t+1}^i = (1-\beta_{w,t})\mu_t^i + \beta_{w,t} r_{t+1}^i$;
    $p_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij} p_t^j$;
    $\delta_t^i = (r_{t+1}^i - \mu_t^i)(N \cdot p_{t,i}^i)^{-1} + Q_{t+1}(w_t^i) - Q_t(w_t^i)$;
    $\tilde{w}_{t+1}^i = w_t^i + \beta_{w,t}\delta_t^i \nabla_w Q_t(w_t^i)$;
    $w_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}\tilde{w}_{t+1}^i$;
    $A_t^i = Q_t(w_t^i) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta_t^i}^i(s_t, a^i) Q(s_t, a^i, a_t^{-i}; w_t^i)$;
    $\psi_t^i = \nabla_{\theta^i} \log \pi_{\theta_t^i}^i(s_t, a_t^i)$;
    $\theta_{t+1}^i = \Gamma^i(\theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i)$;
  end
  Update the iteration counter $t \leftarrow t+1$;
until *Convergence*;

Algorithm 2 Distributed AC Algorithm for Directed Graphs With Changing Topologies

Set the initial values of $C(t) = [c_{ij}(t)]_{N\times N}$, μ_0^i, $\tilde{w}_0^i$, θ_0^i for all $i \in \mathcal{N}$, the initial state s_0, and the stepsizes $\{\beta_{w,t}\}_{t\geq 0}$, $\{\beta_{\theta,t}\}_{t\geq 0}$;
Set $t = 0$, $o_0^i = 1$ for all $i \in \mathcal{N}$;
Each agent $i \in \mathcal{N}$ observes the global state s_0, executes the local action $a_0^i \sim \pi_{\theta_0^i}^i(s_0, \cdot)$, and observes the joint action $a_0 = (a_0^1, \ldots, a_0^N)$;
repeat
  for $i \in \mathcal{N}$ **do**
    Agent i observes the reward r_{t+1}^i and the global state s_{t+1}, executes the local action $a_{t+1}^i \sim \pi_{\theta_t^i}^i(s_{t+1}, \cdot)$, and observes the joint action $a_{t+1} = (a_{t+1}^1, \ldots, a_{t+1}^N)$;
    $\mu_{t+1}^i = (1-\beta_{w,t})\mu_t^i + \beta_{w,t} r_{t+1}^i$;
    $o_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t) o_t^j$;
    $\tilde{\delta}_t^i = r_{t+1}^i - \mu_t^i + Q_{t+1}(\tilde{w}_t^i) - Q_t(\tilde{w}_t^i)$;
    $\tilde{w}_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t)\tilde{w}_t^j + \beta_{w,t}\tilde{\delta}_t^i \nabla_w Q_t(\tilde{w}_t^i)$;
    $w_{t+1}^i = (1/o_{t+1}^i)\sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t)\tilde{w}_t^j$;
    $A_t^i = Q_t(w_t^i) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta_t^i}^i(s_t, a^i) Q(s_t, a^i, a_t^{-i}; w_t^i)$;
    $\psi_t^i = \nabla_{\theta^i} \log \pi_{\theta_t^i}^i(s_t, a_t^i)$;
    $\theta_{t+1}^i = \Gamma^i(\theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i)$;
  end
  Update the iteration counter $t \leftarrow t+1$;
until *Convergence*;

topologies is designed as follows:

$$\begin{cases} \mu_{t+1}^i = (1-\beta_{w,t})\mu_t^i + \beta_{w,t} r_{t+1}^i & \text{(10a)} \\ o_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t) o_t^j & \text{(10b)} \\ \tilde{w}_{t+1}^i = \sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t)\tilde{w}_t^j + \beta_{w,t}\tilde{\delta}_t^i \nabla_w Q_t(\tilde{w}_t^i) & \text{(10c)} \\ w_{t+1}^i = \frac{1}{o_{t+1}^i}\sum_{j\in\mathcal{N}_i^{\text{in}}} c_{ij}(t)\tilde{w}_t^j & \text{(10d)} \\ \theta_{t+1}^i = \Gamma^i(\theta_t^i + \beta_{\theta,t} A_t^i \psi_t^i) & \text{(10e)} \end{cases}$$

where $o_0^i = 1$ and $\tilde{\delta}_t^i = r_{t+1}^i - \mu_t^i + Q_{t+1}(\tilde{w}_t^i) - Q_t(\tilde{w}_t^i)$ for all $i \in \mathcal{N}$. $A_t^i = Q_t(w_t^i) - \sum_{a^i\in\mathcal{A}^i} \pi_{\theta_t^i}^i(s_t, a^i) Q(s_t, a^i, a_t^{-i}; w_t^i)$ is the advantage function of agent i and $\psi_t^i = \nabla_{\theta^i} \log \pi_{\theta_t^i}^i(s_t, a_t^i)$. The distinguishing features of the proposed algorithm can be remarked as follows.

Remark 2: In Algorithm 2, (10b)–(10d) are designed based on the push-sum protocol. The push-sum protocol was also used in [40] under the assumption of column stochastic weight matrices. Since the column stochastic weight matrices as in Assumption 3 cannot guarantee average consensus in general, the ratios $(1/o_{t+1}^i)\sum_{j\in\mathcal{N}} c_{ij}(t)\tilde{w}_t^j$ are introduced to track the average $(1/N)(\mathbf{1}^\top \otimes I)\tilde{w}_t$, where $\tilde{w}_t = ((\tilde{w}_t^1)^\top, \ldots, (\tilde{w}_t^N)^\top)^\top$. In order to achieve a similar convergence result as the centralized algorithm (7), we propose the different TD error in (10c) containing $\nabla_w Q_t(\tilde{w}_t^i)$ in place of the TD error in [40]. It is noted that this new TD error requires a different convergence analysis.

V. Theoretical Results

In this section, we establish theoretical convergence results for the proposed distributed AC algorithms. Specifically, the convergence proofs are given for the case of linear function approximation of the action value function, according to the following standard assumptions that are taken from [36], [37], and [39].

Assumption 4: The instantaneous reward r_t^i is uniformly bounded for any agent i and $t \geq 0$.

Assumption 5: For each agent i, the action value function is parameterized by the class of linear functions, i.e., $Q(s, a; w) = \phi(s, a)^\top w$, where $\phi(s, a) = (\phi_1(s, a), \ldots, \phi_K(s, a))^\top \in \mathbb{R}^K$ is the feature vector associated with the state–action pair (s, a). The feature vectors $\phi(s, a)$ are uniformly bounded for any $s \in \mathcal{S}, a \in \mathcal{A}$. Furthermore, the feature matrix $\Phi \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|\times K}$ whose kth column is $(\phi_k(s, a), s \in \mathcal{S}, a \in \mathcal{A})^\top$ has full column rank. In addition, for any $v \in \mathbb{R}^K$, $\Phi v \neq \mathbf{1}$.

Assumption 6: The stepsizes $\beta_{w,t}$ and $\beta_{\theta,t}$ satisfy $\beta_{w,t}, \beta_{\theta,t} > 0$, $\sum_t \beta_{w,t} = \infty$, $\sum_t \beta_{\theta,t} = \infty$, $\sum_t \beta_{w,t}^2 + \sum_t \beta_{\theta,t}^2 < \infty$, and $\beta_{\theta,t} = o(\beta_{w,t})$. In addition, $\lim_{t\to\infty} \beta_{w,t+1} \cdot \beta_{w,t}^{-1} = 1$.

Assumption 7: The joint policy parameter set $\Theta = \prod_{i=1}^N \Theta^i$ is large enough to include at least one local minimum of $J(\theta)$.

Before moving on, for notational simplicity, we define $D_\theta^{s,a} = \text{diag}[d_\theta(s)\pi_\theta(s,a), s \in \mathcal{S}, a \in \mathcal{A}] \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|\times|\mathcal{S}|\cdot|\mathcal{A}|}$ and $\bar{R} = [\bar{R}(s,a), s \in \mathcal{S}, a \in \mathcal{A}]^\top \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$. The Bellman operator $\mathcal{T}_\theta : \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|} \to \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$ is denoted as

$$\mathcal{T}_\theta(Q) = \bar{R} - J(\theta)\mathbf{1} + P^\theta Q \tag{11}$$

where $Q \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$ is the action value function. Define the vector $\hat{\Gamma}^i(\cdot)$ as

$$\hat{\Gamma}^i\big(g^i(\theta)\big) = \lim_{0<\eta\to 0}\big\{\Gamma^i\big(\theta^i + \eta g^i(\theta)\big) - \theta^i\big\}/\eta \tag{12}$$

where $g^i{:}\Theta \to \mathbb{R}^{m_i}$ is a continuous function.

