Journal of Automation and Intelligence 3 (2024) 73–91

Contents lists available at ScienceDirect

Journal of Automation and Intelligence

journal homepage: www.keaipublishing.com/en/journals/journal-of-automation-and-intelligence/

Review article

A survey on multi-agent reinforcement learning and its application

Zepeng Ning, Lihua Xie [*]

School of Electrical and Electronic Engineering, Nanyang Technological University, Nanyang Avenue, 639798, Singapore

ARTICLE INFO

Keywords:
Benchmark environments
Multi-agent reinforcement learning
Multi-agent systems
Stochastic games

ABSTRACT

Multi-agent reinforcement learning (MARL) has been a rapidly evolving field. This paper presents a comprehensive survey of MARL and its applications. We trace the historical evolution of MARL, highlight its progress, and discuss related survey works. Then, we review the existing works addressing inherent challenges and those focusing on diverse applications. Some representative stochastic games, MARL means, spatial forms of MARL, and task classification are revisited. We then conduct an in-depth exploration of a variety of challenges encountered in MARL applications. We also address critical operational aspects, such as hyperparameter tuning and computational complexity, which are pivotal in practical implementations of MARL. Afterward, we make a thorough overview of the applications of MARL to intelligent machines and devices, chemical engineering, biotechnology, healthcare, and societal issues, which highlights the extensive potential and relevance of MARL within both current and future technological contexts. Our survey also encompasses a detailed examination of benchmark environments used in MARL research, which are instrumental in evaluating MARL algorithms and demonstrate the adaptability of MARL to diverse application scenarios. In the end, we give our prospect for MARL and discuss their related techniques and potential future applications.

1. Introduction

Multi-agent reinforcement learning (MARL) is an important subfield in the community of machine learning. The emergence of MARL marks a significant advancement in artificial intelligence, particularly in handling complex and dynamic environments with multiple interacting agents. Unlike traditional reinforcement learning (RL) that is applicable to single-agent systems, MARL concentrates on the behaviors of multiple learning agents coexisting within a shared environment. Hence, it must account for the interactions between agents and the environment, as well as the interactions between agents themselves.

1.1. Historical evolution and background

The initial concept of RL was rooted in trial-and-error borrowed from animal behaviors [1]. Based on trial-and-error learning, [2] invented neural-analog reinforcement calculators. Subsequently, innovations, like Q-learning [3], transitioned RL to a computational model. The introduction of deep reinforcement learning (DRL) by [4] was a pivotal milestone, enabling agents to handle high-dimensional problems and partly overcoming the curse of dimensionality. The versatility of DRL is demonstrated in its applications across diverse areas, from gameplay [5] to robotics [6].

Multi-agent systems (MASs) involve multiple interacting agents in a common environment and have found wide applications in robotics, telecommunications, and autonomous systems. In robotics, MASs can facilitate tasks, such as precision agriculture, underwater exploration, and rescue tasks, by leveraging the coordination among robots [7–9]. In terms of telecommunications, MASs play a vital role in optimizing network access, transmitting power control, and task off-loading, significantly improving network performance and efficiency [10,11]. Autonomous systems, particularly in vehicle networks, can benefit from MASs through improved decision-making and optimization of traffic management [12,13]. Both the versatility of MASs in dealing with complex dynamic environments and the ability of MASs to facilitate decentralized decision-making and learning have made them potentially useful in solving sophisticated real-world problems.

Obviously, MARL is beyond the scope of RL for single-agent systems. In MARL, the actions of an agent can have impacts on the rewards for the others, leading to a non-stationary environment that can affect learning efficiency and performance [14]. This issue is evident in applications such as precision agriculture [7], underwater exploration [8], and autonomous vehicles [12]. The critical role of sensory data in shaping the state space of each agent is emphasized in these applications [15], highlighting the need to develop MARL algorithms. The application of MARL in these areas requires algorithms that consider the actions of other agents and learn cooperative strategies to optimize global objectives, like maximizing network throughput or minimizing task execution latency [16]. Despite the existence of various challenges,

Peer review under responsibility of Chongqing University.

[*] Corresponding author.

E-mail addresses: zepeng.ning@ntu.edu.sg (Z. Ning), elhxie@ntu.edu.sg (L. Xie).

https://doi.org/10.1016/j.jai.2024.02.003
Received 16 January 2024; Accepted 15 February 2024
Available online 18 February 2024

the successful applications of MARL in a variety of domains showcase its potential in solving complex real-world problems. In telecommunications, MARL has been applied for dynamic multichannel access [10] and transmit power control [11]. In robotics, MARL enhances capabilities in cooperative navigation [17]. Swarm intelligence represents another facet of research in MASs, where agents interact locally without centralized control to achieve a collective objective [18]. To facilitate a unified decision-making process among agents, the development of swarm algorithms has received much attention [19]. On the other hand, MARL-based swarm behaviors have emerged as a prominent research area. [20] introduced a Q-learning-based algorithm for swarm systems named Q-learning real-time swarm (Q-RTS). Catering to real-time swarm intelligence systems, [21] proposed an FPGA implementation of Q-RTS. Recent advancements in the application of MARL to swarm systems have been made. Specifically, [22] concentrated their efforts on employing MARL to enhance the efficacy of UAV swarm communications in the presence of jamming; the work by [23] presented an innovative approach using an actor–critic DRL strategy to manage a collective of cooperative agents, where the learning of a Q-function utilizes global state information acting as a camera in swarm robots.

In addition, the adaptability of DRL leads to its extension into the multi-agent context, which brings about new challenges, such as training schemes [24] and computational complexity [25]. As a transition of DRL technique from a single-agent setting to a multi-agent setting marked by a shift from simpler representations to more nuanced and intricate environments, the extension of DRL to multi-agent environments amplifies the complexity and introduces a unique challenge to its applications. MADRL has become a rapidly growing field due to its capability of modeling complex and real-world problems. [26] delved into the learning mechanisms of agents from a game-theoretical perspective, highlighting the foundational role of strategic interaction in MADRL. This approach aligns with the broader trend in MADRL research, which increasingly addresses real-world applications [27,28]. The works [29,30] offer a structured classification of the deep learning algorithms employed in recent MADRL studies, providing a framework for understanding the algorithmic landscape of this field.

1.2. Previous surveys of MARL

Over ten years ago, the pace of studying MARL was relatively slow, with a limited number of surveys available. During this period, review works [31,32] provided significant insights into MARL algorithms, which built a stage for understanding the evolution of MARL strategies. Then, [33] made a review of coordination challenges in cooperative Markov games, highlighting the intricacies of achieving effective collaboration among agents. Further advancing this topic, the survey [34] offered a unique perspective by analyzing the evolutionary dynamics in MASs, which is crucial for understanding how MARL strategies evolve and adapt to changing environments and agent interactions.

Recent years have witnessed a surge in studies on MARL, and correspondingly, many more surveys can be found. Some surveys explored the integration of deep learning in MARL. [35] examined emergent behaviors, communication, and cooperative learning, which highlighted the significance of interaction dynamics in MADRL. [29] provided a critical analysis of MADRL and its applications in various environments, including competitive, cooperative, and mixed settings. This was then complemented by the systematic review of [28], which not only explored classical algorithms but also illuminated the current research progress and future directions, including model-free DRL and transfer learning. [24] categorized challenges in MADRL and provided a nuanced understanding of its implementation complexities. Their survey also highlighted advanced strategies to address these challenges in practical applications. Moreover, [36] focused on knowledge reuse and autonomy in MASs. This perspective is crucial for developing advanced learning strategies that can leverage past experience. Afterward, by integrating transfer learning into MARL, [37] presented a promising perspective for enhancing the learning capabilities of agents. Very recently, [38] comprehensively reviewed recent developments in MARL algorithms, categorized them, addressed their challenges and resolutions, and listed real-world applications and available MARL environments. [25] identified and anatomized some practical challenges such as centralized training and decentralized execution, which are important in understanding the operational dynamics of MARL systems. [39] provided an in-depth review of MARL covering its basic methods, application scenarios, and research trends, while highlighting the limitations and ethical constraints in its practical applications. As MARL continues to evolve, these perspectives provide a foundation for future research and real-world applications in increasingly complex and dynamic environments.

Several surveys on the applications of MARL have been reported. One primary area is the realm of network technology. [40] explored the MARL applied in vehicular networks, emphasizing its role in optimizing network functionality and enhancing communication efficiency. Similarly, the application of MARL to the future internet, as reviewed by [41], showcased its potential in managing complex networks and facilitating efficient data transfer and connectivity. [27] explored the potential of MARL in future wireless networks, with focuses on the areas of mobile edge computing, unmanned aerial vehicle (UAV) networks, and cell-free massive multiple-input–multiple-output (MIMO) communication. In the field of autonomous mobility, the survey conducted by [42] provided insights into how MARL could contribute to the development of autonomous vehicular systems, particularly focusing on the aspects of traffic management, route optimization, and vehicular coordination. [43] presented an extensive survey on MARL applied to connected and automated vehicles, which provided a classification-based analysis of current advancements and discussed existing challenges. As for robots, [44] probed the application of MARL to multi-robot systems by emphasizing its critical role in scenarios where individual robots are insufficient for task completion, and then classified relevant papers based on robot applications. Overall, these surveys on applications of MARL have offered a detailed overview of its implementation across diverse real-world systems.

1.3. Motivations of the current survey

MARL is still significant and is advancing quite fast nowadays. Novel approaches are proposed, and more challenges and applications are emerging. The existing surveys of previous works on the challenges and methods of MARL are getting outdated. Thus, survey works on MARL deserve to be updated, and new prospects for advances in MARL are necessary. Moreover, the communication issue and the complexity of learning have not been fully addressed in previous surveys. With respect to the MARL applications to chemical engineering, biotechnology, and medical treatment, related survey work is quite scanty, though there have been notable developments in MARL addressing these aspects. The same problem still exists in previous surveys involving MARL applied to human social issues, even though several surveys have reviewed this issue, see, e.g., [24,38,39,45], where quite a few related works have not been mentioned. Moreover, simulators and benchmark environments are crucial for the development and test of MARL algorithms. However, the existing related reviews are also insufficient. As MARL evolves, an in-depth review of these tools becomes essential. These aforementioned deficiencies motivate us to undertake a new survey of MARL theory and its challenges and applications.

The contributions of this survey are summarized as follows: *First*, we present an updated survey of the challenges in MARL and recent powerful approaches, as well as detailed reviews of the challenges in communication and learning complexity. This survey seeks to highlight these emerging challenges and to offer a current view of the difficulties and opportunities in MARL, and also aims to guide and encourage future research and application. *Second*, our survey intends to make up for the shortage of applications of MARL in chemical engineering, biotechnology, and medical treatment. We provide directions for the interdisciplinary nature of science and MARL and enhance the cross between them. *Third*, considering the effects of MARL on social dynamics and human behaviors have not been fully reviewed, we aim to fill this gap by giving a more detailed review of the applications of MARL to these realms. *At last*, regarding the limited discussion in current surveys of simulators and benchmark environments for MARL, our paper will provide an in-depth review of these simulation tools.

2. Related paradigms and strategies in MARL

2.1. Stochastic games for MARL

Stochastic game theory is a subfield of game theory that deals with the problem of decision-making in situations where the outcomes are partly random and partly under the control of one or more decision-makers. In MARL, stochastic games are used to model the interactions between multiple agents and their environment, where the agents may receive incomplete information about the state of the environment. Thus, the stochastic games may be partially observable. We first consider the fully observable games, then review the partially observable case.

2.1.1. Fully observable stochastic games

Before introducing the notion of stochastic games, we define the fundamental Markov decision processes (MDPs).

Markov decision processes. In an RL mechanism, the interaction between a learning agent and its environment is generally modeled as an MDP. Formally, an MDP can be defined by a tuple $\langle \mathbb{S}, \mathbb{A}, \mathcal{P}, \mathcal{R}, \gamma \rangle$, where $\mathbb{S}$ denotes the state space; $\mathbb{A}$ symbolizes the action space; $\mathcal{P} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow [0,1]$ means the transition probability function from state $s \in \mathbb{S}$ to $s' \in \mathbb{S}$ under the action $a \in \mathbb{A}$, i.e., $\mathcal{P}\left(s'|s,a\right) \triangleq \text{Prob}\left(s(t+1) = s'|s(t) = s, a(t) = a\right)$, $\forall t \in \mathbb{N}$; $\mathcal{R} : \mathbb{S} \times \mathbb{A} \times \mathbb{S} \rightarrow \mathbb{R}$ defines the reward function in the form of $\mathcal{R}\left(s,a,s'\right) = \mathcal{R}(s(t), a(t), s(t+1))$, $\forall t \in \mathbb{N}$; $\gamma \in [0,1]$ is the discount factor used for weighing the effect of immediate and future rewards. The learning agent interacts with the environment in discrete time steps. At each time step t, the agent is in some state $s(t) \in \mathbb{S}$ and selects an action $a(t) \in \mathbb{A}$. At time step $t+1$, the agent receives a reward for its action at time t, denoted by $r(t) = \mathcal{R}\left(s(t), a(t), s(t+1)\right)$, and moves into a new state $s(t+1)$ immediately. The value of the reward received by the agent is dependent on the transition from the state–action pair (s,a) to the state s'. The state transition function describes the model dynamics. Since the dynamics of the state have the Markov property, the future state only depends on the current state and current action but not on the historical states and actions. A policy π dominates which action to take based on the current state in either a deterministic or a probabilistic manner. The objective of learning a policy is to maximize its overall expected reward that is calculated as the expected sum of the discounted rewards $\mathcal{E}_\tau\left[\sum_{t=0}^{T} \gamma^t r(t)\right]$ over the trajectory $\tau \triangleq (s(0), a(0), s(1), a(1), \dots, s(T), a(T))$ that is a sequence of states and actions within the horizon T. A smaller value of the discount factor γ implies that the agent gives more emphasis to immediate rewards, whereas a larger value suggests a preference for rewards in the future. The optimal policy is the one that can maximize the reward function, which is the goal of RL.

