Fully Decentralized Multiagent Communication via Causal Inference

Han Wang, Yang Yu, and Yuan Jiang

***Abstract*— Many real-world tasks can be cast into multiagent (MA) reinforcement learning problems, and most algorithms in this field obey to the centralized learning and decentralized execution framework. However, enforcing centralized learning is impractical in many scenarios. Because it requires integrating the information from agents, while agents may not hope to share local information due to the issue of privacy. Thus, this article proposes a novel approach to achieve fully decentralized learning based on communication among multiple agents via reinforcement learning. Benefiting from causality analysis, an agent will choose the counterfactual that has the most significant influence on communication information of others. We find that this method can be applied in classic or complex MA scenarios and in federated learning domains, which are now attracting much attention.**

***Index Terms*— Federated learning (FL), multiagent (MA) communication, multiagent reinforcement learning (MARL).**

I. Introduction

THE essence of reinforcement learning (RL) is to enforce agents' learning ability on how to make sequential decisions through interactions with the environment [1]. RL has been widely applied in various fields including robot controlling [2], video games, and game theory [3] after being integrated with deep learning technologies.

Although the development of RL is in full swing, and there exist well-known algorithms concluded in [4] performing satisfactorily in simulation tests and in computer games such as game of Go [5] and StarCraft II [6]. However, the real-world systems are much complex [7], [44] and most tasks require multiple agents to be coordinated at the same time [8]. Those agents will be scattered in each subspace [9], [45] to perform tasks, respectively. Actually, the overall decision-making requires cooperation between them on the condition that the local decision from each agent will influence others. In such multiagent (MA) systems, we hope that agents can cooperate to accomplish tasks in incompletely observable environments without stability assurance.

Recently, researchers have proposed many creative methods for MA reinforcement learning (MARL) problems, which can be roughly divided into the following categories. *On learning framework design* [10]–[13]: The research findings on this topic mainly intend to explore learning frameworks by integrating the existing single-agent RL background into the setting of MA systems. *On joint-action learning* [14]–[16]: These methods usually treat the MA system from the single system perspective, which means the actions of multiple agents are considered as the subactions of the system. However, when faced with high dimensional state or action space, such methods may cause learning inefficiency under the curse of dimension. *On enforcing communication between agents* [17]–[21]: Agents are allowed to send and receive brief communication messages to analyze the situation of other agents in the perceived environment to coordinate their local policies.

As discussed on methods mentioned above, directly learning the globally joint policy in a centralized way is infeasible from a practical perspective, shown in Fig. 1(a). While distributed learning in Fig. 1(b) is a independent, scalable approach, but may yield poor global performance. To achieve both scalability and optimality, recent research works are mainly based on the centralized training and decentralized execution (CTDE) framework [10], which permits local information exchange and integration such as q-values, actions a, or even observation-action pairs (o, a) during training, as depicted in Fig. 1(c). Generally, centralized training ensures agents' fundamental understanding of the global world [22].

However, the condition of centralized learning is also hard to meet in practical systems, especially in federated learning (FL) tasks [23]: multiple agents participating in the task are invisible to others, and they are unwilling to share information due to the issue of privacy [24]. CTDE methods are mainly based on assumptions including centralized learning or sharing confidential information (such as policy parameters), which makes those methods impractical to work. We hope that agents can break the dependency on centralized knowledge but to communicate with each other and perceive other intelligence in an incompletely known environment. Thus, using communication messages as complement observations for missing information in environments is more natural for real systems' generalization practically. Unfortunately, we find that many existing communication-based MARL methods are contrary to the privacy requirements, which is quite demanding especially in FL cases.

Main Contributions: With this issue, we make the following primary contributions in this article.

1) This article suggests that agents can automatically adjust the communication messages while learning the policy to achieve fully decentralized learning in MA

Manuscript received 16 June 2020; revised 30 January 2021 and 14 January 2022; accepted 25 March 2022. Date of publication 22 April 2022; date of current version 1 December 2023. This work was supported in part by the National Key Research and Development Program of China under Grant 2020AAA0107200 and in part by NSFC under Grant 61876077. *(Corresponding author: Yang Yu.)*

The authors are with the National Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China (e-mail: yuy@lamda.nju.edu.cn).

Digital Object Identifier 10.1109/TNNLS.2022.3165114

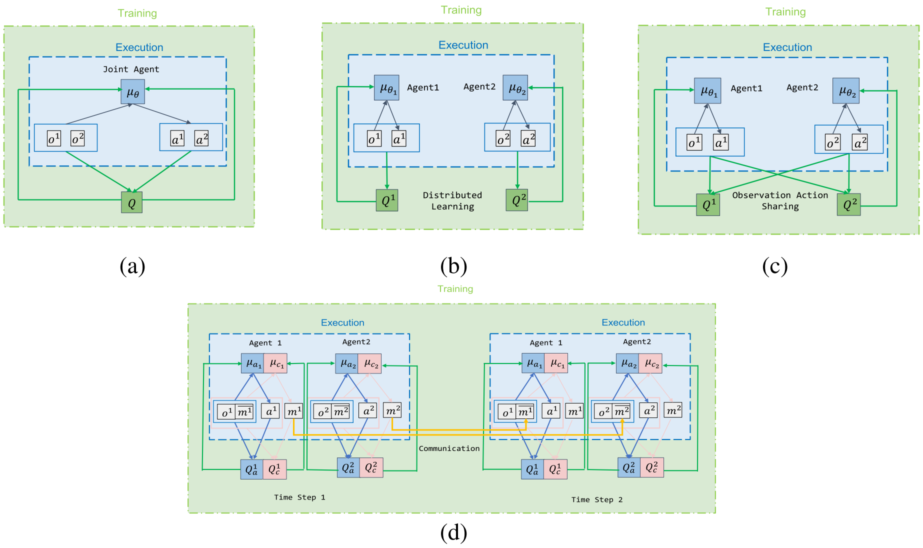


Fig. 1. Structures of several learning frames. (a) Centralized learning and centralized execution. (b) Distributed learning and distributed execution. (c) CTDE. (d) Fully decentralized learning with communication.

systems based on RL (FD-MARL). The result is a fully decentralized algorithm over a MA system, in which agents continuously communicate with the environment, receive the communication messages from others, and give meaningful feedback without centralized learning, as shown in Fig. 1(d).

2) This article provides structural causal model (SCM) for Markov Games, which frames the MA communication problem as a MA causal model. Our analysis on the causal relationship of communication messages reveals that FD-MARL matches the requirements of decentralized learning and execution with limited information sharing.

3) We evaluate FD-MARL on a diverse suite of MA domains, including the cooperative environments and federated reinforcement learning (FRL) tasks. The experimental results demonstrate that, in contrast with previous state-of-the-art approaches based on CTDE, FD-MARL consistently achieves superior or comparable performances on each benchmark.

II. Related Work

There are many branches of research on multi-agent systems (MAS). One of the well-known fields is distributed constraint optimization problems (DCOPs) [25]. Generally, DCOPs is a boarder concept. If we focus on team coordination in presence of uncertainty about agents' actions and observations, the agent architectures in DCOPs can be built on decentralized partially observable Markov decision processes (Dec-POMDPs) [26].