A. Convergence of Algorithm 1

Denote an increasing σ-algebra $\mathcal{F}_t$ as the filtration with $\mathcal{F}_t = \sigma(r_\tau, \mu_\tau, w_\tau, s_\tau, a_\tau, \tau \le t)$, where $r_\tau = (r_\tau^1, \ldots, r_\tau^N)^\top$, $\mu_\tau = (\mu_\tau^1, \ldots, \mu_\tau^N)^\top$, and $w_\tau = ((w_\tau^1)^\top, \ldots, (w_\tau^N)^\top)^\top$. Before moving on, the following lemmas are presented.

Lemma 2 [36]: Under Assumptions 1 and 4, the sequence $\{\mu_t^i\}$ generated from (8a) is bounded almost surely (a.s.), i.e., $\sup_t |\mu_t^i| < \infty$ a.s. for all $i \in \mathcal{N}$.

Proof: Define $h^i(\mu_t^i, s_t, a_t) = -\mu_t^i + \mathbb{E}[r_{t+1}^i|\mathcal{F}_t]$. It is obvious that $h^i(\mu_t^i, s_t, a_t)$ is Lipschitz continuous in μ_t^i. According to Assumption 1, $\{(s_t, a_t)\}_{t\ge 0}$ is an irreducible and aperiodic Markov chain. In addition, by Assumption 6, the stepsize $\beta_{w,t}$ satisfies $\sum_t \beta_{w,t} = \infty$ and $\sum_t \beta_{w,t}^2 < \infty$. Since r_t^i is uniformly bounded by Assumption 4, we can get that $\mathbb{E}[|r_{t+1}^i - \mathbb{E}[r_{t+1}^i|\mathcal{F}_t]|^2|\mathcal{F}_t] \le K_0^i(1+|\mu_t^i|^2)$ for some $K_0^i < \infty$. Hence, the conditions (a.1)–(a.5) in Appendix A are satisfied. Define $\bar{h}^i(\mu^i) = -\mu^i + \sum_{s\in\mathcal{S}} d_\theta(s)\sum_{a\in\mathcal{A}}\pi_\theta(s,a)R^i(s,a)$. According to the stochastic approximation results in Appendix A, the asymptotic behavior of (8a) can be captured by the ordinary differential equation (ODE): $\dot{\mu}^i = \bar{h}^i(\mu^i)$. Define $\bar{h}_c^i(\mu^i) = \bar{h}^i(c\mu^i)/c$, and then, $h_\infty^i(\mu^i) = \lim_{c\to\infty}\bar{h}_c^i(\mu^i) = -\mu^i$. By Lemma 11, it can be concluded that $\sup_t |\mu_t^i| < \infty$ a.s. for all $i \in \mathcal{N}$. ∎

Lemma 3: Under Assumptions 1 and 4–6, the sequence $\{w_t^i\}$ generated from (8d) is bounded a.s., i.e., $\sup_t |w_t^i| < \infty$ a.s. for all $i \in \mathcal{N}$.

Proof: The proof is along similar lines as that in [36, Lemma 5.1]. ∎

Lemma 4 [41]: Under Assumption 2, $\lim_t C^t = \mathbf{1}p^\top$, where $p = (p_1, \ldots, p_N)^\top \in \mathbb{R}^N > 0$ is the normalized left Perron eigenvector of C. For each $\lambda \in (\lambda_2(C), 1)$, there exists $M_1 > 0$ such that $|C_{ji}^t - p_i| \le M_1\lambda^t$ and $|p_{t,i}^i - p_i| \le M_1\lambda^t$ for any $i, j \in \mathcal{N}$ and $t > 0$, where $\lambda_2(C)$ is the second largest eigenvalue of C, C_{ji}^t is the jth row and ith column element in C^t and $p_{t,i}^i$ is the ith element of p_t^i. Furthermore, there exists $\eta_1 > 0$ satisfying $\eta_1^{-1} \le p_{t,i}^i \le 1$ for any $i \in \mathcal{N}$ and $t \ge 0$.

Lemma 5 [28]: Let $0 < \beta < 1$ and $\{\gamma_k\}$ be a positive scalar sequence, which satisfies $\lim_k \gamma_k = 0$. Then, $\lim_k \sum_{l=0}^k \beta^{k-l}\gamma_l = 0$.

In Algorithm 1, we are concerned with demonstrating the convergence result for w_t^i and θ_t^i for all $i \in \mathcal{N}$. In order to demonstrate the convergence of w_t^i, we first establish the asymptotic consensus of w_t^i for all $i \in \mathcal{N}$. Note that the update of $w_t = ((w_t^1)^\top, \ldots, (w_t^N)^\top)^\top \in \mathbb{R}^{KN}$ in Algorithm 1 can be rewritten in a compact form as follows:

$$w_{t+1} = (C \otimes I)\big(w_t + \beta_{w,t}y_{t+1}\big) \tag{13}$$

where $y_{t+1} = (\delta_t^1\phi_t^\top, \ldots, \delta_t^N\phi_t^\top)^\top \in \mathbb{R}^{KN}$ and $\phi_t = \phi(s_t, a_t)$. Denote the operator $\langle\cdot\rangle : \mathbb{R}^{KN} \to \mathbb{R}^K$ as follows:

$$\langle w\rangle = \big(p^\top \otimes I\big)w = \sum_{i\in\mathcal{N}} p_i w^i \tag{14}$$

where $w^i \in \mathbb{R}^K$ for any $i \in \mathcal{N}$ and $w = ((w^1)^\top, \ldots, (w^N)^\top)^\top \in \mathbb{R}^{KN}$. Denote $\mathcal{J} = (\mathbf{1}p^\top) \otimes I$ and $\mathcal{J}_\perp = I - \mathcal{J}$, and we can obtain that $\mathcal{J}w = \mathbf{1} \otimes \langle w\rangle$ and $w_\perp \triangleq \mathcal{J}_\perp w = w - \mathbf{1} \otimes \langle w\rangle$.

Let $z_t^i = (\mu_t^i, (w_t^i)^\top)^\top$ and $z_t = ((z_t^1)^\top, \ldots, (z_t^N)^\top)^\top$. By Lemmas 2 and 3, we have $\mathbb{P}(\sup_t \|z_t\| < \infty) = 1$. Hence, it is sufficient to show that $\lim_t \|w_t^i - \langle w_t\rangle\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} = 0$ for any $M > 0$, to establish the asymptotic consensus of w_t^i for all $i \in \mathcal{N}$.

Lemma 6: Under Assumptions 2 and 4–6, for any $i \in \mathcal{N}$, we have $\lim_t w_t^i - \langle w_t\rangle = 0$ a.s.

Proof: From (13), for each agent i, we have $w_{t+1}^i = \sum_{j\in\mathcal{N}} c_{ij}w_t^i + \varepsilon_t^i$, where $\varepsilon_t^i = \beta_{w,t}\sum_{j\in\mathcal{N}} c_{ij}((r_{t+1}^i - \mu_t^i)(N \cdot p_{t,i}^i)^{-1} + \phi_{t+1}^\top w_t^i - \phi_t^\top w_t^i)\phi_t$ is an error term. As a result,

$$w_t^i = \sum_{j\in\mathcal{N}} C_{ij}^t w_0^j + \sum_{l=0}^{t-1}\sum_{j\in\mathcal{N}} C_{ij}^{t-1-l}\varepsilon_l^j.$$

According to the fact that $p^\top C = p^\top$, we can obtain

$$\langle w_t\rangle = \sum_{j\in\mathcal{N}} p_j w_0^j + \sum_{l=0}^{t-1}\sum_{j\in\mathcal{N}} p_j\varepsilon_l^j.$$

As a result, it can be obtained that $w_t^i - \langle w_t\rangle = \sum_{j\in\mathcal{N}}(C_{ij}^t - p_j)w_0^j + \sum_{l=0}^{t-1}\sum_{j\in\mathcal{N}}(C_{ij}^{t-1-l} - p_j)\varepsilon_l^j$. By Lemma 4, the following holds for any $i \in \mathcal{N}$ and $t \ge 0$:

$$\begin{aligned}
&\big\|w_t^i - \langle w_t\rangle\big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \\
&\le \sum_{j\in\mathcal{N}}\big|C_{ij}^t - p_j\big|\Big\|w_0^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \\
&\quad + \sum_{l=0}^{t-1}\sum_{j\in\mathcal{N}}\Big|C_{ij}^{t-1-l} - p_j\Big|\Big\|\varepsilon_l^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \\
&\le \sum_{j\in\mathcal{N}} M_1\lambda^t\Big\|w_0^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \\
&\quad + \sum_{l=0}^{t-1}\sum_{j\in\mathcal{N}} M_1\lambda^{t-1-l}\Big\|\varepsilon_l^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}}.
\end{aligned} \tag{15}$$