Stochastic games. If multiple agents are involved, an MDP will become unsuitable to accurately represent the environment due to the impact of the actions of other agents. To tackle this problem, stochastic games are introduced as an expanded version of MDPs, which are also known as Markov games. A Markov game is described by the set $\left\langle \mathbb{I}, \mathbb{S}, \overline{\mathbb{A}}, \mathcal{P}, \left\{\mathcal{R}_i\right\}_{i \in \mathbb{I}}, \gamma \right\rangle$, where $\mathbb{I} \triangleq \{1, 2, \dots, N\}$ represents the set of N agents; $\mathbb{S}$ denotes the state space for all the agents; $\overline{\mathbb{A}} \triangleq \mathbb{A}_1 \times \mathbb{A}_2 \times \cdots \times \mathbb{A}_N$ represents the joint action space concatenated by $\mathbb{A}_i$, $i \in \mathbb{I}$, in which $\mathbb{A}_i$ represents the action space of the ith agent; $\mathcal{P} : \mathbb{S} \times \overline{\mathbb{A}} \times \mathbb{S} \rightarrow [0,1]$ is the function that governs the dynamic of the state $s(t)$ expressed as $\mathcal{P}\left(s'|s,\boldsymbol{a}\right) \triangleq \text{Prob}\left(s(t+1) = s'|s(t) = s, \boldsymbol{a}(t) = \boldsymbol{a}\right)$, $\forall t \in \mathbb{N}$, which indicates the probability of the game transition into the next state $s' \in \mathbb{S}$ if the current state is $s \in \mathbb{S}$ and the joint action is $\boldsymbol{a} \in \overline{\mathbb{A}}$; $\mathcal{R}_i : \mathbb{S} \times \overline{\mathbb{A}} \times \mathbb{S} \rightarrow \mathbb{R}$ denotes the function that assigns a reward for the ith agent, which indicates the immediate reward from $(s, \boldsymbol{a})$ to s' and is formulated as $\mathcal{R}_i\left(s, \boldsymbol{a}, s'\right) = \mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1))$, $\forall t \in \mathbb{N}$; the parameter $\gamma \in [0,1]$ is the discount factor, which is used in the valuation of rewards. Under the summation of the rewards of all the participating agents, stochastic games can be classified into general-sum setting and zero-sum setting, where zero-sum games mean that the gain of one agent equates to the loss of other agents [46], which is not the case in general-sum games. If all the agents can share and optimize the same objective function, even though the individual reward functions of different agents can be different, we will get identical-interest games. Further, if the mutual interests of different agents are subject to a shared potential function, identical-interest games will become potential games that will degenerate into team games if the reward functions are chosen to be a potential function [26]. In the realm of zero-sum games, harmonic games can be viewed as a general subclass that has a harmonic property, such as rock–paper–scissors [26].

2.1.2. Partially observable stochastic games

The MDP model is established on the ideal assumption that an agent can fully observe the state of the stationary environment. If the assumption is eliminated, we can get a partially observable Markov decision process (POMDP), which is a generalized form of an MDP. A POMDP incorporates unobservable states in an MDP, which enables an agent to access only the probabilistic information about the unobservable states.

POMDPs. The model of a POMDP is defined to be a tuple $\langle \mathbb{S}, \mathbb{A}, \mathbb{O}, \mathcal{P}, \mathcal{O}, \mathcal{R}, \gamma \rangle$, where $\mathbb{O}$ denotes the observation space; $\mathcal{O} : \mathbb{S} \times \mathbb{A} \times \mathbb{O} \rightarrow [0,1]$ is used to express the probability of observing $o' \in \mathbb{O}$ for the state $s' \in \mathbb{S}$ after the action $a \in \mathbb{A}$, i.e., $\mathcal{O}\left(o'|s',a\right) \triangleq \text{Prob}\left(o(t+1) = o'|s(t+1) = s', a(t) = a\right)$, $\forall t \in \mathbb{N}$ [47], or alternatively, the function of observation probability is considered to be $\mathcal{O} : \mathbb{S} \times \mathbb{O} \rightarrow [0,1]$, which is expressed as $\mathcal{O}(o|s) = \text{Prob}(o(t) = o|s(t) = s)$ [45]; the other symbols are the same as in the definition of MDPs. Actually, a POMDP is characterized by two stochastic processes: one is a hidden core process $\{s(t)\}_{t \in \mathbb{N}}$, and the other is an observation process $\{o(t)\}_{t \in \mathbb{N}}$. The hidden process tracks the dynamic of the state and is considered to be a Markov process with a finite number of states; the observation process is the sequence of observations the agent obtains. These two processes are connected by the probabilistic function $\mathcal{O}$ that determines the likelihood of observing a specific value of $o(t)$ with the concurrent state $s(t)$, $\forall t \in \mathbb{N}$. The choices of $\mathcal{O}\left(o'|s',a\right)$ and $\mathcal{O}(o|s)$ are determined by the model of sensors. In this setting, the policy chooses the actions based on all the historical observations. In view of this, POMDPs are more general and effective for modeling a wider scope of problems than traditional MDPs in the RL framework.

Decentralized POMDPs. As for the scenario involving multiple agents, a POMDP can be extended to a decentralized POMDP (Dec-POMDP) that is characterized by a tuple $\langle \mathbb{I}, \mathbb{S}, \overline{\mathbb{A}}, \mathcal{P}, \mathcal{R}, \overline{\mathbb{O}}, \mathcal{O}, \gamma \rangle$, where $\overline{\mathbb{O}} \triangleq \mathbb{O}_1 \times \mathbb{O}_2 \times \cdots \times \mathbb{O}_N$ denotes the joint observation space with $\mathbb{O}_i$ being the observation space specific to the ith agent; $\mathcal{O} : \mathbb{S} \times \overline{\mathbb{A}} \times \overline{\mathbb{O}} \rightarrow [0,1]$ denotes the observation function defined by $\mathcal{O}\left(\boldsymbol{o}|s',\boldsymbol{a}\right) \triangleq \text{Prob}\left(\boldsymbol{o}(t+1) = \boldsymbol{o}|s(t+1) = s', \boldsymbol{a}(t) = \boldsymbol{a}\right)$, $\forall t \in \mathbb{N}$, where $\boldsymbol{o} \triangleq \left(o_1, o_2, \dots, o_N\right) \in \overline{\mathbb{O}}$; the other symbols can be referred to those in the definition of stochastic games. At each time step, the ith agent receives an observation $o_i \in \mathbb{O}_i$ generated according to the joint observation probability $\mathcal{O}\left(\boldsymbol{o}|s',\boldsymbol{a}\right)$, $s' \in \mathbb{S}$, $\boldsymbol{a} \in \overline{\mathbb{A}}$. Here, we need to note that the observation of an agent can either contain or exclude the actions of other agents [48]. For each agent i, we denote its historical observation at time t as $\vec{\boldsymbol{o}}_i(t) \triangleq \left(o_i(1), o_i(2), \dots, o_i(t)\right)$, where

$$\vec{\boldsymbol{o}}_i(t) \in \underbrace{\mathbb{O}_i \times \mathbb{O}_i \times \cdots \times \mathbb{O}_i}_{t}.$$

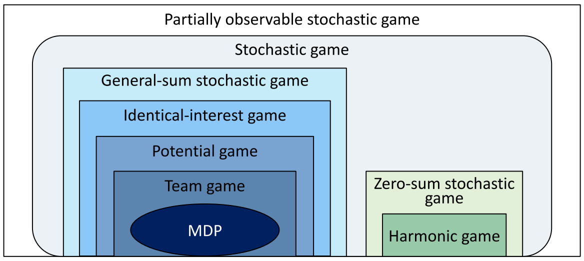


Fig. 1. Relationship among different scopes of stochastic games.

The ith agent leverages its historical observations to choose its actions. At each time step, the agent team earns a joint reward $r(t) \triangleq \mathcal{R}(s(t), \boldsymbol{a}(t), s(t+1))$, which is dependent on the joint action and the current state. The aim is to maximize the expected reward over the horizon T. Similar to the framework of stochastic games, the goal is to find an optimal joint policy that can maximize the expected reward.

Partially observable stochastic games. If we consider that the ith agent earns a reward formulated as $r_i(t) \triangleq \mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1))$ instead of $r(t) \triangleq \mathcal{R}(s(t), \boldsymbol{a}(t), s(t+1))$ in a Dec-POMDP, then the Dec-POMDP will be extended to be a partially observable stochastic game. This sequence of steps continues for either a set number of time steps or until the game reaches a certain terminal state. Partially observable stochastic games can also be defined analogously with continuous-valued (or mixed discrete–continuous) observations. Since partially observable stochastic games with common rewards, in which all agents have the same reward function, will reduce to Dec-POMDPs, which can be regarded as a special class of partially observable stochastic games. Moreover, partially observable stochastic games encompass stochastic games as a special case where the observation for each agent $o_i(t)$ at time t is in analogy with the pair of the current state and the previous action $(s(t), \boldsymbol{a}(t-1))$, $\forall t \in \mathbb{N}\backslash\{0\}$ [49]. Partially observable stochastic games also cover POMDPs as a special case when the set $\mathbb{I}$ contains only one agent. Thus, the observation functions in partially observable stochastic games are more general and can represent a wider range of decision processes than in Dec-POMDPs, conventional stochastic games, and POMDPs. By referring to [26], the relationship among these different stochastic games is summarized in Fig. 1.

Analogous to the evolution from MDPs to POMDPs, stochastic games can be generalized to decentralized POMDP (Dec-POMDP) and partially observable stochastic games if the agents are unable to access all the exact states of the environment; instead, they can only obtain an observation of the environment states [50]. The partially observable stochastic games are also known as partially observable Markov games. While a Dec-POMDP focuses on a collective reward function, agents in a partially observable stochastic game aim to maximize their own distinct reward functions. A partially observable stochastic game inherently incorporates a model that represents a distribution across the belief states of other agents [50].

In addressing stochastic games within MARL, several strategies have been developed, such as Q-learning [49,51], actor–critic [52,53], and policy gradient techniques [54–56]. These methodologies are designed to facilitate learning from agent interactions and environmental factors, thereby optimizing strategies towards achieving set objectives.

2.2. MARL methods

To solve an MARL problem is actually to learn policies for all the agents. Here, we denote $a_i \in \mathbb{A}_i$ as the action that the ith agent takes, $a_{-i} \in \mathbb{A}_1 \times \cdots \times \mathbb{A}_{i-1} \times \mathbb{A}_{i+1} \times \cdots \times \mathbb{A}_N$ as the joint actions of all the agents except the ith agent, π_i as the local policy of the ith agent, and π_{-i} as a compact representation of the joint policy of all the complementary agents of the ith agent. The state-value function $V_i(s)$ for MARL can be formulated as

$$V_i(s) \triangleq \mathcal{E}_{\pi_i,\pi_{-i}} \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1)) \middle| s(0) = s \right],$$

where we write π_i and π_{-i} separately to distinguish the policy between the ith agent and the other agents. Therein, the joint action $\boldsymbol{a}$ can be represented by $\left(a_i, a_{-i}\right)$. In this process, the sequence of the state–action of the ith agent is generated by $s(t+1) \sim \mathcal{P}(\cdot|s(t), \boldsymbol{a}(t)), a_i(t) \sim \pi_i(\cdot|s(t))$ and $a_{-i}(t) \sim \pi_{-i}(\cdot|s(t)), \forall t \in \mathbb{N}$.

Analogous to the state-value function, we can define a state–action Q-function $Q_i(s, \boldsymbol{a})$ below:

$$Q_i(s, \boldsymbol{a}) \triangleq \mathcal{E}_{\pi_i,\pi_{-i}} \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1)) \middle| \boldsymbol{a}(0) = \boldsymbol{a}, s(0) = s \right],$$

which is the expected reward by selecting the initial joint action $\boldsymbol{a}$ in state s. Then, the ith agent follows its policy π_i to select subsequent actions, while the other agents are subject to the policy π_{-i} from the perspective of the ith agent. To learn the policies for all the agents, related techniques in MARL include the value-based method [57–59] and the policy-based method [60,61].

2.2.1. Value-based MARL

One most popular value-based method is Q-learning that employs Q-functions to approximate the optimal values of these Q-functions. The Q-functions update their values by means of temporal difference learning. For each agent $i \in \mathbb{I}$ in a value-based MARL process, given the transition data $\left\{\left(s(t), \boldsymbol{a}(t), \mathcal{R}_i(t), s(t+1)\right)\right\}_{t\in\mathbb{N}}$ sampled from the replay buffer, the agent merely updates the value of $Q_i(s(t), \boldsymbol{a}(t))$ but does not alter the other entries of the Q-function at the t th iteration. To be more specific, this process can be formulated as

$$\begin{aligned} Q_i(s(t), \boldsymbol{a}(t)) \leftarrow & (1-\alpha)\, Q_i(s(t), \boldsymbol{a}(t)) \\ & + \alpha \left(\mathcal{R}_i(t) + \gamma \max_{a_l \in \mathbb{A}_l, l \in \mathbb{I}} \left\{ Q_l(s(t), \boldsymbol{a}(t)) \right\}_{l\in\mathbb{I}} \right), \end{aligned}$$

where α is the learning rate and $\mathcal{R}_i(t)$ is the abbreviation of $\mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1))$. As indicated by the set of their Q-functions, each agent has not only to consider itself but also to take the interests of all the other agents into account when the agent is assessing the stage game at time step $t+1$. Then, the optimal policy can be worked out. The method returns the part of the optimal policy corresponding to the ith agent at some equilibrium point, which does not always align with the maximum possible reward for that agent. Based on the assumption that all other agents agree to play at the same equilibrium, the value-based MARL method can then calculate the expected long-term reward of the ith agent under this equilibrium.