Principled on Dec-POMDPs, research on MARL has a long history. At first, researchers mainly focused on finding a learning framework that could transfer RL technology to MA Settings. Tan [27] showed that when we ignore the interaction between agents in static environments, independent RL learning cannot solve the problem in MA systems. Tesauro [15] applied Q-Learning in RL to MARL, and illustrated that this learning framework could be used under the premise of getting appropriate rewards and incentives. Besides, Lowe *et al.* [10] migrated the Actor-Critic Learning framework from RL to MARL, and [12] demonstrated that the proposed method could be applied in collaboration, competition, or a hybrid environment. Moreover, many works on cooperation scenarios have appeared in MARL. Experimental work in [28] enables the successful combination of experience replay in deep Q-learning with MARL.

Method in [15] is based on joint action and communication, and can be applied in cooperative environments. In [20], the authors proposed CommNet to take advantage of a continuous communication network that can be updated in a gradient manner to accomplish the cooperative tasks of agents, which is also based on observation and reward sharing during training and execution. In short, these methods treat distributed agents as a global agent and simplify the MARL to RL. However, these methods often fail to work on large-scale scenarios.

Recently, communication-based MARL methods have attracted much attention among researchers. Therefore, since Foerster *et al.* [19] proposed the method based on Q-learning that can update communication messages by gradient, many methods based on low-dimensional continuous communication messages have successfully solved many cooperative tasks of MAs. Rashid *et al.* [29] proposed MARL methods based on the Q-function to combine the values of each agent based on local observations in a sophisticated nonlinear manner to estimate the combined action-value, and obtained excellent results on StarCraftII. Researchers also tried to explain the meaning of communication messages in more general environments, such as sequential social dilemmas (SSDs) [17] and referential game [21]. Foerster *et al.* [30] solved the problem of MARL based on Actor-Critic framework from a causal inference perspective, and [31] implemented such inference-based MARL method to solve SSDs. As we move toward complex environments, researchers notice that it's imperative to improve communication efficiency. Under the limited bandwidth constraint, agents sharing redundant messages which are unsustainable under bandwidth limitations. Thus, in Wang *et al.* [32], compress the communication messages by communication theory, which allows agents to deliver compact messages. In Singh *et al.* [18], proposed IC3 based on CommNet. IC3 allows agents have an option to communicate but can choose when to actually communicate to increase scalability, performance, and competitive edge.

While there has been some interest in merging ideas from causality with current single-agent methods, to the best of our knowledge no research explicitly aims at bridging graphical causal methods with MARL. Prior works have investigated on exploiting counterfactual knowledge for the MA cooperative setting. For example, [33] improves performance on MA tasks where credit assignment is challenging by using counterfactual information. Jaques *et al.* [31] proposed using causality for intrinsic motivation via social influence. Ding *et al.* [34] enabled agents to learn a prior for agent-agent communication via causal inference. Overall, recent works in the field of MA communication focus on learning what messages to collect and when to send and whom to address them, but mainly based on the CTDE framework.

Actually, a more realistic problem is that the agents participating in a certain task maintain strict privacy protection requirements, which is the setting of FL. In Zhuo *et al.* [35] developed an MARL framework for FL settings based on MA Q-learning, which can use the integration of local Q-values to generate a global Q-function to guide the actions of agents. Moreover, this learning involved task can be long-term, in which agents are supposed to make time-varying sequential decisions. This article proposes to model the MA environment into a causal graph, and the communication among agents can be represented by the directed edge in the graph. Further, the learning performance can be improved by the communication among agents based on the causal relationship derived from inference.

III. Background

In MARL shown in Fig. 2, interactions between multiple agents can be represented by stochastic games, defined by a tuple $\mathcal{G} = (N, S, A, T, R, \gamma)$. Specifically, $S = (S_1, \ldots, S_N)$ is the state space, $A = (A_1, \ldots, A_N)$ is the action space and $R = (R_1, \ldots, R_N)$ is the set of reward functions in an N-agent stochastic game. In this setting, agents are invisible to each other [36]. At each time step t, each agent k performs action a_t^k based on its observation o_t^k of the true state $s_t^k \in S$.

If all agents aim to maximize the cumulative team reward, which means they share the same reward from environment as a function of the states and agents' actions $r : S \times A \to R$. Such cooperative MARL can be modeled by decentralized MA Partially Observable Markov Processes (MA-POMDP) [37], which is defined as $\mathcal{M} = (S, T, A, R, \gamma)$. Technically, we adopt the distributed asynchronous advantage actor-critic (A3C) approach [30] to train each agent's policies but in a de CTDE manner. In such approaches, actors are policy approximators, while critics are estimating value functions. Further, actors' updates are following gradients that depend on critics. In single-agent RL, critic's loss and actor's update can be mathematically expressed as follows:

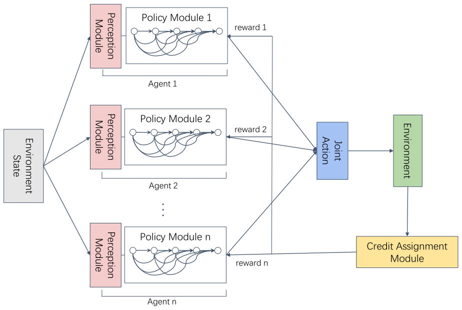


Fig. 2. MARL schema.

$$\text{loss} = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\left(r_t^n + V^{\pi}\left(s_{t+1}^n\right) - V^{\pi}\left(s_t^n\right)\right)$$

$$\nabla \bar{R}_\theta = \frac{1}{N}\sum_{n=1}^{N}\sum_{t=1}^{T}\left(Q^{\pi_\theta}\left(s_t^n, a_t^n\right) - V^{\pi_\theta}\left(s_t^n\right)\right) \cdot \nabla \log p\left(a_t^n | s_t^n\right).$$

If communication is allowed, agents can encode their information, such as observations and actions, into a finite length of messages sent to each other and receive messages from others. Thus, agents may take the received messages as a supplement to observations o_t^k, to obtain the optimal action conditioning on the approximate global state s_t.

Communication-based methods mainly focus on how to transmit high-quality communication messages. Specifically, the following contents should be considered: What should be encoded in the communication messages; How to update communication policy.

The rest of this article is structured as follows. In Section IV, we propose our learning framework based on answering the above question, but focus on "How to update communication policy." We then evaluate this approach in different environment settings in Section V, and conclude with Section VI.

IV. Causal Inference Based Communication

Humans can make effective use of communication through language to share information for social learning and group coordination. The agent's motivation for adjusting the communication vector arises from the influence intrinsic motivation which gives an agent bonus for having causal influences on another agent's communication messages. Specifically, in communication-based MARL, agents redesign the communication policy by an influential reward as i_t^k, which is the causal influence reward calculated from the observed communication vectors from other agents.