By Assumptions 4 and 5 and Lemma 4, r_{t+1}^i and ϕ_t are uniformly bounded for any $i \in \mathcal{N}$ and $t \ge 0$, and $1/p_{t,i}^i \le \eta_1$, which can guarantee that $\|\varepsilon_l^j\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}}$ is bounded for any $M > 0$, i.e., there exists $K_1 < \infty$ such that

$$\sum_{j\in\mathcal{N}}\Big\|\varepsilon_l^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \le \beta_{w,l}K_1 \quad \forall\, l \ge 0.$$

As a result, we can obtain

$$\big\|w_t^i - \langle w_t\rangle\big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} \le \sum_{j\in\mathcal{N}} M_1\lambda^t\Big\|w_0^j\Big\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}}$$

$$+\sum_{l=0}^{t-1} K_1 M_1 \lambda^{t-1-l}\beta_{w,l}. \quad (16)$$

By Assumption 6 and Lemmas 2–5, we have $\lim_t \|w_t^i - \langle w_t\rangle\|\mathbb{I}_{\{\sup_t \|z_t\|\le M\}} = 0$ a.s. Consider that $\{\sup_t \|z_t\| \le \infty\}$ holds with probability 1, and it can be obtained that $\lim_t w_t^i - \langle w_t\rangle = 0$ a.s. ∎

Lemma 7: Under Assumptions 1 and 4–6, we have $\lim_t \bar{\mu}_t = J(\theta)$ and $\lim_t \langle w_t\rangle = w_\theta$ a.s.

Proof: According to the update of (13), the iteration of $\langle w_t\rangle$ has the form

$$\langle w_{t+1}\rangle = \langle w_t\rangle + \beta_{w,t}\langle y_{t+1}\rangle. \quad (17)$$

The updates for $\bar{\mu}_t$ and $\langle w_t\rangle$ are as follows:

$$\begin{cases} \bar{\mu}_{t+1} = \bar{\mu}_t + \beta_{w,t}\mathbb{E}[\bar{r}_{t+1} - \bar{\mu}_t|\mathcal{F}_t] + \beta_{w,t}\xi_{t+1,1} & (18a)\\ \langle w_{t+1}\rangle = \langle w_t\rangle + \beta_{w,t}\mathbb{E}[\bar{\delta}_t\phi_t|\mathcal{F}_t] + \beta_{w,t}\xi_{t+1,2} \\ \qquad\qquad +\beta_{w,t}\xi_{t+1,3} & (18b)\end{cases}$$

where $\bar{\mu}_t = (1/N)\sum_{i\in\mathcal{N}}\mu_t^i$ and $\bar{\delta}_t = (1/N)\sum_{i\in\mathcal{N}}(r_{t+1}^i - \mu_t^i) + \sum_{i\in\mathcal{N}} p_i(\phi_{t+1}-\phi_t)^\top w_t^i$. Moreover, $\xi_{t+1,1}$, $\xi_{t+1,2}$, and $\xi_{t+1,3}$ are defined as follows:

$$\begin{cases} \xi_{t+1,1} = \bar{r}_{t+1} - \mathbb{E}[\bar{r}_{t+1}|\mathcal{F}_t] & (19a)\\ \xi_{t+1,2} = \bar{\delta}_t\phi_t - \mathbb{E}[\bar{\delta}_t\phi_t|\mathcal{F}_t] & (19b)\\ \xi_{t+1,3} = \dfrac{1}{N}\displaystyle\sum_{i\in\mathcal{N}}\Big(p_i\big(p_{t,i}^i\big)^{-1} - 1\Big)\big(r_{t+1}^i - \mu_t^i\big)\phi_t. & (19c)\end{cases}$$

Recall that, from the definition of $\bar{\delta}_t$ in (18b), we have

$$\bar{\delta}_t = \bar{r}_{t+1} - \bar{\mu}_t + (\phi_{t+1}-\phi_t)^\top\langle w_t\rangle.$$

Since $\mathbb{E}[\bar{r}_{t+1} - \bar{\mu}_t|\mathcal{F}_t]$ is Lipschitz continuous in $\bar{\mu}_t$, we can obtain that $\mathbb{E}[\bar{\delta}_t\phi_t|\mathcal{F}_t]$ is Lipschitz continuous in both $\bar{\mu}_t$ and $\langle w_t\rangle$.

Consider that $\xi_{t+1,1}$ is a martingale difference sequence and $\bar{r}_{t+1}$ is uniformly bounded, and we can obtain

$$\mathbb{E}\big[\|\xi_{t+1,1}\|^2|\mathcal{F}_t\big] \le K_2\big(1 + \|\bar{\mu}_t\|^2 + \|\langle w_t\rangle\|^2\big)$$

for some $K_2 < \infty$ [36]. By the definition of $\xi_{t+1,2}$ in (19b), it is obvious that $\xi_{t+1,2}$ is also a martingale difference sequence and satisfies

$$\mathbb{E}\big[\|\xi_{t+1,2}\|^2|\mathcal{F}_t\big] \le 2\mathbb{E}\big[\|\bar{\delta}_t\phi_t\|^2|\mathcal{F}_t\big] + 2\|\mathbb{E}\big[\bar{\delta}_t\phi_t|\mathcal{F}_t\big]\|^2.$$

Due to the boundedness of r_t^i and ϕ_t for any $i \in \mathcal{N}$ and $t \ge 0$, there exists $K_3 < \infty$ such that $\mathbb{E}[\|\xi_{t+1,2}\|^2|\mathcal{F}_t] \le K_3(1 + \|\bar{\mu}_t\|^2 + \|\langle w_t\rangle\|^2)$ over the set $\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}$ for any $M > 0$. By Lemma 4, for any $M > 0$, it holds that

$$\begin{aligned}&\|\xi_{t+1,3}\|\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}\\ &= \frac{1}{N}\sum_{i\in\mathcal{N}}\left|\frac{p_i}{p_{t,i}^i} - 1\right|\big|r_{t+1}^i - \mu_t^i\big|\|\phi_t\|\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}\\ &\le \frac{1}{N}\sum_{i\in\mathcal{N}} M_1\eta_1\lambda^t\big|r_{t+1}^i - \mu_t^i\big|\|\phi_t\|\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}. \quad (20)\end{aligned}$$

According to the uniformly boundedness of r_t^i and ϕ_t, we can obtain that, for any $M > 0$ and $t \ge 0$, there exists a constant $K_4 > 0$ such that $\|\xi_{t+1,3}\| < K_4\lambda^t$ on the set $\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}$. This verifies that $\{\xi_{t,3}\}$ is a bounded random sequence with $\lim_t \xi_{t,3} = 0$ a.s. on the set $\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}$ for any $M > 0$.

Consider that the ODE captures the asymptotic behavior of (18a) and (18b) as follows:

$$\begin{pmatrix}\dot{\bar{\mu}}\\ \dot{\langle w\rangle}\end{pmatrix} = \begin{pmatrix}-1 & 0\\ -\Phi^\top D_\theta^{s,a}\mathbf{1} & \Phi^\top D_\theta^{s,a}\big(P^\theta - I\big)\Phi\end{pmatrix}\begin{pmatrix}\bar{\mu}\\ \langle w\rangle\end{pmatrix} + \begin{pmatrix}J(\theta)\\ \Phi^\top D_\theta^{s,a}\bar{R}\end{pmatrix}. \quad (21)$$

Recall that $D_\theta^{s,a} = \mathrm{diag}[d_\theta(s)\pi_\theta(s,a), s\in\mathcal{S}, a\in\mathcal{A}] \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$, and $\bar{R} = [\bar{R}(s,a), s\in\mathcal{S}, a\in\mathcal{A}]^\top \in \mathbb{R}^{|\mathcal{S}|\cdot|\mathcal{A}|}$. According to Assumption 1 and the Perron–Frobenius theorem in [36], the stochastic matrix P^θ has a simple eigenvalue of 1 and the remaining eigenvalues have real parts less than 1. Since Φ satisfies the full column rank condition in Assumption 5, we can obtain that all eigenvalues of $\Phi^\top D_\theta^{s,a}(P^\theta - I)\Phi$ have negative real parts, except one eigenvalue that is zero. Due to the fact that $\alpha\mathbf{1}$ ($\alpha \ne 0$) lies in the eigenspace of $D_\theta^{s,a}(P^\theta - I)$ associated with zero, it is possible for the simple eigenvalue of zero to have eigenvector υ, which satisfies $\Phi\upsilon = \alpha\mathbf{1}$. However, by Assumption 5, this will not happen with any choice of Φ since $\Phi\upsilon \ne \mathbf{1}$ for any $\upsilon \in \mathbb{R}^K$. As a result, the ODE (21) is globally asymptotically stable and has its equilibrium satisfying

$$\begin{cases} \bar{\mu} = J(\theta) & (22)\\ \Phi^\top D_\theta^{s,a}(\bar{R} - \bar{\mu}\mathbf{1} + P^\theta\Phi\langle w\rangle - \Phi\langle w\rangle) = 0. & (23)\end{cases}$$