2.2.2. Policy-based MARL

Since the state of the environment is affected by the actions of all agents in the multi-agent scenario, it will be tricky to leverage the value-based method. On the other hand, as the number of agents grows, the value-based method faces the challenge of the curse of dimensionality, which is a result of the combinatorial nature inherent in multi-agent systems. These drawbacks necessitate the exploration of policy-based algorithms incorporating function approximations, which can alleviate this problem to some extent. To be specific, each agent learns its own optimal policy $\pi_i^{(\theta_i)}$ via updating the parameter θ_i of a neural network. We denote the collective policy parameter for all agents as $\boldsymbol{\theta} \triangleq \left(\theta_1, \theta_2, \ldots, \theta_N\right)$, and further express the parameterized joint policy by $\pi^{(\boldsymbol{\theta})} \triangleq \prod_{i\in\{1,2,\ldots,N\}} \pi_i^{(\theta_i)}\left(a_i|s\right)$. To optimize each parameter θ_i, $\forall i \in \mathbb{I}$, the policy gradient theorem has been extended to the multi-agent setting. Given the objective function for the ith agent: $J_i^{(\boldsymbol{\theta})} \triangleq \mathcal{E}_{s\sim\mathcal{P}, \boldsymbol{a}\sim\pi^{(\boldsymbol{\theta})}} \left[\sum_{t=0}^{\infty} \gamma^t \mathcal{R}_i(s(t), \boldsymbol{a}(t), s(t+1))\right]$, the gradient of the objective function with respect to the parameter θ_i can be deduced as

$$\nabla_{\theta_i} J_i^{(\boldsymbol{\theta})} = \mathcal{E}_{s\sim\rho(\pi^{(\boldsymbol{\theta})}), \boldsymbol{a}\sim\pi^{(\boldsymbol{\theta})}} \left[\nabla_{\theta_i} \log \pi_i^{(\theta_i)}\left(a_i|s\right) \cdot Q_i(s, \boldsymbol{a})\right],$$

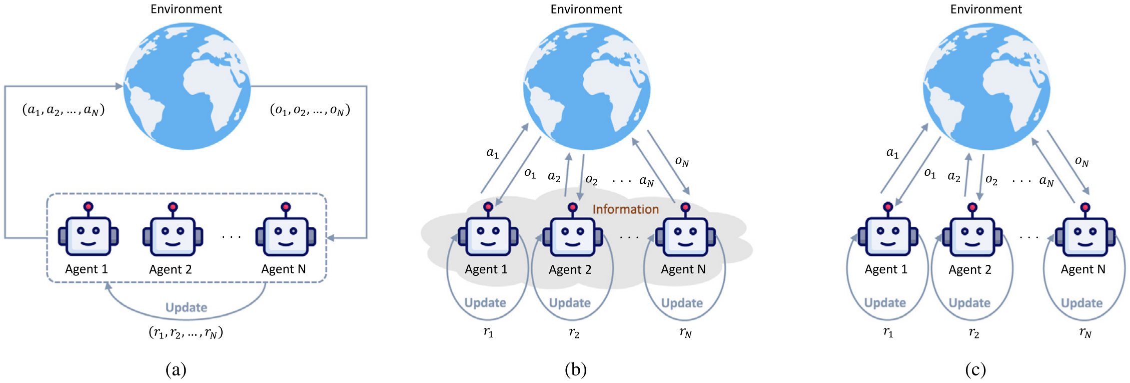


Fig. 2. Paradigms of training and execution in MARL. (a) CTCE, in which there is a unified policy applicable to all agents. (b) CTDE, which allows agents to share additional information in the training phase, but the agents abandon the information in the subsequent execution phase. (c) DTDE, where each agent updates its personal policy independently.

where $\rho(\pi^{(\theta)})$ denotes the state occupancy measure under the joint policy $\pi^{(\theta)}$; $\nabla_{\theta_i} \log \pi^{(\theta_i)}\left(a_i|s\right)$ is the updating score of the local policy of the ith agent. If the policy is deterministic while the action set is continuous, one can derive the deterministic policy gradient (DPG) algorithm [60], which can be expressed as

$$\nabla_{\theta_i} J_i^{(\theta)} = \mathcal{E}_{s\sim\rho(\pi^{(\theta)})}\left[\nabla_{\theta_i}\log\pi_i^{(\theta_i)}(a_i|s)\cdot\nabla_{a_i}Q_i(s,\boldsymbol{a})\Big|\,\boldsymbol{a}=\mu^{(\theta)}(s)\right],$$

where $\mu^{(\theta)} : \mathbb{S} \to \overline{\mathbb{A}}$ signifies the deterministic joint policy. It should be noted that the expectation over the joint policy $\pi^{(\theta)}$ necessitates a strong assumption that the policies of the other agents should be observable. Moreover, the error of the gradient estimates within the critic network will be larger as the number of agents grows.

2.3. Cooperative, competitive, and mixed tasks

The purposes or interests of controlled agents are expected to be achieved by establishing a reward configuration. Stemming from varied reward structures and learning aims, the behaviors of agents can be broadly classified into three distinct categories: cooperative, competitive, and a mix of both, termed as a cooperative–competitive setting. In a cooperative setting, MARL aims to maximize the collective reward of all agents. Generally speaking, cooperative tasks are those where agents are incentivized to work collaboratively, even if the rewards are not always equally distributed. However, in a fully cooperative task, all the agents get the same reward, that is, $r_1 = r_2 = \cdots = r_N$. The mechanism of equal reward distribution can encourage agents to collaborate and avoid the failure of any individual agents, such that the overall team performance can be maximized. As for the competitive setting, it focuses on maximizing the individual reward of each agent. In general, competitive games are those where agents strive to outperform each other, though the total reward may not always sum to zero. However, in a fully competitive task, the case can be characterized by a zero-sum game. In this setting, the total reward across all agents equals zero, i.e., $\sum_{i=1}^{N} r_i = 0$. Here, each agent aims to maximize their individual reward while simultaneously minimizing the rewards of others. The mixed setting, also known as a general-sum game, is neither entirely cooperative nor entirely competitive [30]. In this case, MARL seeks to strike a balance between cooperative and competitive elements [31]. This type of tasks allows for more flexibility, as it does not impose specific constraints on the goals or objectives of the agents.

2.4. Centralized and decentralized training

The dichotomy of centralized and decentralized training paradigms has long been adopted in the MARL framework [62]. In a centralized training paradigm for agents, policy updates are performed based on the information exchange during the training phase, and this additional information is generally abandoned during the testing phase. Alternatively, in a distributed training paradigm, each agent updates its policy independently and does not exchange information with other agents. After training, the agents have to choose actions to take in the execution phase. The paradigm of execution can also be categorized into the centralized type and the decentralized type. Centralized execution means that the joint actions of all the agents are worked out and taken, while in decentralized execution, each agent decides actions independently based on its own policy.

Combining the training and execution phases, three common paradigms of training and execution can be obtained, that is, centralized training with centralized execution (CTCE), centralized training with decentralized execution (CTDE), and decentralized training with decentralized execution (DTDE), which are schematized in Fig. 2. The CTCE scheme is about a centralized agent executing a joint policy π that maps a set of distributed observations to various distributions over the individual actions of each agent. The CTCE is a relatively simple paradigm in cooperative MARL, which converts a multi-agent problem into a single-agent one by considering the joint state/action space of all agents as that of a single virtual agent. CTDE is a popular alternative paradigm, where agents are trained using centralized information but execute their actions independently based on their local observations [63]. Such a paradigm can address the issue of partial observability while circumventing the extensive input and output dimensions typically emerging in centralized execution. In DTDE, agents are trained independently to optimize team rewards, and each agent regards other agents as a part of the environment. This paradigm is suitable for self-interested tasks and mixed tasks requiring a balance of cooperation and competition, which can handle the scalability problem caused by the growth of the agent number [28]. The choice between centralized and decentralized training in MARL depends on the specific requirements of the task and the number of involved agents, since each paradigm offers distinct benefits and challenges.

3. Challenges in MARL applications

When applying MARL to real-world problems, researchers and practitioners may inevitably encounter a series of formidable challenges that critically impact the development and efficacy of MARL, including scalability, non-stationarity, partial observability, credit assignment, continuous action spaces, communication challenges, and learning complexity. An overview of these challenges is presented in Fig. 3. These challenges often coexist in practice. Some challenges can lead to or exacerbate other problems. For example, non-stationarity can deteriorate scalability; partial observability can make both the non-stationarity and credit assignment problems severer; and all the first six challenging issues can add additional learning complexity.

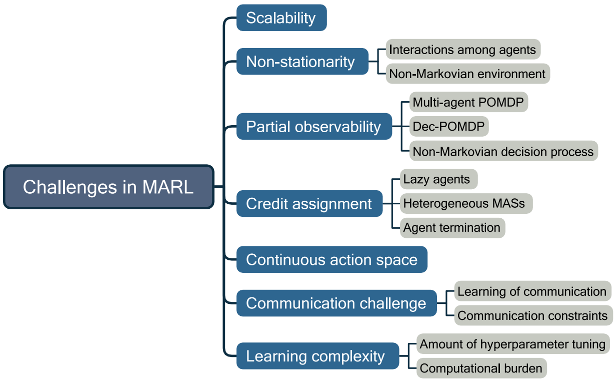


Fig. 3. An overview of challenges in MARL.

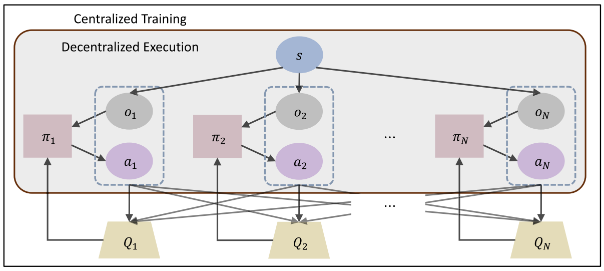


Fig. 4. A schematic of the architecture of CTDE-based MADDPG.

3.1. Scalability

To address the scalability challenge in MARL, powerful techniques are required due to the exponential increase in complexity with the addition of each agent. Knowledge reuse, particularly through parameter sharing and transfer learning, is essential to overcoming this challenge. Information sharing, which is effective in scenarios where agents share structural similarities, can promote a more efficient training process that can scale with the number of agents. This standpoint has been demonstrated by a variety of applications [58,63–68]. Transfer learning can accelerate the learning process via enabling agents to apply knowledge from earlier tasks to new and related activities [36,37,69,70]. Techniques like the mean-field approach have shown significant potential in managing scalability [71]. Curriculum learning emerges as a vital strategy for managing the increased difficulty in training multiple agents and has shown to be capable of enhancing policy generalization and hastening convergence [64,72]. The robustness of learned policies against environmental perturbations is also paramount in developing scalable MARL. Regularization techniques, such as policy ensembles and adversarial training, foster the development of resilient policies against perturbations [73]. Additionally, self-play has been instrumental in achieving robust policies by continuously challenging the agents with evolving strategies, especially in imperfect-information games [74–77]. In large-scale and highly interactive settings, the dynamic nature of multi-agent environments can still arouse other challenges. DTDE can address the scalability problem but may result in non-stationarity in the environment. Fortunately, one way to implement this DTDE is the application of independent deep Q-networks (DQNs) to the multi-agent setting [59,78].

3.2. Non-stationarity

Environmental non-stationarity in MARL is a critical issue that stems from the policy changes and dynamic interactions among multiple agents that share a common environment during simultaneous learning. This may disrupt the Markov assumption that is foundational in traditional RL because the environment, with which an agent interacts, is not solely determined by the actions of this agent but also by the collective actions of all the agents [14,79]. The changing environment that the ith agent faces can be formulated as $\mathcal{P}\left(s'|s,a_i,\pi_1,\ldots,\pi_N\right) \neq \mathcal{P}\left(s'|s,a_i,\pi_1',\ldots,\pi_N'\right)$ due to $\pi_i \neq \pi_i'$, $\forall i \in \mathbb{I}$. In view of this, the assumption of Markov games cannot be satisfied in a non-stationary environment [38]. Early approaches to addressing non-stationarity often ignore other agents or assume agent behaviors to be static [33,80]. While effective in simple environments, this approach is limited in complex or stochastic setting [81,82]. Addressing the non-stationarity has led to the development of various techniques. [83] introduced a technique called lenient-DQN to address the non-stationarity problem, in which policy was updated by the samplings from experience replay memory. More representative works can be found in [60,84,85], where centralized critics were used to augment the capabilities of agents. Another effective strategy is the adaptation of the experience replay mechanism. However, this approach is challenged by the evolving policies of agents in MARL, rendering past experiences less relevant. To tackle this issue, the methods of decaying obsolete data and conditioning the value function of each agent on a fingerprint were proposed in [78]. Further, [83] extended the use of experience replay mechanisms, which can better handle the non-stationarity phenomenon in MARL. Meta-learning emerges as an alternative approach, where agents learn to adapt to the evolving behaviors of others. [86,87] provided exemplary applications of meta-learning in MARL to construct predictive models of the actions of other agents. Combining centralized training, experience replay modification, and meta-learning, MARL can become more adaptable and robust to the non-stationarity problem.

3.3. Partial observability

In implementations of MARL, each agent can only observe a limited amount of information about the environment, which affects the ability of the agent to assess the quality of the actions of surrounding agents. Hence, agents have to operate with limited access to the global state. As mentioned before, one class of decision processes resulting from partial observability in MARL is Dec-POMDPs, which can enhance the complexity of investigated problems.

The paradigm of CTDE can address this challenge effectively, as indicated by [88]. To enhance the training process by incorporating the policies with the information from other agents, [60] implemented this CTDE paradigm, where the actor–critic framework was extended by adding additional information about the policies of other agents. The method is called CTDE-based MADDPG, and its architecture is presented in Fig. 4. Value decomposition networks (VDN) proposed by [58] and QMIX by [89] are effective solutions. They focus on decomposing the collective goal of a team into individual components, ensuring an active and efficient participation of each agent. Incorporating the memory mechanism in agents is also a means of countering partial observability in MARL, such as deep recurrent Q-networks (DRQNs) [90]. The use of DRQNs to equip agents with memory capability is further investigated by [64]. The incorporated memory mechanism was further refined in [91], which equipped agents with the ability to make more informed decisions based on historical action-observations. [92] introduced a Bayesian method to enhance the capability of agents in cooperative settings under partial observability. By introducing an attention mechanism to establish dynamic communication under partial observations, incorporating communication capability into agents was explored in [65], where agents are allowed to communicate on demand rather than at fixed intervals.

A notable advancement in handling partial observability in the application of MARL can be observed in the application of video games, which often feature environments with unobserved information, see, e.g., [75,76]. Further, [93,94] showcased the application of MARL to complex, non-Markovian environments typical in competitive video games. The intersection of the scalability and non-stationarity challenges in MARL is closely tied to the issue of partial observability.

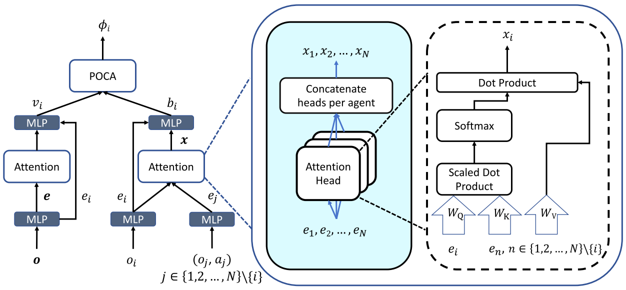


Fig. 5. Structure of the critic in MA-POCA algorithm, where o_i and a_i denote the state observation and the action of the ith agent, respectively, with $i \in \{1, 2, \dots, N\}$ signifying the index of the N agents; e_i is an embedding variable mapped from o_i; x_i, v_i, and b_i denote an intermediate variable outputted by self-attention, an output of the centralized state value function, and a baseline, respectively. $\boldsymbol{o}$, $\boldsymbol{e}$, and $\boldsymbol{x}$ are the vectors concatenated from o_i, e_i, and $x_i, i \in \{1, 2, \dots, N\}$, respectively.