A. Communication Framework

The common choice for modeling communication policy π_c^k of agent k is to encode the current observation o_t^k, communication vectors $\bar{c}_t^k$ (note that at time step t, the $\bar{c}_t^k$ observed

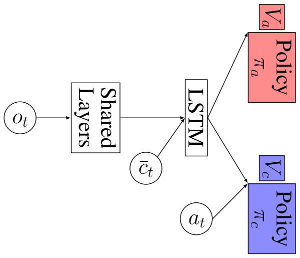


Fig. 3. Classic policy learning structure.

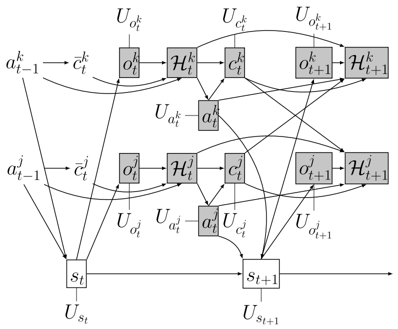


Fig. 4. SCM for MA-POMDPs. All conditional distributions are expressed as deterministic functions (squares colored gray). SCMs model environments using random variables U to summarize immutable aspects related to observations. Here, some of the immutable aspects are observed, such as each agent's own memory. Some are unobserved, which represent characteristics of environments or the noise.

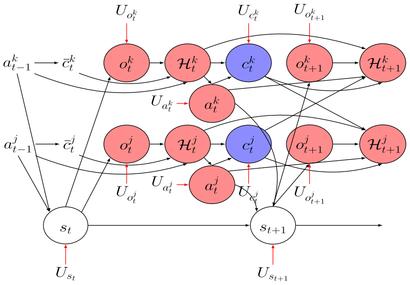


Fig. 5. CD. We mark immutable aspects as arrow colored red such as noise in the environment (U_o) and unobservant controlling variables (U_s and U_a).

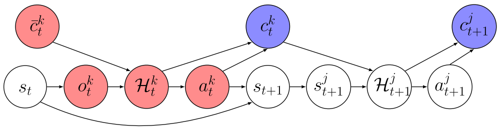


Fig. 6. CCM. To provide concise causal analysis, we ignore several unobservant variables from environment or noise. Because these unobservant variables have no mathematical compact on theoretical analysis in communication vectors.

by agent k are outputs from other agents' communication policy at time step $t-1$) and action a_t^k when creating policy network. Meanwhile, the action policy π_a is built on the current observation o_t^k, communication vectors $\bar{c}_t^k$.

The two policies π_a and π_c according to Fig. 3 may maintain shared layers to extract the features from the observed o_t^k and $\bar{c}_t^k$, but followed by two separate heads. Each head corresponds to the generation of actions and communication vectors from two policies. Thus, both π_a and π_c take o_t^k and $\bar{c}_{t-1}^k$ as inputs, except that π_c also need current action a_t^k to generate a more comprehensive communication vector c_t^k. For action head, we optimize the policy π_a^i by A3C algorithms

$$\begin{aligned}\nabla_\omega J\left(\pi_i^a\right) &= \mathbb{E}_{\pi_i^a,\pi_i^c}\left[\nabla_\omega \log \pi_i^a(a_i|s_i)\hat{Q}_i^a(s_i,a_i)\right]\\ &\approx \mathbb{E}_{\pi_i^a,\pi_i^c}\left[\nabla_\omega \hat{Q}_i^a\left(o_t^i,\bar{c}_t^i,a_t^i\right)\right].\end{aligned} \tag{1}$$

B. Counterfactual Policy Update

The proposed methods based on the above policy model are mainly updated in a centralized learning and decentralized execution manner. This condition is hard to guarantee in practice. In this section, we propose the counterfactual communication policy update, which overcomes this limitation. Main ideas underlay this counterfactual update: 1) analyze the communication procedure from a causal inference perspective and 2) update the communication policy by an extra influential reward i_t^k calculated from causal inference. The remainder of this section describes these ideas.

1) Causal Communication Models: Assessing causal influence rewards through counterfactual reasoning involves conditioning on a set of variables observed in a given state, and asking how the outcome would change if some variables were different, while others remained the same. This method allows us to evaluate causal effects of an agent k's communication vector c_t^k at time step t, on the other agent j's communication vector c_{t+1}^j, in the global environment state s_t at two consecutive time steps. The SCM of MA-POMDP can be concluded as Fig. 4, and the causal diagram (CD) behind the SCM can be present as Fig. 5. Further, according to [38], the immutable variables are concerned following RL-based assumption 1, and $\mathcal{H}$ is the replay buffer for policy updating.

Assumption 1 (Causal Markov Assumption): The causal Markov assumption, states that conditional on its direct causes, a variable V_t^j is independent of any variable for which it is not a cause, which is mathematically equivalent to the statement that the density $f(v_t^j)$ of the variables V_t^j in directed acyclic graph (DAG) satisfies the Markov factorization $f(v_t^j) = \prod f(v_t^j|pa_t^j)$ in POMDP CDs, where pa_t^j is the parent node of v_t^j.

The Causal Markov Assumption 1 guarantees the independence of each agent at a fixed time step. Thus, the causal relationship among agents can be captured *if and only if* they are considered at different time steps. We also can formalize the two agents' (k and j) interactions with environment and communication with each other at time steps t and $t+1$ as a CD in Fig. 6, in which the $\bar{c}_t^k = \oplus c_{t-1}^j (j \neq k)$ is a concatenation of comm. vectors received from other agents at time step $t-1$.

Theorem 1 (Properties of Communication Causal Model (CCM)): The CCM satisfies three conditions: Positivity, Consistency, and Exchangeability required for causal inference.

Such CCMs satisfy conditions of Properties of CCM 1, shown in Fig. 6. Thus, the counterfactual comm. reward of agent k to agent j at time step t is quantified by the causal effect from c_t^k to c_{t+1}^j colored blue.

TABLE I

THREE DIFFERENT SETTINGS CORRESPOND TO THREE DIFFICULTY LEVELS. THE REWARD OF THE AGENT IS THE SUM OF THE WAITING TIME AND THE PENALTY OF COLLISIONS. THEREFORE, MORE ENTRY POINTS, VEHICLES, ROUTES WILL INCREASE THE PROBABILITY ON THE VEHICLE'S COLLISION, WHICH LEADS TO LOW REWARDS

Difficulty	P-arrive		Grid Size	N-total	Arrival Points	Routes per Entry Point	Two-Way	Junctions
	Start	End						
Easy	0.1	0.2	7	5	2	1	F	1
Medium	0.02	0.05	14	10	4	3	T	1
Hard	0.03	0.05	18	20	8	7	T	4

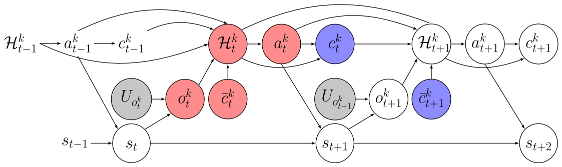


Fig. 7. Single-agent CD at consecutive time steps. Nodes colored gray are immutable variables representing noises from interactions with environment. Nodes colored blue and red are treatment/outcome and confounders considered in Fig. 6. Thus, in this sequential (not time-fixed) situation, treatment is a cause of the confounder at the next time step.