Note that the solution for $\langle w\rangle$ has the form of $w_\theta + \alpha\upsilon$ with any $\alpha \in \mathbb{R}$ and $\upsilon \in \mathbb{R}^K$ such that $\Phi\upsilon = \mathbf{1}$. By Assumption 5 that for any $\upsilon \in \mathbb{R}^K$, $\Phi\upsilon \ne \mathbf{1}$, the term w_θ is unique solution and it follows that $\Phi^\top D_\theta^{s,a}[\mathcal{T}_\theta(\Phi w_\theta) - \Phi w_\theta] = 0$. Recall from Lemmas 2 and 3 that $(\bar{\mu}_t, \langle w_t\rangle^\top)^\top$ is bounded a.s. According to Lemma 12, we have $\lim_t \bar{\mu}_t = J(\theta)$ and $\lim_t\langle w_t\rangle = w_\theta$ over the set $\mathbb{I}_{\{\sup_t\|z_t\|\le M\}}$ for any $M > 0$. Hence, $\lim_t \bar{\mu}_t = J(\theta)$ and $\lim_t\langle w_t\rangle = w_\theta$ a.s. ∎

Based on Lemmas 6 and 7, we obtain the following theorem.

Theorem 1: Under Assumptions 1, 2, and 4–6, for any given policy π_θ, the sequences $\{\mu_t^i\}$ and $\{w_t^i\}$ generated from Algorithm 1 satisfying $\lim_{t\to\infty}(1/N)\sum_{i\in\mathcal{N}}\mu_t^i = J(\theta)$ and $\lim_{t\to\infty} w_t^i = w_\theta$ a.s. for all $i \in \mathcal{N}$, where $J(\theta) = \sum_{s\in\mathcal{S}} d_\theta(s)\sum_{a\in\mathcal{A}}\pi_\theta(s,a)\bar{R}(s,a)$ and w_θ is the unique solution to $\Phi^\top D_\theta^{s,a}[\mathcal{T}_\theta(\Phi w_\theta) - \Phi w_\theta] = 0$. Suppose further that Assumption 7 holds, and the sequence $\{\theta_t^i\}$ for any $i \in \mathcal{N}$ obtained from Algorithm 1 converges a.s. to a point in the set of the asymptotically stable equilibria of

$$\dot{\theta}^i = \hat{\Gamma}^i\big(\mathbb{E}_{s_t\sim d_\theta, a_t\sim\pi_\theta}\big(A_{t,\theta}^i\psi_{t,\theta}^i\big)\big). \quad (24)$$

Proof: By Lemmas 6 and 7, we can obtain that $\lim_t(1/N)\sum_{i\in\mathcal{N}}\mu_t^i = J(\theta)$ and $\lim_t w_t^i = w_\theta$ a.s. for all $i \in \mathcal{N}$. As for the convergence of $\{\theta_t^i\}$, and the proof is along similar lines as that in [36, Th. 4.7] based on the Kushner–Clark lemma in Appendix B. ∎

B. Convergence of Algorithm 2

In order to obtain the convergence of w_t^i and θ_t^i in Algorithm 2, some preliminary definitions and lemmas are introduced.

In Algorithm 2, we define $\tilde{w}_t = ((\tilde{w}_t^1)^\top, \ldots, (\tilde{w}_t^N)^\top)^\top$ and $\epsilon_{t+1} = ((\epsilon_{t+1}^1)^\top, \ldots, (\epsilon_{t+1}^N)^\top)^\top$ with $\epsilon_{t+1}^i = \beta_{w,t}\tilde{\delta}_t^i\phi_t$ for all $i \in \mathcal{N}$. Let $\tilde{z}_t^i = (\mu_t^i, (\tilde{w}_t^i)^\top)^\top$ and $\tilde{z}_t = ((\tilde{z}_t^1)^\top, \ldots, (\tilde{z}_t^N)^\top)^\top$.

Lemma 8 [42]: Suppose that the graph sequence $\{\mathcal{G}_t\}$ is uniformly strongly connected. For each integer $l \geq 0$, there is a stochastic vector sequence $\{\varphi(l)\}$ such that for all $i, j \in \mathcal{N}$ and $t \geq l$

$$|C_{ij}(t:l) - \varphi_i(t)| \leq M_2\lambda^{t-l}$$

for some $M_2 > 0$ and $\lambda \in (0,1)$, where $C(t:l) = C(l)C(l+1), \ldots, C(t)$, $C_{ij}(t:l)$ being the ith row and the jth column element in $C(t:l)$, and $\varphi_i(t)$ is the ith element in $\varphi(t)$.

Lemma 9: Under Assumptions 3–6, for any $i \in \mathcal{N}$, we have $\lim_t w_{t+1}^i - (1/N)(\mathbf{1}^\top \otimes I)\tilde{w}_t = 0$ a.s.

Proof: The compact form of (10c) is

$$\begin{aligned}\tilde{w}_{t+1} &= (C(t)\otimes I)\tilde{w}_t + \epsilon_{t+1}\\ &= (C(t:0)\otimes I)\tilde{w}_0 + \sum_{l=1}^{t}[C(t:l)\otimes I]\epsilon_l + \epsilon_{t+1} \quad (25)\end{aligned}$$

which implies that $(C(t+1)\otimes I)\tilde{w}_{t+1} = (C(t+1:0)\otimes I)\tilde{w}_0 + \sum_{l=1}^{t+1}(C(t+1:l)\otimes I)\epsilon_l$. Since $C(t)$ is column stochastic for all $t \geq 0$, we have that $(\mathbf{1}^\top \otimes I)\tilde{w}_{t+1} = (\mathbf{1}^\top \otimes I)\tilde{w}_0 + \sum_{l=1}^{t+1}(\mathbf{1}^\top \otimes I)\epsilon_l$. Then, it can be obtained that $(C(t+1)\otimes I)\tilde{w}_{t+1} - ((\varphi(t+1)\cdot\mathbf{1}^\top)\otimes I)\tilde{w}_{t+1} = ((C(t+1:0) - \varphi(t+1)\cdot\mathbf{1}^\top)\otimes I)\tilde{w}_0 + \sum_{l=1}^{t+1}((C(t+1:l) - \varphi(t+1)\cdot\mathbf{1}^\top)\otimes I)\epsilon_l$. Define $D(t,l) = C(t:l) - \varphi(t)\cdot\mathbf{1}^\top$, and we have $(C(t+1)\otimes I)\tilde{w}_{t+1} = ((\varphi(t+1)\cdot\mathbf{1}^\top)\otimes I)\tilde{w}_{t+1} + (D(t+1:0)\otimes I)\tilde{w}_0 + \sum_{l=1}^{t+1}(D(t+1:l)\otimes I)\epsilon_l$. For convenience, denote $D_i(t:0)$ as the ith row of $D(t:0)$. Consider that $o_{t+1} = N\cdot\varphi(t) + D(t:0)\cdot\mathbf{1}$, and we have

$$\begin{aligned}&w_{t+1}^i - \frac{1}{N}\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t\\ &\quad= \frac{\left(\varphi_i(t)\cdot\mathbf{1}^\top\otimes I\right)\tilde{w}_t + (D_i(t:0)\otimes I)\tilde{w}_0}{N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1}}\\ &\qquad+\frac{\sum_{l=1}^{t}(D_i(t:l)\otimes I)\epsilon_l}{N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1}} - \frac{\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t}{N}\\ &\quad= \frac{(D_i(t:0)\otimes I)\tilde{w}_0 + \sum_{l=1}^{t}(D_i(t:l)\otimes I)\epsilon_l}{N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1}}\\ &\qquad-\frac{(D_i(t:0)\cdot\mathbf{1})\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t}{N\cdot(N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1})}. \quad (26)\end{aligned}$$

Based on [42, Proof of Lemma 1], there exists $\kappa > 0$ such that $N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1} \geq \kappa$. Therefore, we have

$$\begin{aligned}&\left\|w_{t+1}^i - \frac{1}{N}\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t\right\|\\ &\quad\leq \frac{\|(D_i(t:0)\otimes I)\tilde{w}_0\| + \left\|\sum_{l=1}^{t}(D_i(t:l)\otimes I)\epsilon_l\right\|}{N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1}}\\ &\qquad+\frac{\|(D_i(t:0)\cdot\mathbf{1})\cdot(\mathbf{1}^\top\otimes I)\tilde{w}_t\|}{N\cdot(N\cdot\varphi_i(t) + D_i(t:0)\cdot\mathbf{1})}\\ &\quad\leq \frac{\sqrt{N}}{\kappa}\left(M_2\lambda^t\|\tilde{w}_0\| + \sum_{l=1}^{t}M_2\lambda^{t-l}\|\epsilon_l\|\right)\\ &\qquad+\frac{M_2\lambda^t}{\kappa}\|\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t\|. \quad (27)\end{aligned}$$