3.4. Credit assignment

In MARL, the effectiveness of agent learning and cooperation is heavily contingent on the successful assignment of credit for collective actions. Due to the interactions of individual and collective agent dynamics in MASs, it becomes difficult to determine the effect of each agent action on the whole team success [30,95]. The complexity therein is particularly significant in mixed incentives or social dilemmas, which can lead to issues, such as the lazy agent problem [58]. Credit assignment is a notable challenge in MARL when tackling fully cooperative tasks. To address this problem, the method of VDNs was proposed by [58], which is a representative decomposition strategy that factorizes the joint action-value function into linear combinations of individual action-value functions. Subsequent advancements, including QMIX [89] and QTRAN [57], have refined this strategy by allowing for more sophisticated and nonlinear combinations of value functions. In conjunction with decomposition, marginalization strategies were developed, e.g., [91,96], by marginalizing the actions of individual agents to reduce the variance in gradient estimations. Imitation MARL [97] and inverse MARL [98] have emerged as two alternative strategies to tackling credit assignment, which have shown potential in learning high-dimensional policies and addressing the implicit differences for multi-agent interactions.

A recent study [99] focused on factorizing and reshaping the team reward into individual rewards, which can address the problem of continuous action space and agent heterogeneity. In [95], the authors proposed a hierarchical MARL method called MLCA, which can efficiently utilize different hierarchical information to reason and achieve credit assignment across multiple hierarchies. Additionally, the multi-agent polarization policy gradient (MAPPG) method, developed by [100], exemplified the ongoing efforts to address the centralized–decentralized mismatch to achieve a nice credit assignment in MARL. In practice, premature termination of an agent can raise the so-called posthumous credit assignment problem due to the propagation of values from the rewards earned by the remaining agents to the terminated agents. Traditional MARL settled this problem by setting the terminated agents in absorbing states, which can lead to inefficiencies in training and resource utilization. To address this critical issue, [101] proposed multi-agent posthumous credit assignment (MA-POCA). Instead of a fully connected layer, MA-POCA adopts a self-attention mechanism for active agents in the critic network, as shown in Fig. 5.

3.5. Continuous action space

Traditional DRL methods are generally confined to discrete action spaces. For example, DQNs have been leveraged to handle high-dimensional observation spaces but are inherently limited to discrete and low-dimensional action spaces [4]. In the scenario of continuous action spaces, both the curse of dimensionality and the necessity for iterative optimization at each step pose significant challenges. One of the pivotal advancements in this area was the trust region policy optimization (TRPO) method proposed by [102]. TRPO is adaptable to both continuous states and actions. Based on this foundation, the PS-TRPO method was introduced for MARL by [64]. Additionally, [103] made a significant contribution by developing a fully decentralized actor–critic algorithm for collaborative MARL, which provided convergence guarantees with linear function approximation. Recent contributions focused on enhancing the strategic depth and adaptability of MARL algorithms in continuous action spaces. [104] introduced a probabilistic recursive reasoning (PR2) framework, which posited the benefit of agents by considering the reactions of opponents to their future actions and led to the development of the algorithms PR2-Q and PR2-Actor–Critic. To address the challenges of both continuous action space and non-stationarity in MARL simultaneously, [105] proposed the time dynamical opponent model (TDOM). This model leverages the progressive improvement of opponent policies over time for better policy adaptation. Their multi-agent actor–critic with time dynamical opponent model (TDOM-AC) outperforms the existing actor–critic methods, especially in mixed cooperative–competitive environments.

3.6. Communication challenge

Agent communication plays an important role in guaranteeing system performance and adaptability for MARL. The exploration is twofold: one is to encompass the challenges and innovations in learning communication among agents; the other is about communication constraints.

3.6.1. Learning of communication

Developing multi-agents that can build communication languages is a significant challenge. In fact, learning a communication policy and message content is one aspect of communication learning, which focuses on updating and modifying a communication protocol [106]. The exploration of communication learning in MARL encompasses how agents develop effective communication protocols and languages. The means of communication can be classified as broadcasting communication, targeted communication, and networked communication [30], as illustrated in Fig. 6. In fully cooperative settings, agents have to learn to cooperate and communicate efficiently. Methods, such as RIAL and DIAL [63], CommNet [67], and BiCNet [66], explored various aspects of broadcasting communication, from discrete protocols to continuous ones and even coordinating heterogeneous agents. Some attentional architectures proposed in recent few years, such as ATOC [65], VAIN [107] and TarMAC [108], demonstrated targeted communication through an attention mechanism, which dynamically selected communication recipients and refined the process. In mixed cooperative–competitive settings, IC3Net [109] expanded upon these ideas by introducing individualized communication control, while TBONE [110] used a two-stage communication strategy, in which an initial exchange of messages followed by responses to the messages received in the first stage. As a common form of multi-agent communication, networked communication focuses on localized exchanges within agent neighborhoods. This type of communication is crucial in decentralized systems, and some related studies in the MARL framework can be found in [111,112].

3.6.2. Communication constraints

Addressing communication constraints, such as limited bandwidth, noisy channels, and shared mediums, is crucial for a normal function of MASs in real-world scenarios. The challenge of limited bandwidth in MARL has been approached by various innovative techniques aimed at optimizing communication efficiency. SchedNet, proposed by [113], addresses the issue by prioritizing messages from agents that are deemed more important in scenarios with a shared communication

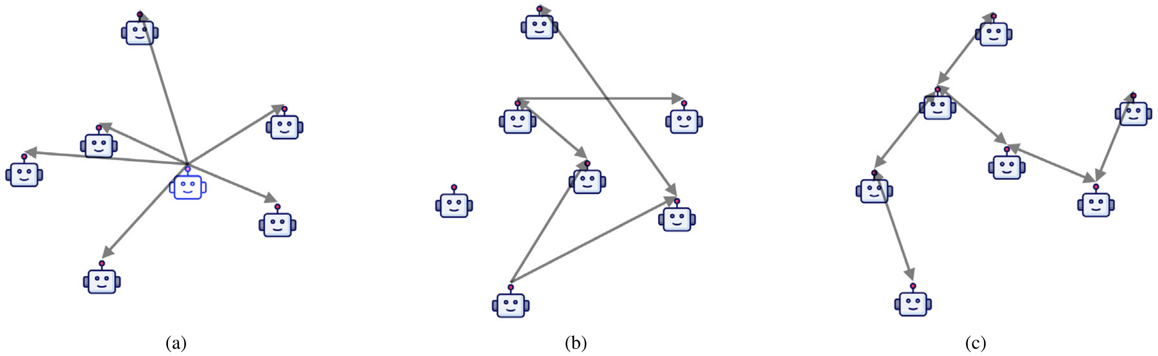


Fig. 6. A schematic of three different communication means among agents. Arrows mean the direction of transmitting messages. (a) Broadcasting communication, where messages from the activated agent are transmitted to all the other agents within the communication network. (b) Targeted communication, where agents can select the target agents according to a supervisory mechanism that determines the time, content, and recipients for the communication. (c) Networked communication, which refers to the localized interactions with neighboring agents in the network.

channel. This approach balances the need for communication with the constraints of limited bandwidth. Variance-based control (VBC) [114] and temporal message control (TMC) [115] further developed this concept by implementing pre-defined thresholds to filter out non-essential communication, thereby reducing overhead. Gated-ACML [116] presented a more nuanced approach by incorporating a probabilistic gate unit that regulated message transmission between agents and a centralized message coordinator, which can offer a more flexible and efficient way of managing bandwidth limitations. Informative multi-agent communication (IMAC) method [17] took a unique approach by explicitly modeling bandwidth limitation within its optimization process. It requires agents to send low-entropy messages, thus directly addressing the bandwidth constraints. Event-triggered communication network (ETCNet) [117] built on these ideas and established a constrained model that transforms bandwidth into a penalty threshold, thus restricting sending behaviors. Variable-length coding [118] also employed a penalty term but with a focus on encouraging shorter messages. Communication in noisy environments is another significant challenge in MARL, which was tackled by [118] via backpropagating derivatives through discretized real-value messages. The shared medium category deals with the contention issue that arises when messages are transmitted through a single medium. Memory-driven MADDPG (MD-MADDPG) [119] presented a solution by enabling agents to access a shared memory space sequentially, which can thereby avoid conflicts.

3.7. Learning complexity

3.7.1. Hyperparameter tuning

Hyperparameter tuning is a critical and intricate phase in MARL associated with deep neural networks, which can significantly affect the performance and efficiency of learning algorithms. [120] highlighted the importance of effective tuning via demonstrating how long short-term memory networks (LSTMs) can outperform recent models when finely adjusted. [121] further illuminated the broader impact of hyperparameter tuning in machine learning and advocated for a deeper comprehension of the tuning process. They argued that the understanding and refinement of the existing models could be more beneficial than developing new ones. This view was reinforced by [122], which explored the varied effects of hyperparameters across different algorithms and environments and revealed how critical the choice and tuning of these parameters are in determining MARL outcomes. However, the process is quite challenging. [123] exposed cases where the apparent success of certain models was mistakenly attributed to the used methods instead of underlying bugs and errors, which highlighted the delicate balance in tuning for optimal MARL performance. [124] also noted the inherent training difficulties in some neural network models, such as catastrophic forgetting, which added a layer of complexity to the tuning process.

3.7.2. Computational burden

While hyperparameter tuning addresses the nuances of optimizing learning algorithms for improved performance, computational burden confronts the resource-intensive nature of these algorithms. In MARL, a primary concern is the low sample efficiency, as highlighted by [125]. This inefficiency is exemplified in training agents for simple games, such as Pong, where tens of thousands of samples are necessary for effective learning [126]. The computational demands in MADRL can be quite staggering, as reported by [59,93]. Recent developments by [127] showed potential for reducing inference and training time. However, there exist exceptions where computational infrastructure remains a major bottleneck, particularly for smaller entities [128]. To tackle these challenges, two key strategies were proposed. First, there is a need for enhanced awareness and a detailed report of computational traits and demands in the research of MADRL [129]. Second, focusing on algorithmic innovations rather than computational power in MADRL research can yield more sustainable advancements. Despite these efforts, questions still remain on how to best measure and report computational efficiency.

4. Applications of MARL

MARL has found its practical implementations in various engineering applications, science research, and human society. By enabling multiple agents to learn and interact with each other in a shared environment, researchers can develop technologies that can help address some of the most pressing challenges.

4.1. Application in intelligent machines and devices

We provide a comprehensive overview of MARL in intelligent machines and devices, demonstrating the versatility and efficacy of MARL in complicated real-world settings. Table 1 lists the specific applications and related studies leveraging MARL.

4.1.1. Autonomous vehicles

In UAVs, MARL has significantly enhanced operational efficiency. [15] explored the use of MARL in optimizing field coverage by UAVs, achieving optimal sensory coverage with minimal overlap. [130] proposed an MARL framework for multi-UAV target assignment and path planning (MUTAPP), showcasing effective strategies for maneuvering and collision avoidance. [131] developed an agent-independent technique in the framework of MARL, wherein all the agents shared a common structure but executed a decision-making algorithm independently. Recently, [22] focused on leveraging MARL for enhancing UAV swarm communications against jamming by employing policy distribution to refine policy exploration and integrating transfer learning to accelerate learning efficiency. This method can effectively reduce

Table 1
Typical applications of MARL in different areas of intelligent machines and devices.

Scopes	Applications and studies
Autonomous vehicles	UAVs: [15,22,130,131] Autonomous driving: [132–136]
Traffic scheduling	Alleviation of bus congestion: [137] Traffic signal control: [138–140] Large-scale fleet management: [141] Transportation trajectory simplification: [142]
Energy supply and scheduling	Microgrid energy scheduling: [143,144] Electrical power networks: [145] Energy exchange between buildings: [146,147] Charging: [148–150]
Networks	Communication: [67,96,151] Mobile network allocation: [131,152]
Image processing	Image classification: [153,154] Pattern recognition: [155] Object extraction: [156]
Chips and biochips	Multi-core systems: [157] Biochips: [158,159]

state quantization error in dynamic environments and thereby improve anti-jamming capabilities. In autonomous driving, MARL has facilitated significant progress. [132] introduced a safe RL algorithm for autonomous vehicles, integrating driving behaviors with rigorous safety measures. This method separates the policy function into a component for driving preferences and another for safety-conscious trajectory planning. [133] probed the adaptation of multi-agent autonomous driving policies to real-world conditions, notably reducing the gap between simulation and reality. [134] implemented centralized training in MARL for autonomous driving, which led to quicker learning and higher rewards in complex simulations like highway scenarios. [135] advanced their work on highway merging using DRL, showing notable improvements in collision avoidance. Additionally, [136] provided an in-depth analysis of on-ramp merging via a robust testing framework and an MARL simulation for autonomous driving, demonstrating high efficiency in merge situations.

4.1.2. Traffic scheduling

The implementations of traffic control based on MARL have shown remarkable advancements, which leads to significant improvements in urban traffic scheduling. [137] proposed a cooperative MARL framework to alleviate bus congestion on bus lanes in real-time by means of coordination graphs for selecting coordinated holding actions for multiple buses. As for the control problem of traffic signals, [138] introduced a novel IDQN algorithm to address the heterogeneity problem of urban traffic signal control in a multi-agent environment by integrating dueling networks, DDQN, and priority experience replay; [139] developed cooperative MARL models (CMRLM) for intelligent traffic control, which demonstrated the utility of MARL in managing complex traffic scenarios; [140] addressed urban traffic light control in vehicular networks through a novel MARL algorithm named MARDDPG that can improve the coordination among traffic light controllers at multiple intersections. As for the regulation of large-scale fleets, MARL was applied to online ride-sharing platforms in [141], where contextual deep Q-network (DQN) and contextual actor–critic were introduced to strategically allocate vehicles that are conceptualized as individual agents to specific zones within a city. As it is known that trajectory data has been widely used in taxi services, traffic management, mobility analysis, etc., [142] developed an MARL method called MARL4TS for error-bounded online trajectory simplification and then demonstrated its superior effectiveness over existing algorithms through extensive real-world dataset experiments.

4.1.3. Energy supply and scheduling

MARL has been increasingly applied in the energy domain, where MARL can be used to optimize the energy consumption of a group of agents. By learning the optimal energy purchasing and distribution strategies, MARL demonstrates its potential for optimizing complex systems and revolutionizing energy sharing and scheduling. Aiming at enhancing the energy management capabilities of decentralized microgrids, [143] introduced a fuzzy Q-learning approach. [144] leveraged MARL for residential microgrid energy scheduling, where the agents included renewable energy generators and households. In electrical power networks, MARL has been adopted to optimize energy allocation and management. [145] explored RL for cooperating and communicating reactive agents in electrical power grids, where an independent MARL algorithm based on Proximal Policy Optimization (PPO) was proposed to manage energy exchanges in a factory setting with each element of the factory modeled as separate agents. [146] focused on a distributed multi-agent-based protection scheme for transient stability enhancement in power systems, where the improvements in transient stability were validated on a standard IEEE 39-bus New England benchmark system. In the zero-energy community, an MADRL algorithm was developed in [147] to manage energy exchange between buildings with diverse energy production and storage capabilities to further achieve a net-zero annual energy balance. As for the energy management of battery packs, a distinct approach was proposed in an MARL framework for a smart charge–discharge management of lithium battery packs in [148]. Here, each battery cell is treated as an independent agent; the charge and discharge cycles are optimized to improve battery life and ensure safe temperatures. Then, a framework of multi-agent spatio-temporal RL was founded in [149], which recommended public charging stations by considering long-term spatiotemporal factors and outperformed nine baselines in extensive real-world tests. A while ago, [150] developed an MARL approach that integrates MADDPG and LSTM neural networks for enhanced energy scheduling in charging stations, outperforming existing methods in terms of economic gains and user satisfaction. The above-mentioned applications underscore the adaptability of MARL in managing complex energy systems with multiple interacting components and further illustrate the versatility of MARL in scheduling and using energy.