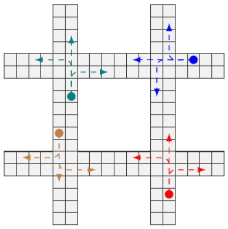


Fig. 8. Simulate a traffic junction environment. Each agent can observe itself and vision, path ahead of it.

TABLE II

COMPARATIVE RESULTS ON THREE LEVELS, WHERE SUCCESS RATES (SUCC. RATES) ARE LISTED. BOLD FONT INDICATES BEST RESULT OBTAINED FOR ANALYSIS

Model	Easy	Medium	Hard
IC	30.2 ± 0.4	3.4 ± 0.5	47.9 ± 2.9
CommNet	93.0 ± 4.2	54.3 ± 14.2	50.2 ± 3.5
IC3	$\mathbf{93.0 \pm 3.7}$	89.3 ± 2.5	72.4 ± 9.6
IC2	$\mathbf{93.0 \pm 3.7}$	$\mathbf{91.0 \pm 4.7}$	$\mathbf{77.0 \pm 13.7}$
FD-MARL ($\lvert c\rvert = 8$)	92.7 ± 3.4	$\mathbf{90.2 \pm 2.4}$	$\mathbf{73.7 \pm 11.2}$
FD-MARL ($\lvert c\rvert = 18$)	92.7 ± 3.9	$\mathbf{90.2 \pm 4.8}$	$\mathbf{73.7 \pm 12.7}$

2) Influential Communication Rewards: Theoretically, influential comm. rewards can be calculated as follows:

$$i_t^k = \sum_{j \neq k} \left[D_{\mathrm{KL}} \left[p\left(c_{t+1}^j | c_t^k, L_t^k\right) \| p\left(c_{t+1}^j | L_t^k\right) \right] \right]. \tag{2}$$

The marginal probability $p(c_{t+1}^j | L_t^k)$ in (2) can be derived if c_{t+1}^j are sampled after interventions on $\tilde{c}_t^k$ (which can also be defined as $do(\tilde{c}_t^k)$ [39]). Further, according to the intuitive idea that training agents to maximize the mutual information (MI) results in more coordinated behaviors [31], the relationship between causal effect and MI also arises in the following equation:

$$\begin{aligned} I\left(c_{t+1}^j, c_t^k\right) &= \mathbb{E}_\tau \left[D_{\mathrm{KL}} \left[p\left(c_{t+1}^j | c_t^k, L_t^k\right) \| p\left(c_{t+1}^j | L_t^k\right) \right] \middle| L_t^k \right] \\ &\approx \frac{1}{N} D_{\mathrm{KL}} \left[p\left(c_{t+1}^j | c_t^k, L_t^k\right) \| p\left(c_{t+1}^j | L_t^k\right) \right]. \end{aligned} \tag{3}$$

Equation (3) suggests that after sampling N independent trajectories τ from the environment with same confounders L_t^k, the MI of the treatment and the outcome variables (c_t^k and c_{t+1}^j) can be approximated by the causal effects. Note that MI does not convey the idea that a directed path from outcome c_{t+1}^j to treatment c_t^k may appear in CCMs. MI reveals the truth that the confounders affect the treatments, and the treatments affect the confounders, which is treatment-confounder feedback commonly seen in Fig. 7 representing sequential time-varying treatments [40]. Thus, MI, as the influential reward, may also bear an implicit link between variables at consecutive time steps within each agent.

Thus, the learning process consists of two sorts of A3C-based policy updating: the action policy π_a based on actual rewards r_a and communication policy π_c of each agent are updated based on local trajectories. And updating the communication policy π_c needs extra estimations on communication vector's effect to obtain influential rewards r_c, shown in the following equation:

$$\begin{aligned} \nabla_\theta J\left(\pi_i^c\right) &= \mathbb{E}_{\pi_i^a, \pi_i^c} \left[\nabla_\theta \log \pi_i^c (c_i | s_i) \hat{Q}_i^c (s_t, c_t) \right] \\ &\approx \mathbb{E}_{\pi_i^a, \pi_i^c} \left[\nabla_\theta \hat{Q}_i^c \left(o_t^i, \bar{c}_t^i, c_t^i\right) \right]. \end{aligned}$$

V. EXPERIMENTS

In this section, we demonstrate FD-MARL's efficacy on a diverse suite of MA cooperative domains, including the full spectrum of benchmarks with homogeneous agents, heterogeneous agents, and SSD games. Besides, a novel FL domain with two agents is considered as a test-bed. And experiments varying in difficulties are conducted to investigate the effectiveness of fully decentralized communication MA frameworks introduced in FD-MARL. All experiments are performed based on RLlib framework [41], which is an open-source library for reinforcement learning that offers both high scalability and a unified air position indicator (API) for a variety of applications.

A. Traffic Control

In the traffic junction [20], cars may enter a junction from every entry points with a probability p_{arr}. The task has three difficulty levels, which vary in the number of possible routes, entry points, and junctions. This environment is a common

TABLE III

DIFFICULTY OF THE GAME AI IS SET TO VERY DIFFICULT SEVEN. OUR EXPERIMENTS SUGGEST THAT ALL GAME TYPES VARYING IN HOMOGENEITY AND SYMMETRIC ARE CONSIDERED

Name	Ally Units	Enemy Units	Type
3m	3 Marines	3 Marines	homogeneous & symmetric
2s3z	2 Stalkers & 3 Zealots	2 Stalkers & 3 Zealots	heterogeneous & symmetric
5m_vs_6m	5 Marines	6 Marines	homogeneous & asymmetric
3s5z_vs_3s6z	3 Stalkers & 5 Zealots	3 Stalkers & 6 Zealots	heterogeneous & asymmetric
2c_vs_64zg	2 Colossi	64 Zerglings	micro-trick: positioning

benchmark for testing whether the communication mechanism is working by setting the vision of cars to be 0, shown in Table I and Fig. 8.

In this domain, the baselines are independent controller (IC), CommNet, IC3, and IC2. In IC, model is applied individually to all observations from agents to produce actions. CommNet allows communication over a channel where an agent is fed with the average of hidden state representations of other agents. IC3 is a revised CommNet equipped with the gating action (gt) and individualized rewards. IC2 measures and quantifies the causal effect between agents via the centralized critic and train the prior network to determine communication. It's worth noting that IC2 is also principled on causal analysis, but IC2 needs joint observations and actions of all agents, which is built on centralized training.

We evaluate FD-MARL in three difficulty levels followed [18]. The results are listed in Table II, where $|c|$ is the length of the communication vector. Each item in Table II is an average of 5 independent runs. Obviously, under the same difficulty level, FD-MARL methods are comparable to or outperform the baselines. Moreover, larger length of the communication vectors may also induce larger residual performance variance of FD-MARL. This is because such long messages make exploration slower for policy network learning, and make communication network learning difficult.