According to [36, Proof of Lemma 5.1], it can show that $\sup_t|\tilde{w}_t^i| < \infty$ a.s. can also be obtained when the weight matrix is column stochastic. Consider $\|w_{t+1}^i - (1/N)(\mathbf{1}^\top\otimes I)\tilde{w}_t\|$ on the set $\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}}$, and then, we have

$$\begin{aligned}&\left\|w_{t+1}^i - \frac{1}{N}\left(\mathbf{1}^\top\otimes I\right)\tilde{w}_t\right\|\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}}\\ &\quad\leq \frac{\sqrt{N}}{\kappa}\left(M_2\lambda^t\|\tilde{w}_0\| + \sum_{l=1}^{t}M_2\lambda^{t-l}\|\epsilon_l\|\right)\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}}\\ &\qquad+\frac{M_2\lambda^t}{\kappa}\|(\mathbf{1}^\top\otimes I)\tilde{w}_t\|\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}}. \quad (28)\end{aligned}$$

By Assumptions 4 and 5, r_{t+1}^i and ϕ_t are uniformly bounded for any $i \in \mathcal{N}$ and $t \geq 0$, and there exists K_5 and K_6 such that $\|w_{t+1}^i - (1/N)(\mathbf{1}^\top\otimes I)\tilde{w}_t\|\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}} \leq K_5\lambda^t + \sum_{l=1}^{t}K_6\beta_{w,t}\lambda^{t-l}$. As a result, we have $\lim_t w_{t+1}^i - (1/N)(\mathbf{1}^T\otimes I)\tilde{w}_t = 0$ a.s. ∎

For notational convenience, we define the consensus vector $\hat{w}_t \triangleq (1/N)(\mathbf{1}^\top\otimes I)\tilde{w}_t$. In the following, we will show the convergence of $\hat{w}_t$.

Lemma 10: Under Assumptions 2, 3, and 4–6, we have $\lim_t\bar{\mu}_t = J(\theta)$ and $\lim_t\hat{w}_t = w_\theta$ a.s.

Proof: By (10), we can write the updates for $\bar{\mu}_t$ and $\hat{w}_t$ as follows:

$$\begin{cases}\bar{\mu}_{t+1} = \bar{\mu}_t + \beta_{w,t}\mathbb{E}[\bar{r}_{t+1} - \bar{\mu}_t|\mathcal{F}_t] + \beta_{w,t}\xi_{t+1,1} & (29a)\\ \hat{w}_{t+1} = \hat{w}_t + \beta_{w,t}\mathbb{E}[\hat{\delta}_t\phi_t|\mathcal{F}_t] + \beta_{w,t}\xi_{t+1,4} & (29b)\end{cases}$$

where $\hat{\delta}_t = (1/N)\sum_{i\in\mathcal{N}}((r_{t+1}^i - \mu_t^i) + (\phi_{t+1} - \phi_t)^\top\tilde{w}_t^i)$ and $\xi_{t+1,4} = \hat{\delta}_t\phi_t - \mathbb{E}[\hat{\delta}_t\phi_t|\mathcal{F}_t]$.

Notice that (29a) is the same as (18a), so we will just analyze (29b) in the following. According to the definition of $\xi_{t+1,4}$, $\xi_{t+1,4}$ is a martingale difference sequence and we have

$$\mathbb{E}[\|\xi_{t+1,4}\|^2|\mathcal{F}_t] \leq 2\mathbb{E}[\|\hat{\delta}_t\phi_t\|^2|\mathcal{F}_t] + 2\|\mathbb{E}[\hat{\delta}_t\phi_t|\mathcal{F}_t]\|^2.$$

For any $M > 0$, since the boundedness of r_t^i and ϕ_t for any $i \in \mathcal{N}$ and $t \geq 0$, there exists $K_7 < \infty$ such that $\mathbb{E}[\|\xi_{t+1,4}\|^2|\mathcal{F}_t] \leq K_7(1 + \|\bar{\mu}_t\|^2 + \|\hat{w}_t\|^2)$ over the set $\mathbb{I}_{\{\sup_t\|\tilde{z}_t\|\leq M\}}$. Consider the following ODE captures the asymptotic behavior of (29a) and (29b):

$$\begin{aligned}\begin{pmatrix}\dot{\bar{\mu}}\\ \dot{\hat{w}}_t\end{pmatrix} &= \begin{pmatrix}-1 & 0\\ -\Phi^\top D_\theta^{s,a}\mathbf{1} & \Phi^\top D_\theta^{s,a}\left(P^\theta - I\right)\Phi\end{pmatrix}\begin{pmatrix}\bar{\mu}\\ \hat{w}_t\end{pmatrix}\\ &\quad+\begin{pmatrix}J(\theta)\\ \Phi^\top D_\theta^{s,a}\bar{R}\end{pmatrix}. \quad (30)\end{aligned}$$

By a similar analysis as in the proof of Lemma 7, we have

$$\begin{cases}\bar{\mu} = J(\theta) & (31)\\ \Phi^\top D_\theta^{s,a}(\bar{R} - \bar{\mu}\mathbf{1} + P^\theta\Phi\hat{w}_t - \Phi\hat{w}_t) = 0. & (32)\end{cases}$$

Meanwhile, $\lim_t\bar{\mu}_t = J(\theta)$ and $\lim_t\hat{w}_t = w_\theta$ a.s. ∎

Based on Lemmas 9 and 10, we obtain the following result.

Theorem 2: Under Assumptions 1, 3, and 4–6, for any given policy π_θ, the sequences $\{\mu_t^i\}$ and $\{w_t^i\}$ generated from Algorithm 2 satisfying $\lim_{t\to\infty}(1/N)\sum_{i\in\mathcal{N}}\mu_t^i = J(\theta)$ and $\lim_{t\to\infty}w_t^i = w_\theta$ a.s. for all $i \in \mathcal{N}$. Furthermore, suppose that Assumption 7 holds, and the sequence $\{\theta_t^i\}$ obtained from Algorithm 2 also converges to a point in the set of the asymptotically stable equilibria of (24).

Proof: By the above result with Lemmas 9 and 10, it can be obtained that $\lim_t (1/N)\sum_{i\in\mathcal{N}} \mu_t^i = J(\theta)$ and $\lim_t w_t^i = w_\theta$ a.s. for all $i \in \mathcal{N}$. As for the convergence of θ_t^i, the proof is along similar lines as that in [36, Th. 4.7] based on the Kushner–Clark lemma in Appendix B. ■

VI. Case Study

In this section, we evaluate the proposed distributed AC algorithms through numerical simulations on directed graphs with fixed and changing topologies.

A. Directed Graph With Fixed Topology

Consider the MARL problem, which can be represented as the multiagent MDP $(\mathcal{S}, \{\mathcal{A}^i\}_{i\in\mathcal{N}}, P, \{R^i\}_{i\in\mathcal{N}}, \mathcal{G}_t)$, where $\mathcal{N} = \{1, \ldots, N\}$, $\mathcal{S}$ has 20 states, and with binary-valued action space at each state, i.e., $\mathcal{A}^i = \{0, 1\}$ for all $i \in \mathcal{N}$. Consider that $|\mathcal{N}| = 20$, and the cardinality of the set of actions $\mathcal{A} = \prod_{i=1}^N \mathcal{A}^i$ at each state is 2^{20}. Assume that all agents are connected according to the communication network in Fig. 1 and the elements in the transition probability matrix P are uniformly sampled from the interval $[0, 1]$ and normalized to be stochastic. For each agent i and each state–action pair (s, a), the expected reward $R^i(s, a)$ is sampled uniformly from $[0, 4]$, which varies among agents. The instantaneous rewards for all $i \in \mathcal{N}$ are sampled from the uniform distribution $[R^i(s, a) - 0.5, R^i(s, a) + 0.5]$. We approximate the action value function by a quadratic function in a and also linear in the parameter $w \in \mathbb{R}^K$, i.e., $Q(s, a; w) = (a/5)^\top E(s)(a/5)w_1 + (a/5)^\top F(s)w_{2:K-1} + w_K$, where $w = (w_1, w_{2:K-1}^\top, w_K)^\top$ with dimensions $K = 10$ and the feature functions $E(s) \in \mathbb{R}^{N\times N}$ and $F(s) \in \mathbb{R}^{N\times(K-2)}$ are Gaussian radial basis functions (RBFs) with their means randomly selected from $[0, 1]$ and variances set as 0.1. The policy $\pi_{\theta^i}^i(s, a^i)$ is parameterized according to the Boltzmann policies, i.e.,