4.1.4. Network communication and allocation

In the past years, MARL has been increasingly witnessed for advancements in the communication and allocation of networks. [151] explored MARL for routing control within an Internet environment, emphasizing the importance of adaptability and learning in dynamic network scenarios. The CommNet model, introduced by [67], is a significant advancement in MARL for communication. This model facilitates communications among agents, which are characterized by deep recurrent Q-networks (DRQN). As for the establishment of communication links among agents during the learning process, which is a crucial aspect when developing MADRL algorithms, [96] defined the communication channel by leveraging human knowledge represented by images. This approach enables DRL agents to communicate with each other by utilizing these shared images. In application to mobile network allocation, [152] explored adaptive learning techniques for dynamic resource allocation and optimization. By adapting the transmission power and quality of service parameters, the network efficiency can be improved. Furthermore, [131] established a novel MARL framework for UAV communications, such that each agent could identify its optimal strategy based on its own observations independently. This setup optimized transmit power and subchannel selection, which demonstrates the MARL capability of maximizing data throughput while minimizing power consumption.

4.1.5. Image processing

MARL finds a range of applications in image processing, from image classification to the optimization of processing parameters. [153] showcased the application of decentralized MARL to image classification on the modified National Institute of Standards and Technology (MNIST) dataset. In this research, multiple agents received partial observations by using an LSTM network to learn policies for classification. A similar approach was employed in another study [154] on the MNIST dataset. [155] conducted a joint top-down active search of multiple objects in interaction. By conceptualizing each detector as an agent, a collaborative MARL algorithm was developed for joint object localization. This method integrates inter-agent communication with an innovative deep Q-learning algorithm, leading to enhanced object detection performance and the discovery of patterns in co-detection. Recently, [156] used the MARL technique to optimize parameter values in image processing tasks, where both robustness and effectiveness were validated by experimental settings.

4.1.6. Chips and biochips

The value of MARL can also be found in chips and biochips. [157] presented an approach, named machine learned machines, for dynamic resource allocation in multi-core systems. The study employed MARL for co-optimizing cores, caches, and networks within a chip, aimed at reducing the energy-delay product with a limited penalty on system throughput and fairness. They demonstrate the efficacy of this approach in organizing on-chip networks with a focus on energy efficiency and computational throughput. [158] proposed an MARL framework to optimize droplet routing in digital microfluidic biochips. This framework addresses the issue of electrode degradation and ensures reliable transportation of droplets. The results have shown to be more effective than the previous methods, especially in large biochip environments. [159] focused on overcoming challenges in microelectrode-dot-array biochips. By formulating droplet transportation as an MARL problem, the research enhanced the reliability of fluidic operations in biochips. This method is particularly useful for executing complex bioassays in parallel, ensuring high throughput and accuracy in biochip operations.

4.2. Applications in chemical engineering, biotechnology, and medical treatment

Recently, the role of MARL in chemical and biological engineering, and health-related realms has gained considerable attention. The applications of MARL in these fields are listed in Table 2, which show promising results and the potential to revolutionize the way we address these real-world tasks.

Table 2
Typical applications of MARL in chemical engineering, biotechnology, and medical treatment.

Scopes	Applications and studies
Chemical engineering	Industrial coal combustion: [160] Wastewater treatment: [161] Textile manufacturing: [162] Crystallization processes: [163] Solid oxide fuel cells: [164] Semi-batch reactors: [165] Multi-loop process control: [166]
Biology and biotechnology	Bio-insects: [167] Animal swarms: [168–170] Ecological systems: [171–173] Production of microbial strains: [174]
Medical science and healthcare	Biomedical text mining: [175] Medical image: processing: [176,177] Drug research: [178] Surgical robots: [179] Disease prediction and treatment: [180,181] Medical sensing networks: [182] Glycemic control: [183] Public health: [184,185]

4.2.1. Chemical engineering

In chemical engineering, MARL has been used to enhance the regulation of process control systems. This can help optimize the performance of chemical processes and the production of chemicals and further improve the efficiency of chemical processes, reduce waste, and minimize the environmental impact of chemical production. The adoption of MARL in broad aspects of chemical engineering demonstrates its adaptability and efficiency in optimizing complex processes. [160] applied an MARL system to the control of an industrial coal combustion process, resulting in notable reductions in nitrogen oxide emissions and air consumption. In recent years, MARL has received much more attention in chemical engineering. [161] employed MARL to optimize wastewater treatment plants, concentrating on sustainable optimization from a life cycle assessment perspective, indicating that LCA-driven strategies effectively balance environmental impacts and costs. [162] investigated MARL in textile manufacturing and presented an innovative framework that converts optimization challenges into stochastic games by leveraging deep Q-networks. This approach outperforms traditional methods, particularly in the textile ozonation process. [163] explored model-based RL control strategies in crystallization processes, revealing that MARL can reduce training costs and that RL-based adaptive PID control shows advantages over traditional methods. [164] introduced a data-driven controller for solid oxide fuel cells via MARL to achieve enhanced robustness and efficiency in managing fuel flux and utilization. [165] gave an RL-based solution for the optimal control of semi-batch reactors. The efficacy of the designed RL controllers was verified in maintaining specific temperature setpoints. Further, [166] developed a novel MARL approach for multi-loop process control, which showcased its effectiveness in managing systems with strong interactions as well as its performance under different game-theoretic strategies.

4.2.2. Biology and biotechnology

With respect to the area of biological science and engineering, MARL still demonstrates its applicability and becomes a competitive research confront. Research in this domain has already begun more than ten years before. For example, [167] studied bio-insect and robot interaction utilizing MARL, which signified an innovative step into guiding bio-insects via artificial intelligence robots and the advancements in interspecific interaction and control systems. In recent years, the combination of biology with MARL has become much more attractive. [168] focused on evolving swarm communication among ants by means of applying neuroevolution in MARL to adjust the topology and weights of neural networks, thus improving the efficiency of foraging behaviors. [169] examined fish schooling using MARL by regarding each fish as an autonomous learning agent and leveraging mean field Q-learning. The study successfully replicated natural collective motion patterns and further shed light on the underlying mechanisms of group dynamics in nature. [170] explored the application of swarm inverse RL in biological systems, focusing on deciphering the collective behavior of animal swarms. This method uniquely combines parameter sharing with deep inverse RL, resulting in significant implications for biological behavioral studies. As for the field of ecosystems, related new studies have been reported in recent years. [171] applied MARL to autonomous marine environmental monitoring with robotic swarms by focusing on addressing the non-stationarity of environmental feature and introducing advanced methods for dynamic area coverage in order to outperform traditional swarming techniques. [172] leveraged MARL to simulate large-scale predator–prey ecosystems. By incorporating a mating mechanism and a real-time evolutionary algorithm, the model therein can reflect the adaptive behaviors of agents, which offers new insights into ecological dynamics. [173] studied the emergence of flocking and symbiotic behaviors in simulated ecosystems. This study reveals that coordination within and across species can spontaneously arise from adaptation processes in MARL-based ecosystems, contributing significantly to our understanding of group behaviors in the natural world. In bioengineering, [174] showcased the application of MARL to synthetic biology to optimize microbial strains for industrial production. This research introduced a model-free MARL approach for tuning metabolic enzyme levels in microbial cells, particularly Escherichia coli and Saccharomyces cerevisiae, to enhance chemical and fuel production.

4.2.3. Medical science and healthcare

For medical treatment, MARL has aroused new methods of coping with complex challenges in various medical fields. Aiming at extracting information from biomedical literature effectively, [175] established an MARL framework for biomedical text mining. This technique can facilitate researchers and practitioners in the biomedical field, as they can handle and extract desired information from biomedical literature and biomedical databases efficiently. In the sphere of combining medicine with engineering, MARL has become an important role in recent years. In the area of medical image processing, [176] studied the problem of efficient anatomical landmark localization in medical images via a collaborative MARL approach with a continuous action space, which significantly outperformed traditional discrete action-based methods and matching supervised regression techniques; [177] used MARL for 3D medical image segmentation for the achievement of significant performance improvements. As for drug research, [178] used MARL to propose a framework termed as multi-agent counterfactual drug target binding affinity (MACDA) for drug target binding affinity prediction, which provided interpretable and optimized insights into drug-target interactions. [179] demonstrated the effectiveness of MARL in robot-assisted surgery, which marks a significant progress in autonomous assistance from robots. For disease prediction and treatment recommendation, [180] introduced an intelligent MARL-based disease prediction model and a treatment recommendation model to utilize mobile agents for data collection and to employ disease attraction weight and disease curing rate; further, [181] developed an updated penguin search optimization algorithm based on MARL-based model, enhancing healthcare decision-making. [182] developed an MARL-based routing protocol for wireless medical sensor networks, such that the reliability and efficiency of the networks can be improved. [183] proposed an MARL strategy for personalized glucose regulation in Type 1 diabetes, showing improved glycemic control. In the aspect of infectious diseases, [184] applied the MARL technique to improve COVID-19 CT image segmentation, which can significantly improve diagnostic accuracy; [185] used MARL for public health policy evaluation, particularly in the context of the U.S. HIV epidemic, demonstrating the potential of the model in informing health strategies. These studies collectively highlight the versatility and effectiveness of MARL in advancing medical research and treatment methodologies in recent years.

4.3. Applications to human and society

The application of MARL to humans and society has also gained considerable attention and transformative insights have been raised. Specific applications are shown in Table 3, covering the scopes from resource management to understanding sophisticated human behaviors.

Table 3
Typical applications of MARL in different branches of human and society.

Scopes	Applications and studies
Resource handling	Common pool resource: [186] Allocation problem: [187,188] Portfolio management: [189–191]
Social dilemmas	Sequential social dilemma (SSD): [192] Matrix game social dilemma (MGSD): [193] Other social dilemmas: [194,195]
Trade	Internet trading: [196] Market order: [197] Risk management: [198,199] Trading parameters optimization: [200,201] Stock exchange: [201–204]
Human and human-like behaviors	Emergent behaviors in pedestrian groups: [205] Intent prediction: [206] Human pose estimation: [207,208] Human behaviors to social dilemma: [209]

4.3.1. Handling of resources

MASs have exhibited considerable potential for tackling the common pool resource (CPR) appropriation problem. For instance, [186] explored the emergent behaviors of agents in CPR environments by using a dynamic CPR environment and independent DQNs [210]. Similarly, [187] contributed to this field by developing methods for improving resource allocation through the MADRL framework. A recent work [188] introduced a contextual budgeting system for the budget allocation problem of network advertising by utilizing contextual multi-armed bandits and transfer learning to optimize online advertising budget allocation. For the portfolio management problem, the MARL strategy has been adopted in various studies in recent years. [189] introduced an MARL system featuring an evolving agent module for signal-based asset information and a strategic agent module utilizing a proximal policy optimization agent for portfolio reallocation. [190] proposed an MARL framework with the utilization of a hierarchical structure and the collective insights of expert traders by applying deep Q-networks for agent training. Lastly, [191] developed an MARL algorithm with trend consistency regularization using dual agents to optimize financial portfolio management. These studies on MARL-based resource management show the capability of managing resources for MASs efficiently.

4.3.2. Social dilemmas

As for the issue of social dilemmas, MARL is beneficial to the modeling and analysis of complex social interactions. Over the past years, several studies have been carried out. [192] conducted a study of sequential social dilemmas (SSDs), conceptualizing them as a class of multi-agent environments where cooperation and competition coexist, in which the agents have to balance short-term gains against the long-term communal benefits. Such a class of dilemma captures the sequential nature of real-world social interactions, addressing the evolution of cooperation in MASs [194]. The complexity of these models involves algorithms that either track potential equilibria for each agent or identify cyclic strategies with multiple policies [211,212]. DQN methods can be used for finding equilibria in SSDs, which is a challenging task for standard evolution and learning methods applicable to matrix game social dilemmas (MGSDs) [193]. The tit-for-tat strategy, as explored in [195], is a game-theoretic approach to tacking complex social dilemmas and exemplifies this balance. This strategy promotes cooperation by responding to the previous action from an opponent, whether cooperative or hostile.

4.3.3. Trade

As another application area of social issue, MARL in trading has received significant attention recently. For Internet trading, [196] developed the distributed coordinated multi-agent bidding (DCMAB) algorithm to optimize real-time online bidding, which can balance competition and cooperation among advertisers effectively. [197] used double deep Q-learning for optimal execution in high-frequency trading and compared its performance with real market data. In terms of risk management, [198] developed a multi-agent simulation of a dealer market. The work demonstrated the adaptability of an RL-based market maker agent to various competitive scenarios, reward structures, and market trends. Lately, [199] has proposed a multi-agent virtual market model using generative adversarial networks to simulate market

Table 4
Benchmark environments for MARL in different applications.

Realms	Environments
Cooperative tasks	Project Malmo [213], Hanabi learning environment [214], CMOTPs [83], PyMARL [215], EPyMARL [216]
Games	Neural MMO [217] , OpenSpiel [218], PettingZoo [219], StarCraft Multi-Agent Challenge [215]
Multiple robots	Soccer robots: RoboCup [220], MuJoCo multi-agent soccer environment [221] Navigation: Multi-agent particle environment [222] Multiple quadcopters: gym-pybullet-drones [223]
Traffic	Traffic flow: SUMO [224,225], Ray platform [226], MACAD-Gym platform [227] Traffic management: CityFlow [228], Flow [229] Railways: Flatland [230] Pedestrian groups: MARL-Ped [231]
Power grids	PowerGridworld [232], PGSim [233]
Economic society	Trading markets: next-generation MAS stock market simulator [203], ABIDES [234] Dynamic economic interactions: TaxAI [235]
Mixed or other tasks	MazeBaze [236], Pommerman [237], Unity [238], Arena [239], MAgent [240]

price changes and devised a trading strategy that outperforms real historical data with higher profit and lower risk. The deployment of MARL in stock trading reveals innovative approaches. [200] combined Q-learning and neural networks to optimize system parameters and value approximations for stock exchange. Continuing this exploration, [202] devised a novel framework using multiple Q-learning agents. The two results were tested on the Korean stock market and showed superior performance in both profit generation and risk management. Recent years, [204] has implemented an MARL strategy for an automatic hedging system in the Vietnam stock market, which can not only minimize losses and maximize gains but also maintain positive profits during market crashes. [203] developed a next-generation MAS stock market simulator, where each agent autonomously learned to exchange stock via RL. Calibrated to real market data from the London stock exchange, the simulator accurately mirrored key metrics of market microstructure. In a subsequent study, [201] extended this approach by analyzing the diversity and effectiveness of exchanging strategies developed by agents. Their findings indicate that successful agents tend to adopt more diverse and fundamentalist exchanging strategies, raising a challenge to conventional models that utilize zero-intelligence agents.