Overall, the results provide FD-MARL empirical support for performance in fully-cooperative scenarios with homogeneous agents. When multiple agents share the team reward, FD-MARL enables causal communication among agents to facilitate efficient learning without centralized controller containing full system information.

B. StarCraftII Domain

StarCraftII is a popular real-time strategy (RTS) game invented by Blizzard company. In a regular full game of StarCraftII, one or more humans compete against each other or against a built-in game artificial intelligence (AI) to gather resources, construct buildings, and build armies of units to defeat their opponents. Thus, StarCraftII is a challenging environment for RL because it has a large observation-action space, many different unit types, and stochasticity.

To fully investigate the scalability of FD-MARL in more realistic and complex scenarios, we test it on the StarCraft multi-agent challenge (SMAC) [42]. SMAC includes more challenging scenarios and offers a set of interesting microtrick challenges that require a higher level of cooperation and a specific micromanagement trick to defeat the enemy. The difficulty of the game AI is set to very difficult (7). FD-MARL is tested on five tasks covering all game types: 3m, 2s3z, 5m_vs_6m, 3s5z_vs_3s6z, and 2c_vs_64zg. The brief declaration on those games is listed in Table III. At each time step, agents receive local observations drawn within their field of view and encompass information about the map within a circular area around each unit and with a radius equal to the sight range. Obviously, the sight range makes the environment partially observable from the standpoint of each agent. Agents can only observe other agents if they are both alive and located within the sight range. Hence, there is no way for agents to determine whether their teammates are far away or dead. The discrete set of actions which agents are allowed to take consists of $move$ (four directions: north, south, east, or west), $attack$ on a specific enemy, $stop$ and $no-op$. Dead agents can only take no-op action while live agents cannot. The overall goal is to get the highest win rate for battle scenarios. We adopt the default setting for a shaped reward signal calculated from the hit-point damage dealt and received by agents, some positive (negative) reward after having enemy (allied) units killed and/or a positive (negative) bonus for winning (losing) the battle suggested by SMAC [42].

TABLE IV

RESULTS ON ALL GAME TYPES ARE LISTED, WITH BOLD FONT INDICATING BEST RESULT OBTAINED FOR ANALYSIS

Name	QMIX	VDN	IMAC	FD-MARL
3m	2.78 ± 1.81	2.82 ± 2.67	3.14 ± 1.63	$\mathbf{3.23 \pm 1.42}$
2s3z	6.12 ± 1.12	6.31 ± 1.42	6.74 ± 1.41	$\mathbf{7.24 \pm 1.53}$
5m_vs_6m	1.12 ± 0.97	1.31 ± 0.86	1.51 ± 0.76	$\mathbf{1.72 \pm 0.58}$
3s5z_vs_3s6z	3.21 ± 1.76	4.23 ± 1.31	4.76 ± 1.87	$\mathbf{5.56 \pm 2.03}$
2c_vs_64zg	5.87 ± 1.26	7.69 ± 0.97	10.32 ± 1.56	$\mathbf{13.82 \pm 1.48}$

The three baselines are mixing individual Q-networks (QMIX) [29], value decomposition network (VDN) [43] and informative multi-agent communication (IMAC) based on QMIX [32]. Both QMIX and VDN are value-based MA algorithms, while VDN trains individual agents with a novel VDN architecture, which learns to decompose the team value function into agent-wise value functions. IMAC adopts a CTDE paradigm, and further relax it by allowing agents to communicate. The final results in Table IV on five tasks suggest FD-MARL's performance, especially in setting.

The results in Fig. 9 demonstrate that FD-MARL is effective, particularly in maps with heterogeneous agents. Using the causality-based communication leads to informative information transmission among agents which we conjecture improves training. Thus, fully decentralized learning can be achieved through causality-based communication in complex domains

TABLE V

COMPARATIVE RESULTS ON GRID-WORLD, WHERE SUCCESS RATE (SUCC. RATE, THE HIGHER, THE BETTER), AVERAGE REWARD (AVG. RWD., THE HIGHER, THE BETTER), AND TRAJECTORY DIFFERENCE (TRAJ. DIFF., THE LOWER, THE BETTER) ARE LISTED

Metric	Method	Grid 8 × 8		Grid 16 × 16		Grid 32 × 32	
		With dist. info.	Without dist. info.	With dist. info.	Without dist. info.	With dist. info.	Without dist. info.
Succ. rate	comm. length 8	94.87%	**92.31%**	81.87%	80.31%	**83.17%**	80.77%
	comm. length 18	**95.32%**	91.81%	**83.32%**	**81.81%**	82.11%	**81.13%**
	independent A3C	56.86%	42.17%	48.86%	39.87%	42.16%	37.64%
Traj. diff.	comm. length 8	**0.23**	**0.31**	1.63	**3.31**	3.12	4.12
	comm. length 18	0.27	0.35	**1.47**	3.57	**2.82**	**3.86**
	independent A3C	0.37	0.45	2.57	5.45	4.57	6.15
Avg. rwd.	comm. length 8	18.91	**18.01**	**−88.91**	−97.61	**−187.01**	−218.61
	comm. length 18	**19.12**	17.72	−89.72	**−95.12**	−189.16	**−205.21**
	independent A3C	0.89	−1.71	−123.19	−148.15	−231.43	−278.12

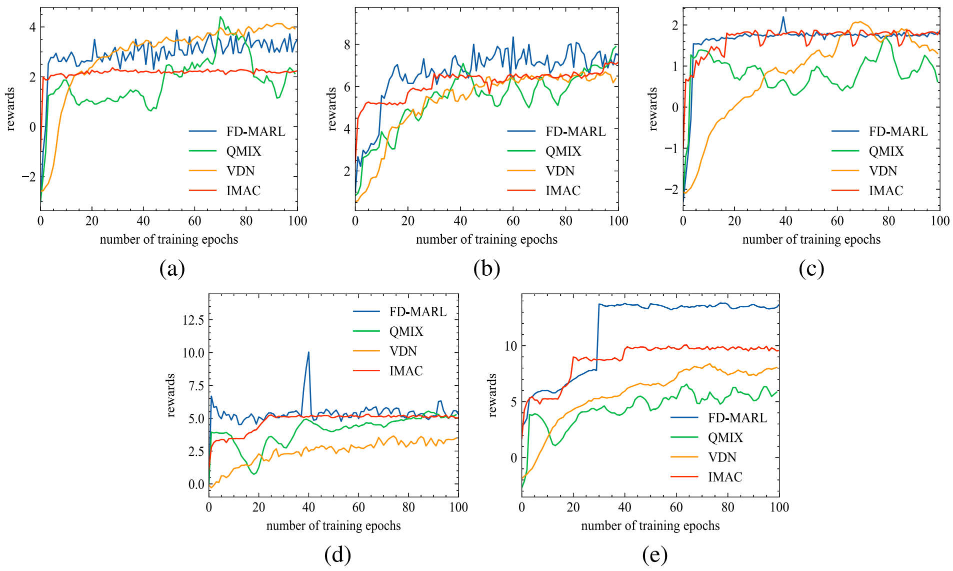


Fig. 9. Average performances of three algorithms on five independent experiments are drawn within each scenario. Obviously, FD-MARL demonstrates its learning ability in speed and performance. (a) 3m scenario. (b) 2s3z scenario. (c) 5m_vs_6m scenario. (d) 3s5z_vs_3s6z scenario. (e) 2c_vs_64zg scenario.

where state spaces are high-dimensional and participants are heterogeneous with satisfactory learning speed.