$$\pi_{\theta^i}^i(s, a^i) = \frac{\exp\left(q_{s,a^i}^\top \theta^i\right)}{\sum_{b^i\in\mathcal{A}^i} \exp\left(q_{s,b^i}^\top \theta^i\right)}$$

where $q_{s,a^i} \in \mathbb{R}^5$ is the feature vector defined as follows:

$$q_{s,a^i} = \begin{cases} \exp\left(\left(-\left(s\mathbf{1}/20 - G_i^\top\right)^2\right)/0.2\right), & \text{if } a^i = 0 \\ \exp\left(\left(-\left(s\mathbf{1}/20 - H_i^\top\right)^2\right)/0.2\right), & \text{otherwise} \end{cases}$$

where $G \in \mathbb{R}^{20\times 5}$ and $H \in \mathbb{R}^{20\times 5}$ are uniformly sampled from the interval $[0, 1]$ and G_i and H_i are the ith row of G and H, respectively. The gradient of the policy function has the form

$$\nabla_{\theta^i} \log \pi_{\theta^i}^i(s, a^i) = q_{s,a^i} - \sum_{b^i\in\mathcal{A}^i} \pi_{\theta^i}^i(s, b^i) q_{s,b^i}. \tag{33}$$

Assume that the weight matrix $C = [c_{ij}]_{N\times N}$ of the communication network in Fig. 1 is as follows:

$$c_{ij} = \begin{cases} 0.1, & \text{if } j \in \mathcal{N}_i^{\text{in}} \\ 0, & \text{otherwise} \end{cases}$$

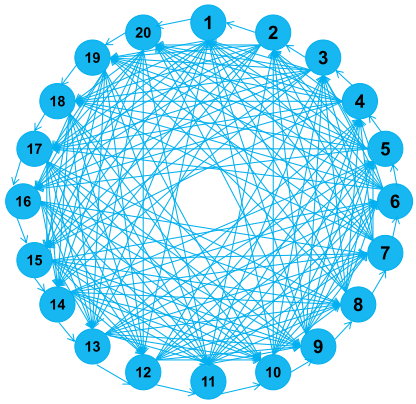

Fig. 1. Directed graph with fixed topology.

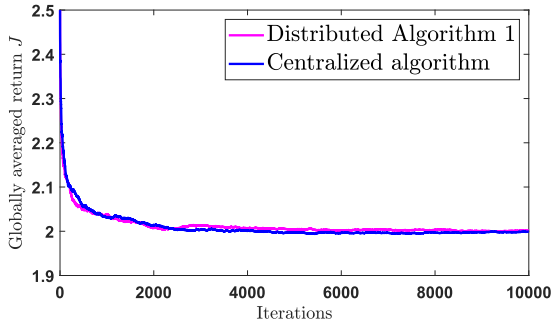

Fig. 2. Globally averaged return of distributed Algorithm 1 and centralized algorithm (7).

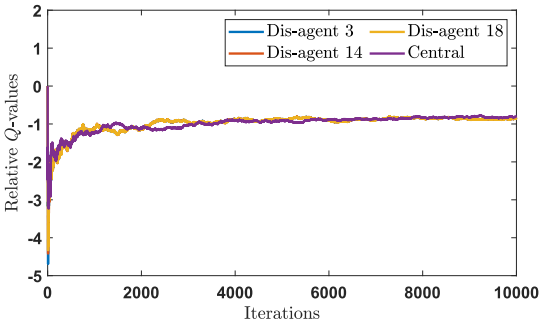

Fig. 3. Relative Q-values in distributed Algorithm 1 and centralized algorithm (7).

which is row stochastic since the in-degree of each agent in Fig. 1 is 10. In particular, the stepsizes are selected as $\beta_{w,t} = 1/(10 \cdot t^{0.65})$ and $\beta_{\theta,t} = 1/(10 \cdot t^{0.85})$.

The performance of Algorithm 1 is compared with the centralized algorithm (7) where the instantaneous rewards r_t^i for all $i \in \mathcal{N}$ are available to a centralized controller. We compare the proposed Algorithm 1 and the centralized algorithm with respect to the globally averaged return, the relative Q-value, and the local policy of each agent at randomly selected states. As shown in Fig. 2, the proposed Algorithm 1 converges to the same globally long-term averaged return of the centralized algorithm. Fig. 3 shows that for agent 3, 14, and 18, the state–action value at state–action pair (s, a) with $s = 4$ and $a = (0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 1\ 1\ 1\ 0\ 1\ 1\ 1\ 0)$ in

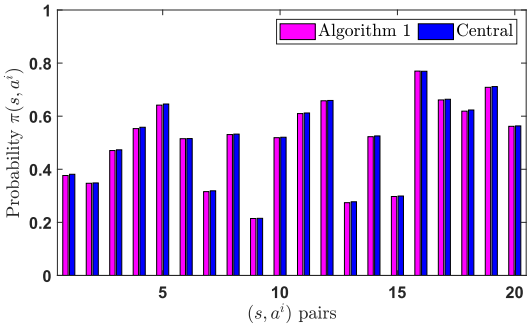

Fig. 4. Probability distribution $\pi^i(s, a^i)$ at a randomly selected state $s = 1$ and $a^i = 0$ for distributed Algorithm 1 and centralized algorithm (7).

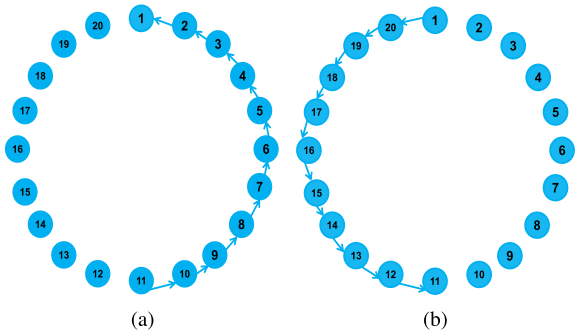

Fig. 5. Directed graphs with changing topologies. Notice that the two resulting graphs are not strongly connected. (a) Graph 1. (b) Graph 2.

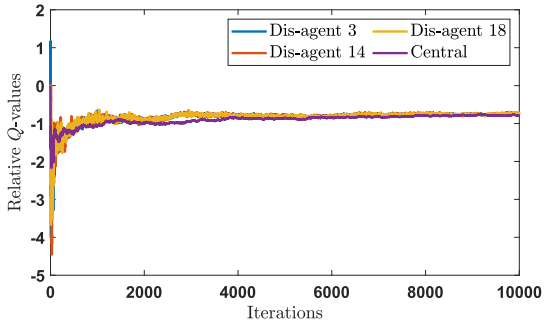

Fig. 6. Relative Q-values in distributed Algorithm 2 and centralized algorithm (7).

Algorithm 1 reaches consensus and converges the corresponding state–action value of the centralized algorithm. In addition, Fig. 4 shows that both algorithms converge to similar policies at state $s = 1$, meaning that the joint policy obtained by agents using local information is close to the policy obtained by the centralized controller with full network information.

B. Directed Graphs With Changing Topologies

We consider a similar MARL as the previous case study, with the difference that the graphs have time-varying topologies, as it switches between the two directed graphs that are shown in Fig. 5, meanwhile satisfying Assumption 3. The weight matrices of the two directed graphs are designed as in (9). We use the same approximation action value function and approximation policy function as the previous case study

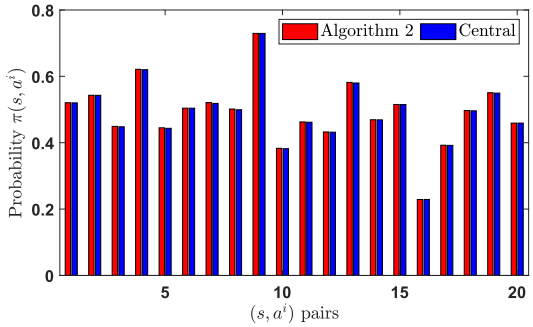

Fig. 7. Probability distribution $\pi^i(s, a^i)$ at a randomly selected state $s = 12$ and $a^i = 0$ for distributed Algorithm 2 and centralized algorithm (7).

and compare the proposed Algorithm 2 with the centralized algorithm (7). Fig. 6 shows that the relative Q-values of agent 3, 14, and 18 at state–action pair (s, a) with $s = 4$ and $a = (0\ 1\ 0\ 1\ 0\ 0\ 0\ 1\ 1\ 0\ 1\ 0\ 1\ 0\ 0\ 0\ 0\ 0\ 1\ 1)$ calculated by Algorithm 2 achieve consensus and converge to the relative Q-values calculated by the centralized algorithm. Fig. 7 shows that both algorithms converge to similar policies at state $s = 12$.