4.3.4. Human and human-like behaviors

During the past several years, MARL has been applied to the understanding and simulation of human behaviors. [205] analyzed emergent behaviors in pedestrian groups using MARL, focusing on scalability and robustness. The proposed model successfully simulated realistic pedestrian dynamics, contributing to our understanding of human movement and interactions in various scenarios. [206] constructed an intent-aware MARL framework for dynamic multi-agent planning by integrating intent prediction with low-level planning. This approach is effective in simulating human-like behaviors in complex interaction scenarios, which shows the versatility of MARL in planning and decision-making processes. In [207], dynamic scenes in multi-view human pose estimation were explored via MARL that was used for optimal camera positioning to avoid occlusions for the estimation of multi-person 3D poses. The result significantly outperforms existing methods, showcasing the efficacy of MARL in real-time, active 3D pose estimation. [208] presented an innovative method for 3D human pose detection by integrating nanosensors and MARL. By extracting electromyogram signals to inform the MARL-based detection model, the method achieved high accuracy in detecting various human poses. The precision and flexibility of this method render it suitable for application in diverse areas, such as medicine and sports. [209] researched the impact of social-cognitive mechanisms on group cooperation, utilizing MARL to model human behavior in a social dilemma task. This finding emphasized the importance of identifiability and reputation tracking in fostering effective cooperation and demonstrated the ability of MARL to mimic human-like cooperation strategies. Altogether, these studies illustrate the significant potential of MARL in advancing our comprehension of complex human behaviors, ranging from physical movements to cooperative strategies and decision-making processes.

4.4. Benchmark environments for MARL

The advance of MARL has been significantly driven by the evolution of benchmark environments, which encapsulate the diversity and intricacy of real-world scenarios. Numerous MARL benchmarks are available for exploring various systems, as listed in Table 4. These benchmark environments play an important role in the development and assessment of MARL algorithms.

4.4.1. Cooperative tasks

In recent years, various MARL environments suitable for cooperative tasks have been launched. Project Malmo [213] provides an open-source AI experimentation platform built on Minecraft, which facilitates research in AI by providing a complex 3D environment for developing flexible agents capable of handling diverse tasks from navigation to collaboration. Focusing on a cooperative learning environment, Hanabi learning environment [214] is a cooperative multiplayer card game in which two to five players are included, which emphasizes communication and shared strategy. By providing a two-agent pixel-based environment, cooperative multi-agent object transportation problems (CMOTPs) [83] stand as a vital benchmark in MASs, which offer a fundamental coordination challenge. Then, [216] introduced EPyMARL that is an extension of the PyMARL codebase [215], along with two new environments focusing on coordination under sparse rewards for multi-agent research.

4.4.2. Games

Inspired by a massively multiplayer online role-playing game setting, Neural MMO [217] simulates large and variable numbers of players in complex worlds, which requires agents to learn resilient policies for combat and navigation, while large groups are trying to reach the same objective. OpenSpiel [218] is a comprehensive framework that integrates a variety of multi-agent environments and algorithms for advanced research in MARL and game theory. PettingZoo was introduced in [219] accompanied by the Agent Environment Cycle (AEC) game model for multi-agent environments, which can address conceptual and practical issues existing in some popular MARL environments. For more intricate scenarios, the StarCraft Multi-Agent Challenge [215], which is based on StarCraft II, provides a rich environment for the evaluation of coordination ability. This benchmark environment serves as a combat-oriented learning environment that facilitates developing algorithms. This StarCraft Multi-Agent Challenge focuses on micromanagement challenges, which incorporates both the granular control of individual agents that act based on local observations and the constraints in the environmental map [215,241]. This benchmark environment is also a nice platform for testing cooperative multi-agent algorithms.

4.4.3. Mixed tasks

To tackle the formidable challenges posed by highly complex multi-agent environments, Pommerman [237] serves as a rigorous testbed for the exploration strategies of MARL, which supports partial observability and inter-agent communication. According to [237,242], Pommerman supports diverse scenarios, including cooperative, competitive, and mixed paradigms; however, the rewards therein are very sparse and delayed. Other related benchmarks for mixed tasks include Unity [238], Arena [239], and MAgent [240], among which Unity and Arena amalgamate many environments to create a unified framework in the multi-agent case, allow a customization of a wide range of multi-agent scenarios, and support various forms of agent interaction. Unity is a versatile platform for developing single- and multi-agent games, from simple grid-worlds to complex strategic games. Arena is a platform built on Unity for developing new multi-agent games and scenarios, where a graphical user interface is offered for the design of game scenarios and the selection from reward schemes to suit different competitive or cooperative game scenarios. It is worth mentioning that MAgent has remarkable scalability, capable of supporting up to one million agents on a single GPU server.

4.4.4. Multiple robots

Simulation-based environments are necessary for the multi-robot application. In the 1990s, soccer server was developed as a simulator of the famous standard problem for robotics research named RoboCup [220]. This environment simulates multiple mobile robots in a two-dimensional space with a network-based graphical interface. MuJoCo [243] is also suitable for the development of robotic motion control, particularly for requiring joint violation avoidance and continuous action spaces. MuJoCo can be used to create customizable simulation environments for the development and evaluation of advanced MARL algorithms. [221] built the MuJoCo multi-agent soccer environment based on the MuJoCo physics engine to simulate a 2-v-2 soccer match, where agents interact within a three-dimensional action space. By consolidating a suite of navigational tasks, Multi-agent particle environment [222] is particularly appropriate for navigation tasks, such as predator–prey game. This environment is applicable to both discrete and continuous action spaces, which also allows players to choose communication actions. As for aerial robotics, [223] introduced an open-source environment called gym-pybullet-drones for multiple quadcopters. This environment leverages the Bullet physics engine, which features multi-agent and vision-based RL interfaces, realistic collisions, and aerodynamic effects.

4.4.5. Traffic management

Simulation of Urban MObility (SUMO) offers a comprehensive, microscopic traffic flow simulation. It addresses issues like comparability among existing models and MARL algorithms for heterogeneous multi-junction urban traffic [224,225]. The Ray platform [226] has expanded to support multi-agent environments with the integration of well-known algorithms. Then, [227] developed the MACAD-Gym platform, focusing on developing control policies for urban driving scenarios using raw sensor data. In traffic management, [228] introduced CityFlow as an environment for traffic signal control in MARL research. Complementarily, [229] constructed a framework called Flow for vehicular control within a hybrid system of human-like drivers and AI agents. In railways, Flatland [230] is a simplified two-dimensional grid environment for the vehicle rescheduling problem, which offers a user-friendly interface to explore and test innovative methods. Incorporating the behaviors of pedestrians, an MARL-based pedestrian virtual environment called MARL-Ped addresses pedestrian behavior modeling at various levels, from local interactions to strategic behaviors and path planning [231]. It adopts model-free RL for agents to implement autonomous navigation.

4.4.6. Power grids

As for the realm of power grids, PowerGridworld [232] and PGSim [233] are two simulation environments. The software of PowerGridworld provides a modular and customizable framework for creating power-systems-focused multi-agent Gym environments, which bridges the gap in rapidly prototyping environments for heterogeneous power systems in MARL. [233] developed an efficient, high-fidelity simulation platform for power grids named PGSim, which was used to facilitate training and evaluating the MADRL algorithm for cooperative controls in power grids.

4.4.7. Economic society

Stock market simulation offers a significant platform for studying the dynamics of multi-agent interactions in a highly stochastic and competitive environment. [203] developed a simulator for the stock market tailored to MASs, making them an important application of MARL. Aiming at facilitating the development, implementation, and analysis of strategic agents within a highly adaptable market setting, an agent-based interactive discrete event simulation environment called ABIDES was introduced in [234]. To simulate dynamic economic interactions among governments, households, firms, and financial intermediaries, TaxAI was designed as an MARL environment in [235]. Based on the Bewley–Aiyagari economic model, TaxAI addresses the complexity of predicting household strategies for effective tax policy implementation, which is a highly realistic economic simulator for generating practical policy recommendations and an effective benchmark for testing MARL methods.

5. Future research directions

MARL has been a dynamic and promising field of AI research, with extensive possibilities for future exploration and application. New ideas are expected to continue developing, and advanced technologies are the key to realizing the full potential of MARL in solving complex challenges.

5.1. Outlook of gap narrowing for MARL implementation

Nowadays, the MARL theory is still far from mature. There exists a certain gap between theory and practical implementation of MARL, which raises critical challenges to the reliability of MARL. It is significant to narrow such gaps in several advanced MARL techniques, including heterogeneous MARL, physics-informed MARL, safe MARL, and the integration of psychology in MARL, such that implementation effectiveness can be ensured in various complex situations.

In heterogeneous MARL, the focus is on agents with different skills and abilities. This includes understanding how agents can use diverse skills for effective policy learning, like human–machine interactions [244,245]. Research should explore algorithms that recognize the unique quality of each agent and support the collaborative learning and decision-making of these agents. This exploration requires balancing performance guarantees with the diverse capabilities of agents, which is crucial for complex applications. By combining physical information with learning algorithms, physics-informed MARL is significant in environments where understanding dynamics is crucial [246,247]. Integrating physical laws into learning is necessary, which can help improve MARL efficiency and robustness in tasks, such as robotic manipulation and environmental monitoring. As an important MARL technique, safe MARL trains agents to achieve long-term rewards while maintaining safety [248]. Although current research in this area is limited, its application in autonomous driving and cooperative games is growing [132,249]. Future work deserves to focus on algorithms that balance risk-taking and safety, incorporating diverse insights to create a framework for safe MARL, especially in sensitive settings. On the other hand, integrating psychology with MARL can redefine agent behaviors and decision-making. This involves applying psychological theories to construct agents with human-like traits, such as reciprocity, intrinsic motivation, and creative problem-solving, and to optimize decision-making in complex situations [244,245,250]. Bridging the gap between psychological theories and practical MARL tasks is expected in the future, which can help improve performance in some real-world tasks, such as robotic cognition and autonomous driving [251,252], and facilitate developing innovative strategies for intrinsic motivation and reward shaping within evolutionary algorithms [93,253].

5.2. Future applications of MARL and technical challenges

MARL holds immense potential in various scopes of applications and helps facilitate complex decision-making processes. One impactful future application of MARL is expected to be healthcare. In intelligent healthcare systems, MARL can refine the service of patient care by optimizing treatment plans for chronic diseases, which enhances operational efficiencies in hospitals and further reduces waiting time. Additionally, MARL has the potential of expediting the progress of drug discovery and development and thus speed up new and more effective medical treatments for a variety of diseases. This promising application may improve patient outcomes significantly. In industrial manufacturing, future MARL may have a great possibility of improving the efficiency of smart factories. Integrated with the Internet of things and cloud computing, MARL can enhance to optimize production processes, reduce waste, and improve product quality. Also, future progress in MARL applications is likely to promote collaborative efforts across various disciplines, especially control theory, focusing on learning stability and adaptability in uncertain conditions. Combining machine learning, game theory, and control theory could greatly strengthen MARL in turn, making it theoretically solid and practically applicable.

6. Concluding remark

We have mainly reviewed different paradigms, practical challenges, and wide-ranging applications of MARL in this survey. Our discussion has covered some types of stochastic games, MARL patterns, different types of MARL tasks, as well as some representative spatial frameworks. A focus has been dedicated to discussing the main challenges, such as scalability, non-stationarity, partial observability, credit assignment, managing continuous action spaces, and complexities in communication and computation. We have highlighted the practical applications of MARL in a variety of fields, such as intelligent machines, chemical engineering, biotechnology, medical treatment, and societal applications. These diverse applications illustrate the broad potential of MARL to provide solutions to some of the most complex and pressing challenges in modern society. We have also reviewed various benchmark environments for MARL in diverse scenarios that cover multi-agent tasks, games, multi-robot systems, traffic simulations, and economic models. These environments not only provide a testbed for MARL algorithms but also demonstrate their potential in addressing real-world problems. Finally, we give our outlook on future research directions. This survey can help strengthen the link between the theoretical aspects of MARL and its practical implementation in a range of real-world scenarios and further promote the developments of both MARL research and application.

CRediT authorship contribution statement

Zepeng Ning: Formal analysis, Investigation, Methodology, Writing – original draft. **Lihua Xie:** Conceptualization, Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

No data was used for the research described in the article.

Acknowledgments

This work was supported by Ministry of Education, Singapore, under AcRF TIER 1 Grant RG64/23, and the Eric and Wendy Schmidt AI in Science Postdoctoral Fellowship, a Schmidt Futures program, USA.

References

[1] E.L. Thorndike, Animal intelligence: An experimental study of the associative processes in animals, Psychol. Rev. Monogr. Suppl. 2 (4) (1898) i–109.

[2] M.L. Minsky, Theory of Neural-Analog Reinforcement Systems and Its Application to the Brain-Model Problem (Ph.D. thesis), Princeton University, Princeton, NJ, USA, 1954.

[3] C.J.C.H. Watkins, P. Dayan, Q-learning, Mach. Learn. 8 (1992) 279–292.

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

[5] D. Silver, A. Huang, C.J. Maddison, A. Guez, L. Sifre, G. van den Driessche, J. Schrittwieser, I. Antonoglou, V. Panneershelvam, M. Lanctot, et al., Mastering the game of Go with deep neural networks and tree search, Nature 529 (2016) 484–489.

[6] S. Levine, C. Finn, T. Darrell, P. Abbeel, End-to-end training of deep visuomotor policies, J. Mach. Learn. Res. 17 (2016) 1–40.

[7] A. Dutta, S. Roy, O.P. Kreidl, L. Bölöni, Multi-robot information gathering for precision agriculture: Current state, scope, and challenges, IEEE Access 9 (2021) 161416–161430.

[8] Z. Zhou, J. Liu, J. Yu, A survey of underwater multi-robot systems, IEEE/CAA J. Autom. Sin. 9 (1) (2022) 1–18.