C. SSD Games

SSD games are MA games with a game-theoretic payoff structure. An individual agent may obtain higher reward in the short-term by taking noncooperative behaviors, but the total reward per agent will be higher if all agents cooperate.

In cleanup scenario, apples are generated based on the amount of waste in a nearby river. Agents can use a cleaning beam action to clean the river when they are positioned in it, or consume the apples the other agent produces. They also have a fining beam action to fine nearby agents by −50 reward. Agents must clean a river before apples can grow, but are not able to harvest apples while cleaning. Thus, agents must efficiently balance harvesting apples and cleaning the river, and allow agents cleaning the river a chance to consume apples. The baselines includes MADDPG and social influential communication. Social influential communication is principled on centralized learning, which gives an agent additional reward for having a causal influence on another agent's actions. As is evident in Fig. 10, introducing centralization of other agents' observations and actions with influential rewards works in SSD, but having causality-based communication eventually leads to higher or comparable collective rewards in decentralized manner. The choice of communication length plays an important role in the model's stability. High communication complexity allows richer information transformation but exhibits less stable training. FD-MARL generally achieves satisfactory performance, suggesting that a decentralized learning framework with causality-based communication.

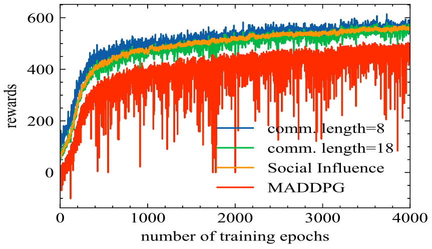
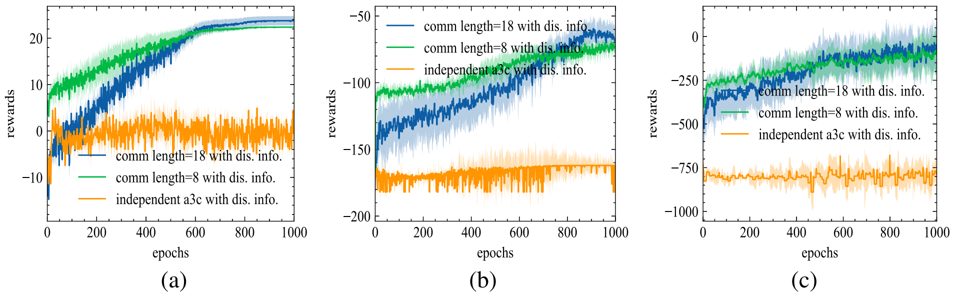
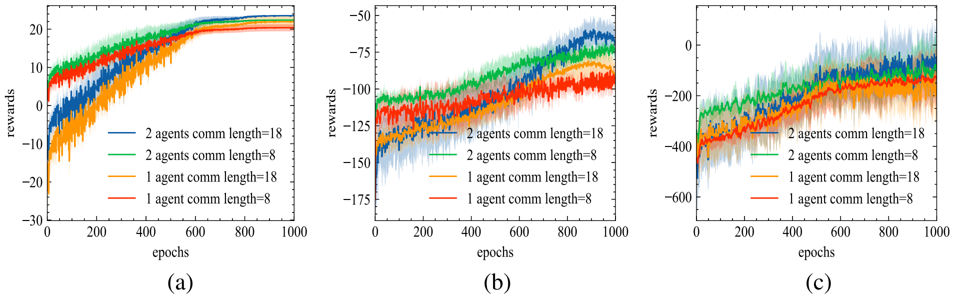


Fig. 10. Total collective reward obtained Cleanup. Agents trained with FD-MARL exhibit comparable performance to the baselines.



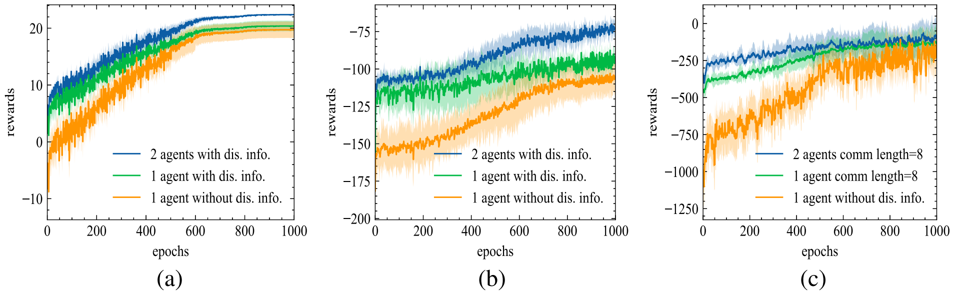
(a) (b) (c)

Fig. 11. When the length of these vectors is fixed to the large size of 18, agents may take a longer time to organize their communication messages and may increase in the variance of the rewards. (a) Grid size 8 × 8. (b) Grid size 16 × 16. (c) Grid size 32 × 32.



(a) (b) (c)

Fig. 12. We disturb the agent's interaction with the environment by reward cancellation. (a) Grid size 8 × 8. (b) Grid size 16 × 16. (c) Grid size 32 × 32.



(a) (b) (c)

Fig. 13. With the communication policy's update, the agents can obtain informative messages to ameliorate the learning process. Thus, the FD-MARL can achieve faster convergence with less variance. (a) Grid size 8 × 8. (b) Grid size 16 × 16. (c) Grid size 32 × 32.

D. Grid-World Domain

In this FRL domain, experiments are conducted in a synthetic grid-world [35] varying in sizes (8 × 8, 16 × 16 and 32 × 32). In this task, two agents are initially placed in different positions, and the goal is to meet each other through the shortest path without hitting walls. Only one agent can take advantage of instant rewards, while the other one could not receive the reward signal from the environment. Thus, learning becomes difficult due to partial observation and limited reward signals. The reward is set to the same values as [35]. Both π_a and π_c are modeled as long short-term memory (LSTM) with fixed sequence length 4, and the observation is 3 × 3 grid centered on agents' positions. All these settings remain the same in all difficulty levels, which is challenging when gird size is getting large. The communication messages are real-valued vectors with each dimension bounded within $[-1, 1]$. We list the main results in Table V.

At first, we conduct fundamental tests by letting both agents receive rewards shown in Fig. 11. The independent A3C performs traditional A3C without communication. FD-MARL with the length of comm. vector set to be 8 and 18 are tested in three different settings of grid world. Obviously, larger length of comm. vector enforces a higher learning upper bound, while require more epochs organizing the comm. policy to achieve convergence and is accessible to suffer from instability.