VII. Conclusion

In this article, MARL over directed graphs has been investigated. Two new distributed AC algorithms have been proposed to make each agent collaborate to the maximization of the globally averaged return. More specifically, a first distributed AC algorithm using a row stochastic weight matrix has been proposed for MARL over directed graphs with fixed topology, while the other distributed AC algorithm has been proposed for MARL over directed graphs with changing topologies using a push-sum idea. The convergence with linear function approximation has been proved for both algorithms. The proposed algorithms extend the applicability of distributed MARL. Future work will further extend the proposed algorithms to MARL settings with continuous state and action spaces.

Appendix A
Stochastic Approximation

Consider the n-dimensional stochastic approximation iteration in $\mathbb{R}^n$ as follows:

$$x_{t+1} = x_t + \beta_t\big[h(x_t, Y_t) + M_{t+1} + \zeta_{t+1}\big] \tag{A.1}$$

where $\beta_t > 0$ and $\{Y_t\}_{t\geq 0}$ is a Markov chain on a finite set S. Consider the following assumptions.

1) $h(x_t, Y_t)$:$\mathbb{R}^n \times S \to \mathbb{R}^n$ is Lipschitz continuous in x_t.
2) $\{Y_t\}_{t\geq 0}$ is an irreducible Markov chain with stationary distribution d.
3) β_t is the stepsize, which satisfies $\sum_t \beta_t = \infty$ and $\sum_t \beta_t^2 < \infty$.
4) $\{M_t\}$ is a martingale difference sequence, which satisfies that $\mathbb{E}(\| M_{t+1} \|^2 | x_m, M_m, Y_m, m \leq t) \leq K(1+ \| x_t \|^2)$ for some $K \geq 0$ and $t \geq 0$.
5) $\{\zeta_t\}$ is a bounded random sequence with $\lim_{t\to\infty} \zeta_t = 0$ a.s.

Let $\bar{h}(x) = \sum_s d(s)h(x,s)$ and $\bar{h}_c(x) = \bar{h}(cx)$. Under assumptions 1)–5), we have the following lemmas [43].

Lemma 11: If the $\lim_{c\to\infty} \bar{h}(cx)/c = h_\infty(x)$ exists uniformly on compact sets and the ODE $\dot{y} = h_\infty(y)$ has the origin as the unique globally asymptotically stable equilibrium, then $\sup_t \|x_t\| < \infty$ a.s.

Lemma 12: If $\dot{x} = \bar{h}(x)$ has a unique globally asymptotically stable equilibrium x^* and $\sup_t \|x_t\| < \infty$, then $\lim_t x_t \to x^*$ a.s.

APPENDIX B
KUSHNER–CLARK LEMMA

Let $\Gamma : \mathbb{R}^N \to \mathbb{R}^N$ be an operator that projects a vector onto a compact set $\mathcal{X} \subseteq \mathbb{R}^N$. Define

$$\hat{\Gamma}(h(x)) = \lim_{0<\eta\to 0} \frac{\Gamma(x+\eta h(x)) - x}{\eta} \tag{B.1}$$

for $x \in \mathcal{X}$ and with $h{:}\mathcal{X} \to \mathbb{R}^N$ continuous. Consider the following iteration:

$$x_{t+1} = \Gamma\big(x_t + \beta_t\big(h(x_t) + \zeta_{t,1} + \zeta_{t,2}\big)\big). \tag{B.2}$$

The ODE associated with (B.2) is

$$\dot{x} = \hat{\Gamma}(h(x)). \tag{B.3}$$

Consider the following assumptions.

1) $h(\cdot)$ is a continuous function.
2) β_t satisfies $\sum_t \beta_t = \infty$ and $\sum_t \beta_t^2 < \infty$.
3) The sequence $\{\zeta_{t,1}\}$ satisfies $\lim_t \mathbb{P}(\sup_{n\geq t} \| \sum_{\tau=t}^n \beta_t \zeta_{t,1} \| \geq \epsilon) = 0$ for all $\epsilon > 0$.
4) The sequence $\{\zeta_{t,2}\}$ is a bounded random sequence with $\zeta_{t,2} \to 0$ a.s.

Under assumptions 1)–4), we have the following lemma [43].

Lemma 13: If (B.3) has a compact set $\mathcal{K}^*$ as its asymptotically stable equilibria, then (B.2) converges to $\mathcal{K}^*$ a.s.

REFERENCES

[1] H. He, Z. Ni, and D. Zhao, "Learning and optimization in hierarchical adaptive critic design," in *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control.*, Piscataway, NJ, USA: IEEE Press, 2013, pp. 78–95.

[2] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. Cambridge, MA, USA: MIT Press, 1998.

[3] H. He, Z. Ni, and J. Fu, "A three-network architecture for on-line learning and optimization based on adaptive dynamic programming," *Neurocomputing*, vol. 78, no. 1, pp. 3–13, Feb. 2012.

[4] B. Luo, D. Liu, T. Huang, and D. Wang, "Model-free optimal tracking control via critic-only Q-learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 10, pp. 2134–2144, Oct. 2016.

[5] B. Luo, H.-N. Wu, and H.-X. Li, "Adaptive optimal control of highly dissipative nonlinear spatially distributed processes with neuro-dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 4, pp. 684–696, Apr. 2015.

[6] Q. Wei, D. Liu, and X. Yang, "Infinite horizon self-learning optimal control of nonaffine discrete-time nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 4, pp. 866–879, Apr. 2015.

[7] Z. Ni, H. He, and J. Wen, "Adaptive learning in tracking control based on the dual critic network design," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 6, pp. 913–928, Jun. 2013.

[8] V. Mnih *et al.*, "Playing Atari with deep reinforcement learning," 2013, *arXiv:1312.5602*.

[9] V. Mnih *et al.*, "Human-level control through deep reinforcement learning," *Nature*, vol. 518, pp. 529–533, Feb. 2015.

[10] K. Zhang, W. Shi, H. Zhu, E. DallAnese, and T. Başar, "Dynamic power distribution system management with a locally connected communication network," *IEEE J. Sel. Topics Signal Process.*, vol. 12, no. 4, pp. 673–687, Aug. 2018.

[11] F. Li, J. Qin, and W. Zheng, "Distributed Q-learning-based online optimization algorithm for unit commitment and dispatch in smart grid," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 4146–4156, Sep. 2020.

[12] P. Dai, W. Yu, G. Wen, and S. Baldi, "Distributed reinforcement learning algorithm for dynamic economic dispatch with unknown generation cost functions," *IEEE Trans. Ind. Informat.*, vol. 16, no. 4, pp. 2258–2267, Apr. 2020.

[13] P. Dai, W. Yu, and D. Chen, "Distributed Q-learning algorithm for dynamic resource allocation with unknown objective functions and application to microgrid," *IEEE Trans. Cybern.*, early access, Jun. 24, 2021, doi: 10.1109/TCYB.2021.3082639.

[14] H. Yuan and Y. Xia, "Resilient strategy design for cyber-physical system under DoS attack over a multi-channel framework," *Inf. Sci.*, vols. 454–455, pp. 312–327, Jul. 2018.

[15] K. Ding, Y. Li, D. E. Quevedo, S. Dey, and L. Shi, "A multi-channel transmission schedule for remote state estimation under DoS attacks," *Automatica*, vol. 78, pp. 194–201, Apr. 2017.

[16] P. Dai, W. Yu, H. Wang, G. Wen, and Y. Lv, "Distributed reinforcement learning for cyber-physical system with multiple remote state estimation under DoS attacker," *IEEE Trans. Netw. Sci. Eng.*, vol. 7, no. 4, pp. 3212–3222, Oct. 2020.

[17] K. Zhang, Z. Yang, and T. Başar, "Multi-agent reinforcement learning: A selective overview of theories and algorithms," in *Studies Systems Decision Control Handbook RL Control*. New York, NY, USA: Springer, 2020.

[18] M. L. Littman, "Markov games as a framework for multi-agent reinforcement learning," in *Proc. 11th Int. Conf. Mach. Learn. (ICML)*, New Brunswick, NJ, USA, Jul. 1994, pp. 157–163.

[19] M. Lauer and M. Riedmiller, "An algorithm for distributed reinforcement learning in cooperative multi-agent systems," in *Proc. 17th Int. Conf. Mach. Learn. (ICML)*, vol. 2, Stanford, CA, USA: Stanford Univ., Jun. 200, pp. 535–542.

[20] J. Hu and M. P. Wellman, "Nash Q-learning for general-sum stochastic games," *J. Mach. Learn. Res.*, vol. 4, pp. 1039–1069, Nov. 2003.

[21] R. Olfati-Saber and R. M. Murray, "Consensus problems in networks of agents with switching topology and time-delays," *IEEE Trans. Autom. Control*, vol. 49, no. 9, pp. 1520–1533, Sep. 2004.