[9] J.P. Queralta, J. Taipalmaa, B.C. Pullinen, V.K. Sarker, T.N. Gia, H. Tenhunen, M. Gabbouj, J. Raitoharju, T. Westerlund, Collaborative multi-robot search and rescue: Planning, coordination, perception, and active vision, IEEE Access 8 (2020) 191617–191643.

[10] S. Wang, H. Liu, P.H. Gomes, B. Krishnamachari, Deep reinforcement learning for dynamic multichannel access in wireless networks, IEEE Trans. Cogn. Commun. Netw. 4 (2) (2018) 257–265.

[11] Y. Chen, Y. Li, D. Xu, L. Xiao, DQN-based power control for IoT transmission against jamming, in: Proceedings of the IEEE 87th Vehicular Technology Conference, 2018, pp. 1–5.

[12] C.S. Arvind, J. Senthilnath, Autonomous RL: Autonomous vehicle obstacle avoidance in a dynamic environment using MLP-SARSA reinforcement learning, in: Proceedings of the IEEE 5th International Conference on Mechatronics System and Robots, 2019, pp. 120–124.

[13] A. Petrillo, A. Salvi, S. Santini, A.S. Valente, Adaptive multi-agents synchronization for collaborative driving of autonomous vehicles with multiple communication delays, Transp. Res. C 86 (2018) 372–392.

[14] P. Hernandez-Leal, M. Kaisers, T. Baarslag, E.M. de Cote, A survey of learning in multiagent environments: Dealing with non-stationarity, 2017, arXiv:1707.09183.

[15] H.X. Pham, H.M. La, D. Feil-Seifer, A. Nefian, Cooperative and distributed reinforcement learning of drones for field coverage, 2018, arXiv:1803.07250v2.

[16] Q. Mao, F. Hu, Q. Hao, Deep learning for intelligent wireless networks: A comprehensive survey, IEEE Commun. Surv. Tutor. 20 (4) (2018) 2595–2621.

[17] R. Wang, X. He, R. Yu, W. Qiu, B. An, Z. Rabinovich, Learning efficient multi-agent communication: An information bottleneck approach, in: Proceedings of the 37th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 119, 2020, pp. 9908–9918.

[18] J. Kennedy, Swarm intelligence, in: Handbook of Nature-Inspired and Innovative Computing, Springer, Boston, MA, USA, 2006, pp. 187–219.

[19] J. Tang, G. Liu, Q. Pan, A review on representative swarm intelligence algorithms for solving optimization problems: Applications and trends, IEEE/CAA J. Autom. Sin. 8 (10) (2021) 1627–1643.

[20] M. Matta, G.C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Re, F. Silvestri, S. Spanò, Q-RTS: A real-time swarm intelligence based on multi-agent Q-learning, Electron. Lett. 55 (10) (2019) 589–591.

[21] G.C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Matta, A. Nannarelli, M. Re, S. Spanò, FPGA implementation of Q-RTS for real-time swarm intelligence systems, in: Proceedings of the 54th Asilomar Conference on Signals, Systems, and Computers, 2020, pp. 116–120.

[22] Z. Lv, L. Xiao, Y. Du, G. Niu, C. Xing, W. Xu, Multi-agent reinforcement learning based UAV swarm communications against jamming, IEEE Trans. Wireless Commun. 22 (12) (2023) 9063–9075.

[23] M. Hüttenrauch, A. Šošić, G. Neumann, Guided deep reinforcement learning for swarm systems, 2017, arXiv:1709.06011.

[24] T.T. Nguyen, N.D. Nguyen, S. Nahavandi, Deep reinforcement learning for multi-agent systems: A review of challenges, solutions, and applications, IEEE Trans. Cybern. 50 (9) (2020) 3826–3839.

[25] A. Wong, T. Bäck, A.V. Kononova, A. Plaat, Deep multi-agent reinforcement learning: Challenges and directions, Artif. Intell. Rev. 56 (2023) 5023–5056.

[26] Y. Yang, J. Wang, An overview of multi-agent reinforcement learning from game theoretical perspective, 2021, arXiv:2011.00583v3.

[27] A. Feriani, E. Hossain, Single and multi-agent deep reinforcement learning for AI-enabled wireless networks: A tutorial, IEEE Commun. Surv. Tutor. 23 (2) (2021) 1226–1252.

[28] W. Du, S. Ding, A survey on multi-agent deep reinforcement learning: From the perspective of challenges and applications, Artif. Intell. Rev. 54 (2021) 3215–3238.

[29] P. Hernandez-Leal, B. Kartal, M.E. Taylor, A survey and critique of multiagent deep reinforcement learning, Auton. Agents Multi-Agent Syst. 33 (2019) 750–797.

[30] S. Gronauer, K. Diepold, Multi-agent deep reinforcement learning: A survey, Artif. Intell. Rev. 55 (2022) 895–943.

[31] L. Buşoniu, R. Babuška, B. De Schutter, A comprehensive survey of multiagent reinforcement learning, IEEE Trans. Syst. Man. Cybern. C 38 (2) (2008) 156–172.

[32] L. Buşoniu, R. Babuška, B. De Schutter, Multi-agent reinforcement learning: An overview, in: Innovations in Multi-Agent Systems and Applications-1, Springer, Berlin, Heidelberg, 2010, pp. 183–221.

[33] L. Matignon, G.J. Laurent, N. Le Fort-Piat, Independent reinforcement learners in cooperative Markov games: A survey regarding coordination problems, Knowl. Eng. Rev. 27 (1) (2012) 1–31.

[34] D. Bloembergen, K. Tuyls, D. Hennes, M. Kaisers, Evolutionary dynamics of multi-agent learning: A survey, J. Artif. Intell. Res. 53 (2015) 659–697.

[35] P. Hernandez-Leal, B. Kartal, M.E. Taylor, Is multiagent deep reinforcement learning the answer or the question? A brief survey, 2018, arXiv:1810.05587.

[36] F.L. Da Silva, M.E. Taylor, A.H. Reali Costa, Autonomously reusing knowledge in multiagent reinforcement learning, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2018, pp. 5487–5493.

[37] F.L. Da Silva, A.H. Reali Costa, A survey on transfer learning for multiagent reinforcement learning systems, J. Artif. Intell. Res. 64 (2019) 645–703.

[38] A. Oroojlooy, D. Hajinezhad, A review of cooperative multi-agent deep reinforcement learning, Appl. Intell. 53 (2023) 13677–13722.

[39] Z. Zhou, G. Liu, Y. Tang, Multi-agent reinforcement learning: Methods, applications, visionary prospects, and challenges, 2023, arXiv:2305.10091.

[40] I. Althamary, C.W. Huang, P. Lin, A survey on multi-agent reinforcement learning methods for vehicular networks, in: Proceedings of the 15th International Wireless Communications & Mobile Computing Conference, 2019, pp. 1154–1159.

[41] T. Li, K. Zhu, N.C. Luong, D. Niyato, Q. Wu, Y. Zhang, B. Chen, Applications of multi-agent reinforcement learning in future Internet: A comprehensive survey, IEEE Commun. Surv. Tutor. 24 (2) (2022) 1240–1279.

[42] L.M. Schmidt, J. Brosig, A. Plinge, B.M. Eskofier, C. Mutschler, An introduction to multi-agent reinforcement learning and review of its application to autonomous mobility, in: IEEE 25th International Conference on Intelligent Transportation Systems, 2022, pp. 1342–1349.

[43] P. Yadav, A. Mishra, S. Kim, A comprehensive survey on multi-agent reinforcement learning for connected and automated vehicles, Sensors 23 (10) (2023) 4710.

[44] J. Orr, A. Dutta, Multi-agent deep reinforcement learning for multi-robot applications: A survey, Sensors 23 (7) (2023) 3625.

[45] L. Canese, G.C. Cardarilli, L. Di Nunzio, R. Fazzolari, D. Giardino, M. Re, S. Spanò, Multi-agent reinforcement learning: A review of challenges and applications, Appl. Sci. 11 (11) (2021) 4948.

[46] J. Renault, A tutorial on zero-sum stochastic games, 2019, arXiv:1905.06577.

[47] P. Poupart, Partially observable Markov decision processes, in: Encyclopedia of Machine Learning, Springer, Boston, MA, USA, 2011, pp. 754–760.

[48] S.V. Albrecht, F. Christianos, L. Schäfer, Multi-Agent Reinforcement Learning: Foundations and Modern Approaches, MIT Press, Cambridge, MA, USA, 2023.

[49] B. Anahtarci, C.D. Kariksiz, N. Saldi, Q-learning in regularized mean-field games, Dynam. Games Appl. 13 (2023) 89–117.

[50] E.A. Hansen, D.S. Bernstein, S. Zilberstein, Dynamic programming for partially observable stochastic games, in: Proceedings of the 19th National Conference on Artificial Intelligence, 2004, pp. 709–715.

[51] X. Guo, A. Hu, R. Xu, J. Zhang, Learning mean-field games, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, 2019, pp. 4966–4976.

[52] Z. Fu, Z. Yang, Y. Chen, Z. Wang, Actor-critic provably finds Nash equilibria of linear-quadratic mean-field games, in: International Conference on Learning Representations, 2020, https://openreview.net/forum?id=H1lhqpEYPr.

[53] Z. Yang, Y. Chen, M. Hong, Z. Wang, Provably global convergence of actor-critic: A case for linear quadratic regulator with ergodic cost, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, 2019, pp. 8353–8365.

[54] R. Elie, J. Pérolat, M. Laurière, M. Geist, O. Pietquin, On the convergence of model free learning in mean field games, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020, pp. 7143–7150.

[55] X. Guo, A. Hu, R. Xu, J. Zhang, A general framework for learning mean-field games, Math. Oper. Res. 48 (2) (2023) 656–686.

[56] J. Subramanian, A. Mahajan, Reinforcement learning in stationary mean-field games, in: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019, pp. 251–259.

[57] K. Son, D. Kim, W.J. Kang, D.E. Hostallero, Y. Yi, QTRAN: Learning to factorize with transformation for cooperative multi-agent reinforcement learning, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, 2019, pp. 5887–5896.

[58] P. Sunehag, G. Lever, A. Gruslys, W.M. Czarnecki, V. Zambaldi, M. Jaderberg, M. Lanctot, N. Sonnerat, J.Z. Leibo, K. Tuyls, et al., Value-decomposition networks for cooperative multi-agent learning based on team reward, in: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, 2018, pp. 2085–2087.

[59] A. Tampuu, T. Matiisen, D. Kodelja, I. Kuzovkin, K. Korjus, J. Aru, J. Aru, R. Vicente, Multiagent cooperation and competition with deep reinforcement learning, PLoS One 12 (4) (2017) e0172395.

[60] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, I. Mordatch, Multi-agent actor-critic for mixed cooperative-competitive environments, in: Proceedings of the 31st Conference on Neural Information Processing Systems, 2017, pp. 6382–6393.

[61] H. Ryu, H. Shin, J. Park, Multi-agent actor-critic with hierarchical graph attention network, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020, pp. 7236–7243.

[62] G. Weiß, Distributed reinforcement learning, in: The Biology and Technology of Intelligent Autonomous Agents (NATO ASI Series), vol. 144, Springer, Berlin, Heidelberg, 1995, pp. 415–428.

[63] J.N. Foerster, Y.M. Assael, N. de Freitas, S. Whiteson, Learning to communicate with deep multi-agent reinforcement learning, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 2145–2153.

[64] J.K. Gupta, M. Egorov, M. Kochenderfer, Cooperative multi-agent control using deep reinforcement learning, in: Autonomous Agents and Multiagent Systems, Springer, Cham, 2017, pp. 66–83.

[65] J. Jiang, Z. Lu, Learning attentional communication for multi-agent cooperation, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 7265–7275.

[66] P. Peng, Y. Wen, Y. Yang, Q. Yuan, Z. Tang, H. Long, J. Wang, Multiagent bidirectionally-coordinated nets: Emergence of human-level coordination in learning to play StarCraft combat games, 2017, arXiv:1703.10069v4.

[67] S. Sukhbaatar, A. Szlam, R. Fergus, Learning multiagent communication with backpropagation, in: Proceedings of the 30th International Conference on Neural Information Processing Systems, 2016, pp. 2252–2260.

[68] M. Zhou, Y. Chen, Y. Wen, Y. Yang, Y. Su, W. Zhang, D. Zhang, J. Wang, Factorized Q-learning for large-scale multi-agent systems, in: Proceedings of the 1st International Conference on Distributed Artificial Intelligence, 2019, Article 7, 1–7.

[69] M.E. Taylor, P. Stone, Transfer learning for reinforcement learning domains: A survey, J. Mach. Learn. Res. 10 (2009) 1633–1685.

[70] S. Omidshafiei, J. Pazis, C. Amato, J.P. How, J. Vian, Deep decentralized multi-task multi-agent reinforcement learning under partial observability, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, 2017, pp. 2681–2690.

[71] Y. Yang, R. Luo, M. Li, M. Zhou, W. Zhang, J. Wang, Mean field multi-agent reinforcement learning, in: Proceedings of the 35th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 80, 2018, pp. 5571–5580.

[72] Y. Bengio, J. Louradour, R. Collobert, J. Weston, Curriculum learning, in: Proceedings of the 26th Annual International Conference on Machine Learning, 2009, pp. 41–48.

[73] L. Pinto, J. Davidson, R. Sukthankar, A. Gupta, Robust adversarial reinforcement learning, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, 2017, pp. 2817–2826.

[74] J. Heinrich, D. Silver, Deep reinforcement learning from self-play in imperfect-information games, 2016, arXiv:1603.01121v2.

[75] B. Baker, I. Kanitscheider, T. Markov, Y. Wu, G. Powell, B. McGrew, I. Mordatch, Emergent tool use from multi-agent autocurricula, in: International Conference on Learning Representations, 2020, https://openreview.net/forum?id=SkxpxJBKwS.

[76] C. Berner, G. Brockman, B. Chan, V. Cheung, P. Dębiak, C. Dennison, D. Farhi, Q. Fischer, S. Hashme, C. Hesse, et al., Dota 2 with large scale deep reinforcement learning, 2019, arXiv:1912.06680.

[77] D. Silver, T. Hubert, J. Schrittwieser, I. Antonoglou, M. Lai, A. Guez, M. Lanctot, L. Sifre, D. Kumaran, T. Graepel, et al., A general reinforcement learning algorithm that masters chess, shogi, and Go through self-play, Science 362 (6419) (2018) 1140–1144.

[78] J. Foerster, N. Nardelli, G. Farquhar, T. Afouras, P.H.S. Torr, P. Kohli, S. Whiteson, Stabilising experience replay for deep multi-agent reinforcement learning, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, 2017, pp. 1146–1155.