Then, results are provided on the scenario in which only one of the agents can receive rewards from the environment, but both agents can decide their influential rewards shown in Fig. 12. Intuitively, agents may approach convergence with more time-consuming, and the communication variance may be larger due to the rewards' cancellation in an agent. The results in Fig. 12 support our intuitive thoughts. Besides, the learning upper bound may be decreased for the missing of significant information, even if in relatively naive environment. Thus, the agents' ability on gathering information from environments is significant in practice.

In this domain, we suggest that the effectiveness of communication can be demonstrated more reasonably and directly if the positive reward $r_d = c/md(\cdot)$ related to distance information is eliminated. Thus, the agent can only obtain an informative reward when achieving a goal. Note that, in r_d, c is a positive coefficient, and $md(\cdot)$ is the Manhattan Distance between agents. Although only 1 agent can obtain rewards, the distance-related reward signal is quite informative. This setting is functionally equivalent to provide substantial goal-directed information to agents. However, this setting is less challenging for agents when they are ignorant of any goal-related information, which needs agent to ensure appropriate actions and improve communication to achieve the goal.

Thus, we conduct additional comparisons to investigate the effectiveness of communication in such partial observant and even more restrictive conditions. Results in Fig. 13 substantiate our claim that fully decentralized learning can be achieved through communication in FRL domains, even with more strict conditions such as incomplete information provided.

VI. Conclusion

In this work, we introduced FD-MARL which aims to solve MA tasks in various cooperation settings by learning to communicate. FD-MARL allows agents to modify communication message by choosing the counterfactual that bears the most significant influence on others. Thus, the continuous communication based on causal analysis enables efficient information transformation in a fully decentralized manner.

The proposed method is tested on several MA benchmarks. The experimental results show that FD-MARL is able to cooperate efficiently in the fully decentralized manner. Specifically, the experiments in MARL benchmarks illustrate that FD-MARL provides satisfactory asymptotic performance with all forms of agent heterogeneity, including homogeneous agents, heterogeneous agents, and SSDs. In the FL domain, FD-MARL performs satisfactorily under strict conditions such as less informative signals from environment, and exhibits the scalability in such scenarios close to practical requirements.

The experimental results demonstrate FD-MARL is a promising method for dealing with real-world MA problems. In future, we would like to explore multichannel causal communication. Thus, agents can adapt to new tasks by leveraging the domain-specific information contained in prior tasks.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*, (Adaptive Computation and Machine Learning). Cambridge, MA, USA: MIT Press, 1998.

[2] P. Abbeel, V. Ganapathi, and A. Y. Ng, “Learning vehicular dynamics, with application to modeling helicopters,” in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada, Dec. 2005, pp. 1–8.

[3] M. S. Emigh, E. G. Kriminger, A. J. Brockmeier, J. C. Príncipe, and P. M. Pardalos, “Reinforcement learning in video games using nearest neighbor interpolation and metric learning,” *IEEE Trans. Comput. Intell. AI Games*, vol. 8, no. 1, pp. 56–66, Mar. 2016.

[4] Y. Duan, X. Chen, R. Houthooft, J. Schulman, and P. Abbeel, “Benchmarking deep reinforcement learning for continuous control,” in *Proc. Int. Conf. Int. Conf. Mach. Learn.*, May 2016, pp. 1329–1338.

[5] D. Silver *et al.*, “Mastering the game of go without human knowledge,” *Nature*, vol. 550, no. 7676, pp. 354–359, 2017.

[6] O. Vinyals *et al.*, “Grandmaster level in starcraft II using multi-agent reinforcement learning,” *Nature*, vol. 575, no. 7782, pp. 350–354, 2019.

[7] P. Hernandez-Leal, B. Kartal, and E. Taylor, “A survey and critique of multiagent deep reinforcement learning,” *Auto. Agents Multi-Agent Syst.*, vol. 33, pp. 750–797, Oct. 2019.

[8] S. Shalev-Shwartz, S. Shammah, and A. Shashua, “Safe, multi-agent, reinforcement learning for autonomous driving,” 2016, *arXiv:1610.03295*.

[9] M. Lauer and M. Riedmiller, “An algorithm for distributed reinforcement learning in cooperative multi-agent systems,” in *Proc. Int. Conf. Mach. Learn.*, Jun. 2000, pp. 535–542.

[10] R. Lowe, Y. Wu, A. Tamar, J. Harb, O. P. Abbeel, and I. Mordatch, “Multi-agent actor-critic for mixed cooperative-competitive environments,” in *Proc. Adv. Neural Info. Process. Syst.*, Long Beach, CA, USA, Dec. 2017, pp. 6379–6390.

[11] S. Omidshafiei, J. Pazis, C. Amato, J. How, and J. Vian, “Deep decentralized multi-task multi-agent reinforcement learning under partial observability,” in *Proc. Int. Conf. Mach. Learn.*, Mar. 2017, pp. 2681–2690.

[12] S. Iqbal and F. Sha, “Actor-attention-critic for multi-agent reinforcement learning,” in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, Long Beach, CA, USA, Jun. 2019, pp. 2961–2970.

[13] T. T. Doan, S. T. Maguluri, and J. Romberg, “Finite-time analysis of distributed TD(0) with linear function approximation on multi-agent reinforcement learning,” in *Proc. 36th Int. Conf. Mach. Learn.*, Long Beach, CA, Jun. 2019, pp. 1626–1635.

[14] J. Hu and M. P. Wellman, “Multiagent reinforcement learning: Theoretical framework and an algorithm,” in *Proc. 15th Int. Conf. Mach. Learn.*, Jul. 1998, pp. 242–250.

[15] G. Tesauro, “Extending Q-learning to general adaptive multi-agent systems,” in *Proc. Adv. Neural Inf. Process. Syst.*, Vancouver, BC, Canada: MIT Press, Dec. 2003, pp. 871–878.

[16] V. Conitzer and T. Sandholm, “AWESOME: A general multiagent learning algorithm that converges in self-play and learns a best response against stationary opponents,” *Mach. Learn.*, vol. 67, nos. 1–2, pp. 23–43, May 2007.

[17] J. Z. Leibo, V. F. Zambaldi, M. Lanctot, J. Marecki, and T. Graepel, “Multi-agent reinforcement learning in sequential social dilemmas,” in *Proc. 16th Conf. Auto. Agents Multi-Agent Syst.*, São Paulo, Brazil, May 2017, pp. 464–473.

[18] A. Singh, T. Jain, and S. Sukhbaatar, “Learning when to communicate at scale in multiagent cooperative and competitive tasks,” in *Proc. 7th Int. Conf. Learn. Represent.*, New Orleans, LA, USA, May 2019, pp. 1–16.

[19] J. Foerster, I. A. Assael, N. de Freitas, and S. Whiteson, “Learning to communicate with deep multi-agent reinforcement learning,” in *Proc. Adv. Neural Inf. Process. Sys.*, Dec. 2016, pp. 2137–2145.