[22] Z.-W. Liu, G. Wen, X. Yu, Z.-H. Guan, and T. Huang, "Delayed impulsive control for consensus of multiagent systems with switching communication graphs," *IEEE Trans. Cybern.*, vol. 50, no. 7, pp. 3045–3055, Jul. 2020.

[23] H.-X. Hu, G. Wen, W. Yu, T. Huang, and J. Cao, "Distributed stabilization of multiple heterogeneous agents in the Strong–Weak competition network: A switched system approach," *IEEE Trans. Cybern.*, vol. 51, no. 11, pp. 5328–5341, Nov. 2021, doi: 10.1109/TCYB.2020.2995154.

[24] H.-X. Hu, Q. Zhou, G. Wen, W. Yu, and W. Kong, "Robust distributed stabilization of heterogeneous agents over Cooperation–Competition networks," *IEEE Trans. Circuits Syst. II: Exp. Briefs*, vol. 67, no. 8, pp. 1419–1423, Aug. 2020.

[25] H.-X. Hu, G. Wen, X. Yu, Z.-G. Wu, and T. Huang, "Distributed stabilization of heterogeneous MASs in uncertain strong-weak competition networks," *IEEE Trans. Syst., Man, Cybern. Syst.*, early access, Nov. 24, 2020, doi: 10.1109/TSMC.2020.3034765.

[26] M. Chen, H. Yan, H. Zhang, M. Chi, and Z. Li, "Dynamic event-triggered asynchronous control for nonlinear multiagent systems based on T–S fuzzy models," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 9, pp. 2580–2592, Sep. 2021, doi: 10.1109/TFUZZ.2020.3004009.

[27] R. Yang, H. Zhang, G. Feng, H. Yan, and Z. Wang, "Robust cooperative output regulation of multi-agent systems via adaptive event-triggered control," *Automatica*, vol. 102, no. 6, pp. 129–136, Apr. 2019.

[28] A. Nedic and A. Ozdaglar, "Distributed subgradient methods for multi-agent optimization," *IEEE Trans. Autom. Control*, vol. 54, no. 1, pp. 48–61, Jan. 2009.

[29] A. Nedic, A. Ozdaglar, and P. A. Parrilo, "Constrained consensus and optimization in multi-agent networks," *IEEE Trans. Autom. Control*, vol. 55, no. 4, pp. 922–938, Apr. 2010.

[30] S. Kar, J. M. Moura, and H. V. Poor, "QD-learning: A collaborative distributed strategy for multi-agent reinforcement learning through consensus + innovations," *IEEE Trans. Signal. Process.*, vol. 61, no. 7, pp. 1848–1862, Apr. 2013.
[31] P. Sunehag *et al.*, "Value-decomposition networks for cooperative multi-agent learning based on team reward," in *Proc. 17th Int. Conf. Auton. Agents MultiAgent Syst.*, 2018, pp. 2085–2087.
[32] T. Rashid, M. Samvelyan, C. S. De Witt, G. Farquhar, J. Foerster, and S. Whiteson, "QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning," in *Proc. 35th Int. Conf. Mach. Learn.*, 2018, pp. 6846–6859.
[33] K. Son, D. Kim, W. J. Kang, D. Hostallero, and Y. Yi, "Qtran: Learning to factorize with transformation for cooperative multi-agent reinforcement learning," in *Proc. 36th Int. Conf. Mach. Learn.*, 2019, pp. 5887–5896.
[34] A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson, "MAVEN: Multi-agent variational exploration," in *Proc. Adv. Neural Inf. Process. Syst.*, 2019, pp. 7613–7624.
[35] J. Wang, Z. Ren, T. Liu, Y. Yu, and C. Zhang, "QPLEX: Duplex dueling multi-agent Q-learning," in *Proc. Int. Conf. Learn. Represent.*, Sep. 2021, pp. 1–11.
[36] K. Zhang, Z. Yang, H. Liu, T. Zhang, and T. Başar, "Fully decentralized multi-agent reinforcement learning with networked agents," in *Proc. Int. Conf. Mach. Learn.*, 2018, pp. 5867–5876.
[37] K. Zhang, Z. Yang, and T. Başar, "Networked multi-agent reinforcement learning in continuous spaces," in *Proc. IEEE Conf. Decis. Control (CDC)*, Dec. 2018, pp. 2771–2776.
[38] R. S. Sutton, A. R. Mahmood, and M. White, "An emphatic approach to the problem of off-policy temporal-difference learning," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 2603–2631, 2015.
[39] W. Suttle, Z. Yang, K. Zhang, Z. Wang, T. Başar, and J. Liu, "A multi-agent off-policy actor-critic algorithm for distributed reinforcement learning," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 1549–1554, 2020.
[40] Y. Lin *et al.*, "A communication-efficient multi-agent actor-critic algorithm for distributed reinforcement learning," in *Proc. 58th IEEE Conf. Decis. Control*, Dec. 2019, pp. 5562–5567.
[41] V. S. Mai and E. H. Abed, "Distributed optimization over directed graphs with row stochasticity and constraint regularity," *Automatica*, vol. 102, pp. 94–104, Apr. 2019.
[42] A. Nedić and A. Olshevsky, "Distributed optimization over time-varying directed graphs," *IEEE Trans. Autom. Control*, vol. 60, no. 3, pp. 601–615, Mar. 2015.
[43] V. S. Borkar, *Stochastic Approximation: A Dynamical Systems Viewpoint*. Cambridge, MA, USA: MIT Press, 2008.

Pengcheng Dai (Student Member, IEEE) received the B.S. degree in statistics from Yancheng Normal University, Yancheng, China, in 2016, and the M.S. degree in applied mathematics from Southeast University, Nanjing, China, in 2019, where he is currently pursuing the Ph.D. degree in applied mathematics.

His current research interests include distributed optimization and reinforcement learning.

Wenwu Yu (Senior Member, IEEE) received the B.Sc. degree in information and computing science and the M.Sc. degree in applied mathematics from the Department of Mathematics, Southeast University, Nanjing, China, in 2004 and 2007, respectively, and the Ph.D. degree from the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, in 2010.

He is currently the Founding Director of the Laboratory of Cooperative Control of Complex Systems, the Deputy Associate Director of the Jiangsu Provincial Key Laboratory of Networked Collective Intelligence, an Associate Dean of the School of Mathematics, and a Full Professor with the Endowed Chair Honor, Southeast University. He held several visiting positions in Australia, China, Germany, Italy, The Netherlands, and USA. He has published about 100 IEEE TRANSACTIONS journal articles with more than 18 000 citations. His research interests include multiagent systems, complex networks and systems, disturbance control, distributed optimization, machine learning, game theory, cyberspace security, smart grids, intelligent transportation systems, and big data analysis.

Dr. Yu was a recipient of the Second Prize of State Natural Science Award of China in 2016. He serves as an Editorial Board Member for several flag journals, including IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS—II: EXPRESS BRIEFS, IEEE TRANSACTIONS ON INDUSTRIAL INFORMATICS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS: SYSTEMS, *Science China Information Sciences*, and *Science China Technological Sciences*. He was listed by Clarivate Analytics/Thomson Reuters Highly Cited Researchers in Engineering from 2014 to 2020.

He Wang received the B.Sc. and Ph.D. degrees from Southeast University, Nanjing, China, in 2013 and 2019, respectively.

In 2016, he was a Visiting Scholar with the Department of Electronic Engineering, City University of Hong Kong, Hong Kong, for three months. From 2017 to 2018, he was a joint Ph.D. student with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA, USA, supported by the China Scholarship Council. In 2019, he was a Visiting Research Student with the School of Engineering, RMIT University, Melbourne, VIC, Australia, for three months. From 2019 to 2020, he was a Visiting Fellow with the College of Engineering and Computer Science, Australian National University, Canberra, ACT, Australia. He is currently an Associate Researcher with Southeast University. His current research interests include multiagent systems, cooperative antidisturbance control, finite-time control, distributed average tracking control, and unmanned aerial vehicle (UAV) formation control.

Simone Baldi (Senior Member, IEEE) received the B.Sc. degree in electrical engineering and the M.Sc. and Ph.D. degrees in control systems engineering from the University of Florence, Florence, Italy, in 2005, 2007, and 2011, respectively.

He was an Assistant Professor with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands. He is currently a Professor with the School of Mathematics, Southeast University, Nanjing, China, and a guest position at the Delft Center for Systems and Control, Delft University of Technology. His research interests include adaptive and learning systems with applications in networked control and intelligent vehicles.

Dr. Baldi is a Subject Editor of *International Journal of Adaptive Control and Signal Processing* and an Associate Editor of IEEE CONTROL SYSTEMS LETTERS.