[20] S. Sukhbaatar, A. Szlam, and R. Fergus, “Learning multiagent communication with backpropagation,” in *Proc. Adv. Neural Inf. Process. Syst.*, May 2016, pp. 2244–2252.

[21] S. Havrylov and I. Titov, “Emergence of language with multi-agent games: Learning to communicate with sequences of symbols,” in *Proc. 5th Int. Conf. Learn. Represent.*, Toulon, France, Apr. 2017, pp. 1146–1155.

[22] G. Chen, “A new framework for multi-agent reinforcement learning—Centralized training and exploration with decentralized execution via policy distillation,” 2019, *arXiv:1910.09152*.

[23] K. Bonawitz *et al.*, “Towards federated learning at scale: System design,” 2019, *arXiv:1902.01046*.

[24] R. Shokri and V. Shmatikov, “Privacy-preserving deep learning,” in *Proc. 22nd ACM SIGSAC Conf. Comput. Commun. Secur.*, I. Ray, N. Li, and C. Kruegel, Eds., Oct. 2015, pp. 1310–1321.

[25] J. P. Pearce, R. T. Maheswaran, and M. Tambe, “Solution sets for DCOPs and graphical games,” in *Proc. 5th Int. Joint Conf. Auto. Agents Multiagent Syst. (AAMAS)*, 2006, pp. 577–584.

[26] F. Fioretto, E. Pontelli, and W. Yeoh, “Distributed constraint optimization problems and applications: A survey,” *J. Artif. Intell. Res.*, vol. 61, pp. 623–698, Mar. 2018.

[27] M. Tan, “Multi-agent reinforcement learning: Independent vs. cooperative agents,” in *Proc. 10th Int. Conf. Mach. Learn.*, Jun. 1993, pp. 330–337.

[28] J. N. Foerster *et al.*, “Stabilising experience replay for deep multi-agent reinforcement learning,” in *Proc. 34th Int. Conf. Mach. Learn.*, Sydney, NSW, Australia, Aug. 2017, pp. 1146–1155.

[29] T. Rashid, M. Samvelyan, C. S. de Witt, G. Farquhar, J. N. Foerster, and S. Whiteson, “QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning,” in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, Jul. 2018, pp. 4292–4301.

[30] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, “Counterfactual multi-agent policy gradients,” in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2018, pp. 2974–2982.

[31] N. Jaques *et al.*, “Social influence as intrinsic motivation for multi-agent deep reinforcement learning,” in *Proc. 36th Int. Conf. Mach. Learn.*, Long Beach, CA, Jun. 2019, pp. 3040–3049.

[32] R. Wang, X. He, R. Yu, W. Qiu, B. An, and Z. Rabinovich, “Learning efficient multi-agent communication: An information bottleneck approach,” in *Proc. 37th Int. Conf. Mach. Learn., Virtual Event*, vol. 119, Jul. 2020, pp. 9908–9918.

[33] J. N. Foerster, G. Farquhar, T. Afouras, N. Nardelli, and S. Whiteson, “Counterfactual multi-agent policy gradients,” in *Proc. AAAI Conf. Artif. Intell.*, Feb. 2018, pp. 2974–2982.

[34] Z. Ding, T. Huang, and Z. Lu, “Learning individually inferred communication for multi-agent cooperation,” in *Proc. Annu. Conf. Neural Inf. Process. Syst., Virtual Event*, Dec. 2020, pp. 22069–22079.

[35] H. H. Zhuo, W. Feng, Q. Xu, Q. Yang, and Y. Lin, “Federated reinforcement learning,” 2019, *arXiv:1901.08277*.

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

[37] N. Ono and K. Fukumoto, “A modular approach to multi-agent reinforcement learning,” in *Proc. 2nd Int. Conf. Multiagent Syst., Distrib. Artif. Intell. Meets Mach. Learn., Learn. Multi-Agent Environ.*, Dec. 1996, pp. 25–39.

[38] L. Buesing *et al.*, “Woulda, Coulda, Shoulda: Counterfactually-guided policy search,” in *Proc. 7th Int. Conf. Learn. Represent.*, New Orleans, LA, USA, May 2019, pp. 1–15.

[39] J. Pearl, “Causal inference in statistics: An overview,” *Statist. Surv.*, vol. 3, pp. 96–146, Oct. 2009.

[40] M. A. Hernán and J. M. Robins, *Causal Inference: What If*. Boca Raton, FL: Chapman & Hall/CRC, 2019.

[41] E. Liang *et al.*, “Rllib: Abstractions for distributed reinforcement learning,” in *Proc. 35th Int. Conf. Mach. Learn.*, Stockholm, Sweden, Jul. 2018, pp. 3059–3068.

[42] M. Samvelyan *et al.*, “The StarCraft multi-agent challenge,” 2019, *arXiv:1902.04043*.

[43] P. Sunehag *et al.*, “Value-decomposition networks for cooperative multi-agent learning based on team reward,” in *Proc. Int. Conf. Auton. Agents MultiAgent Syst. (AAMAS)*, Jul. 2018, pp. 2085–2087.

[44] X. Zhang *et al.*, “LIDAR: Learning from imperfect demonstrations with advantage rectification,” *Frontiers Comput. Sci.*, vol. 16, no. 161312, 2021.

[45] Z. L. Ning *et al.*, “Intelligent resource allocation in mobile blockchain for privacy and security transactions: A deep reinforcement learning based approach,” *Sci. China Inf. Sci.*, vol. 64, no. 162303, 2021.

Han Wang received the B.Sc. degree in computer science from Xi’dian University, Xi’an, China, in 2014. She is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Nanjing University, Nanjing, China.

Her current research interests include reinforcement learning, multiagent system and machine learning.

Yang Yu received the B.Sc. and Ph.D. degrees in computer science from Nanjing University, Nanjing, China, in 2004 and 2011, respectively.

He is currently a Professor with the School of Artificial Intelligence, Nanjing University. His research interest includes machine learning. His current research interest includes real-world reinforcement learning.

Prof. Yu was granted the CCF-IEEE CS Young Scientist Award in 2020, recognized as one of the AI’s 10 to Watch by IEEE Intelligent Systems in 2018 and received the PAKDD Early Career Award in 2018. He was invited to give an Early Career Spotlight Talk in IJCAI’18. He was granted several conference best paper awards, including IDEAL’16, GECCO’11, PAKDD’08, and so on. His teams won the Champion of the 2018 OpenAI Retro Contest on transfer reinforcement learning and 2021 ICAPS Learning to Run a Power Network Challenge with Trust. He served as an Area Chair for NeurIPS, ICML, IJCAI, AAAI, and ACML. He is an Associate Editor of IEEE TRANSACTIONS ON EMERGING TOPICS IN COMPUTATIONAL INTELLIGENCE for the 2021 term.

Yuan Jiang received the Ph.D. degree in computer science from Nanjing University, Nanjing, China, in 2004.

She joined the School of Computer Science and Technology, Nanjing University, in 2004. She is currently working on machine learning, data mining, information retrieval, and other aspects of research. Her research interest is in machine learning, a subfield of artificial intelligence (AI).