[79] M. van Otterlo, M. Wiering, Reinforcement learning and Markov decision processes, in: Reinforcement Learning, in: Adaptation, Learning, and Optimization, vol. 12, Springer, Berlin, Heidelberg, 2012, pp. 3–42.

[80] M. Lauer, M.A. Riedmiller, An algorithm for distributed reinforcement learning in cooperative multi-agent systems, in: Proceedings of the 17th International Conference on Machine Learning, 2000, pp. 535–542.

[81] C. Claus, C. Boutilier, The dynamics of reinforcement learning in cooperative multiagent systems, in: Proceedings of the 15th National Conference on Artificial Intelligence, 1998, pp. 746–752.

[82] M. Tan, Multi-agent reinforcement learning: Independent vs. Cooperative agents, in: Proceedings of the 10th International Conference on Machine Learning, 1993, pp. 330–337.

[83] G. Palmer, K. Tuyls, D. Bloembergen, R. Savani, Lenient multi-agent deep reinforcement learning, in: Proceedings of the 17th International Conference on Autonomous Agents and Multiagent Systems, 2018, pp. 443–451.

[84] G. Bono, J.S. Dibangoye, L. Matignon, F. Pereyron, O. Simonin, Cooperative multi-agent policy gradient, in: Machine Learning and Knowledge Discovery in Databases, Springer, Cham, 2019, pp. 459–476.

[85] S. Iqbal, F. Sha, Actor-attention-critic for multi-agent reinforcement learning, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, 2019, pp. 2961–2970.

[86] M. Al-Shedivat, T. Bansal, Y. Burda, I. Sutskever, I. Mordatch, P. Abbeel, Continuous adaptation via meta-learning in nonstationary and competitive environments, in: International Conference on Learning Representations, 2018, https://openreview.net/forum?id=Sk2u1g-0-.

[87] N. Rabinowitz, F. Perbet, F. Song, C. Zhang, S.M.A. Eslami, M. Botvinick, Machine theory of mind, in: Proceedings of the 35th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 80, 2018, pp. 4218–4227.

[88] L. Kraemer, B. Banerjee, Multi-agent reinforcement learning as a rehearsal for decentralized planning, Neurocomputing 190 (2016) 82–94.

[89] T. Rashid, M. Samvelyan, C.S. de Witt, G. Farquhar, J. Foerster, S. Whiteson, QMIX: Monotonic value function factorisation for deep multi-agent reinforcement learning, in: Proceedings of the 35th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 80, 2018, pp. 4295–4304.

[90] M. Hausknecht, P. Stone, Deep recurrent Q-learning for partially observable MDPs, in: AAAI 2015 Fall Symposium, 2015, pp. 29–37.

[91] J. Foerster, G. Farquhar, T. Afouras, N. Nardelli, S. Whiteson, Counterfactual multi-agent policy gradients, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 2974–2982.

[92] J. Foerster, F. Song, E. Hughes, N. Burch, I. Dunning, S. Whiteson, M. Botvinick, M. Bowling, Bayesian action decoder for deep multi-agent reinforcement learning, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, 2019, pp. 1942–1951.

[93] M. Jaderberg, W.M. Czarnecki, I. Dunning, L. Marris, G. Lever, A.G. Castañeda, C. Beattie, N.C. Rabinowitz, A.S. Morcos, A. Ruderman, et al., Human-level performance in 3D multiplayer games with population-based reinforcement learning, Science 364 (6443) (2019) 859–865.

[94] O. Vinyals, I. Babuschkin, W.M. Czarnecki, M. Mathieu, A. Dudzik, J. Chung, D.H. Choi, R. Powell, T. Ewalds, P. Georgiev, et al., Grandmaster level in StarCraft II using multi-agent reinforcement learning, Nature 575 (2019) 350–354.

[95] L. Feng, Y. Xie, B. Liu, S. Wang, Multi-level credit assignment for cooperative multi-agent reinforcement learning, Appl. Sci. 12 (14) (2022) 6938.

[96] D.T. Nguyen, A. Kumar, H.C. Lau, Credit assignment for collective multiagent RL with global rewards, in: Proceedings of the 32nd International Conference on Neural Information Processing Systems, 2018, pp. 8113–8124.

[97] H.M. Le, Y. Yue, P. Carr, P. Lucey, Coordinated multi-agent imitation learning, in: Proceedings of the 34th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 70, 2017, pp. 1995–2003.

[98] L. Yu, J. Song, S. Ermon, Multi-agent adversarial inverse reinforcement learning, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, 2019, pp. 7194–7201.

[99] K. Jiang, W. Liu, Y. Wang, L. Dong, C. Sun, Credit assignment in heterogeneous multi-agent reinforcement learning for fully cooperative tasks, Appl. Intell. 53 (2023) 29205–29222.

[100] W. Chen, W. Li, X. Liu, S. Yang, Y. Gao, Learning explicit credit assignment for cooperative multi-agent reinforcement learning via polarization policy gradient, 2023, arXiv:2210.05367v2.

[101] A. Cohen, E. Teng, V.-P. Berges, R.-P. Dong, H. Henry, M. Mattar, A. Zook, S. Ganguly, On the use and misuse of absorbing states in multi-agent reinforcement learning, 2022, arXiv:2111.05992v2.

[102] J. Schulman, S. Levine, P. Moritz, M. Jordan, P. Abbeel, Trust region policy optimization, in: Proceedings of the 32nd International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 37, 2015, pp. 1889–1897.

[103] K. Zhang, Z. Yang, H. Liu, T. Zhang, T. Başar, Fully decentralized multi-agent reinforcement learning with networked agents, in: Proceedings of the 35th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 80, 2018, pp. 5872–5881.

[104] Y. Wen, Y. Yang, R. Luo, J. Wang, W. Pan, Probabilistic recursive reasoning for multi-agent reinforcement learning, in: International Conference on Learning Representations, 2019, https://openreview.net/forum?id=rkl6As0cF7.

[105] Y. Tian, K.-R. Kladny, Q. Wang, Z. Huang, O. Fink, Multi-agent actor-critic with time dynamical opponent model, Neurocomputing 517 (2023) 165–172.

[106] C. Zhu, M. Dastani, S. Wang, A survey of multi-agent reinforcement learning with communication, 2022, arXiv:2203.08975.

[107] Y. Hoshen, VAIN: Attentional multi-agent predictive modeling, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 2698–2708.

[108] A. Das, T. Gervet, J. Romoff, D. Batra, D. Parikh, M. Rabbat, J. Pineau, TarMAC: Targeted multi-agent communication, in: Proceedings of the 36th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 97, 2019, pp. 1538–1546.

[109] A. Singh, T. Jain, S. Sukhbaatar, Learning when to communicate at scale in multiagent cooperative and competitive tasks, in: International Conference on Learning Representations, 2019, https://openreview.net/forum?id=rye7knCqK7.

[110] U. Jain, L. Weihs, E. Kolve, M. Rastegari, S. Lazebnik, A. Farhadi, A.G. Schwing, A. Kembhavi, Two body problem: Collaborative visual task completion, in: Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 6689–6699.

[111] K. Zhang, Z. Yang, T. Başar, Networked multi-agent reinforcement learning in continuous spaces, in: Proceedings of the 2018 IEEE Conference on Decision and Control, 2018, pp. 2771–2776.

[112] T. Chu, S. Chinchali, S. Katti, Multi-agent reinforcement learning for networked system control, in: International Conference on Learning Representations, 2020, https://openreview.net/forum?id=Syx7A3NFvH.

[113] D. Kim, S. Moon, D. Hostallero, W.J. Kang, T. Lee, K. Son, Y. Yi, Learning to schedule communication in multi-agent reinforcement learning, in: International Conference on Learning Representations, 2019, https://openreview.net/forum?id=SJxu5iR9KQ.

[114] S.Q. Zhang, Q. Zhang, J. Lin, Efficient communication in multi-agent reinforcement learning via variance based control, in: Proceedings of the 33rd International Conference on Neural Information Processing Systems, 2019, pp. 3235–3244.

[115] S.Q. Zhang, J. Lin, Q. Zhang, Succinct and robust multi-agent communication with temporal message control, in: Proceedings of the 34th International Conference on Neural Information Processing Systems, 2020, pp. 17271–17282.

[116] H. Mao, Z. Zhang, Z. Xiao, Z. Gong, Y. Ni, Learning agent communication under limited bandwidth by message pruning, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020, pp. 5142–5149.

[117] G. Hu, Y. Zhu, D. Zhao, M. Zhao, J. Hao, Event-triggered multi-agent reinforcement learning with communication under limited-bandwidth constraint, 2020, arXiv:2010.04978.

[118] B. Freed, R. James, G. Sartoretti, H. Choset, Sparse discrete communication learning for multi-agent cooperation through backpropagation, in: Proceedings of the 2020 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2020, pp. 7993–7998.

[119] E. Pesce, G. Montana, Improving coordination in small-scale multi-agent deep reinforcement learning through memory-driven communication, Mach. Learn. 109 (2020) 1727–1747.

[120] G. Melis, C. Dyer, P. Blunsom, On the state of the art of evaluation in neural language models, in: International Conference on Learning Representations, 2018, https://openreview.net/forum?id=ByJHuTgA-.

[121] Z.C. Lipton, J. Steinhardt, Troubling trends in machine learning scholarship, 2018, arXiv:1807.03341v2.

[122] P. Henderson, R. Islam, P. Bachman, J. Pineau, D. Precup, D. Meger, Deep reinforcement learning that matters, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 3207–3214.

[123] G. Tucker, S. Bhupatiraju, S. Gu, R.E. Turner, Z. Ghahramani, S. Levine, The mirage of action-dependent baselines in reinforcement learning, in: Proceedings of the 35th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 80, 2018, pp. 5015–5024.

[124] R. Pascanu, T. Mikolov, Y. Bengio, On the difficulty of training recurrent neural networks, in: Proceedings of the 30th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 28, 2013, pp. 1310–1318.

[125] Y. Yu, Towards sample efficient reinforcement learning, in: Proceedings of the 27th International Joint Conference on Artificial Intelligence, 2018, pp. 5739–5743.

[126] Z. Ding, H. Dong, Challenges of reinforcement learning, in: Deep Reinforcement Learning, Springer, Singapore, 2020, pp. 249–272.

[127] A. Stooke, P. Abbeel, Accelerated methods for deep reinforcement learning, 2019, arXiv:1803.02811v2.

[128] E. Beeching, J. Debangoye, O. Simonin, C. Wolf, Deep reinforcement learning on a budget: 3D control and reasoning without a supercomputer, in: Proceedings of the 25th International Conference on Pattern Recognition, 2021, pp. 158–165.

[129] M. Babaeizadeh, I. Frosio, S. Tyree, J. Clemons, J. Kautz, Reinforcement learning through asynchronous advantage actor-critic on a GPU, in: International Conference on Learning Representations, 2017, https://openreview.net/forum?id=r1VGvBcxl.

[130] H. Qie, D. Shi, T. Shen, X. Xu, Y. Li, L. Wang, Joint optimization of multi-UAV target assignment and path planning based on multi-agent reinforcement learning, IEEE Access 7 (2019) 146264–146272.

[131] J. Cui, Y. Liu, A. Nallanathan, Multi-agent reinforcement learning-based resource allocation for UAV networks, IEEE Trans. Wireless Commun. 19 (2) (2020) 729–743.

[132] S. Shalev-Shwartz, S. Shammah, A. Shashua, Safe, multi-agent, reinforcement learning for autonomous driving, 2016, arXiv:1610.03295.

[133] E. Candela, L. Parada, L. Marques, T.-A. Georgescu, Y. Demiris, P. Angeloudis, Transferring multi-agent reinforcement learning policies for autonomous driving using Sim-to-Real, in: Proceedings of the 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2022, pp. 8814–8820.

[134] S. Bhalla, S. Ganapathi Subramanian, M. Crowley, Deep multi agent reinforcement learning for autonomous driving, in: C. Goutte, X. Zhu (Eds.), Advances in Artificial Intelligence, in: Lecture Notes in Computer Science, vol. 12109, Springer, Cham, 2020, pp. 67–78.

[135] L. Schester, L.E. Ortiz, Automated driving highway traffic merging using deep multi-agent reinforcement learning in continuous state-action spaces, in: Proceedings of the 2021 IEEE Intelligent Vehicles Symposium, 2021, pp. 280–287.

[136] L. Schester, Multi-Agent Reinforcement Learning Autonomous Driving Highway On-Ramp Merge (Ph.D. thesis), College of Engineering & Computer Science, University of Michigan-Dearborn, Dearborn, MI, USA, 2023.

[137] W. Chen, K. Zhou, C. Chen, Real-time bus holding control on a transit corridor based on multi-agent reinforcement learning, in: Proceedings of the IEEE 19th International Conference on Intelligent Transportation Systems, 2016, pp. 100–106.

[138] J.A. Calvo, I. Dusparic, Heterogeneous multi-agent deep reinforcement learning for traffic lights control, in: Proceedings of the 26th Irish Conference on Artificial Intelligence and Cognitive Science, 2018, https://api.semanticscholar.org/CorpusID:57661298.

[139] D.A. Vidhate, P. Kulkarni, Cooperative multi-agent reinforcement learning models (CMRLM) for intelligent traffic control, in: Proceedings of the 1st International Conference on Intelligent Systems and Information Management, 2017, pp. 325–331.

[140] T. Wu, P. Zhou, K. Liu, Y. Yuan, X. Wang, H. Huang, D.O. Wu, Multi-agent deep reinforcement learning for urban traffic light control in vehicular networks, IEEE Trans. Veh. Technol. 69 (8) (2020) 8243–8256.

[141] K. Lin, R. Zhao, Z. Xu, J. Zhou, Efficient large-scale fleet management via multi-agent deep reinforcement learning, in: Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining, 2018, pp. 1774–1783.

[142] Z. Wang, C. Long, G. Cong, Q. Zhang, Error-bounded online trajectory simplification with multi-agent reinforcement learning, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 1758–1768.

[143] P. Kofinas, A.I. Dounis, G.A. Vouros, Fuzzy Q-learning for multi-agent decentralized energy management in microgrids, Appl. Energy 219 (2018) 53–67.

[144] X. Fang, J. Wang, G. Song, Y. Han, Q. Zhao, Z. Cao, Multi-agent reinforcement learning approach for residential microgrid energy scheduling, Energies 13 (1) (2020) 123.

[145] M. Riedmiller, A. Moore, J. Schneider, Reinforcement learning for cooperating and communicating reactive agents in electrical power grids, in: Balancing Reactivity and Social Deliberation in Multi-Agent Systems, in: Lecture Notes in Computer Science, vol. 2103, Springer, Berlin, Heidelberg, 2001, pp. 137–149.

[146] M.S. Rahman, M.A. Mahmud, H.R. Pota, M.J. Hossain, T.F. Orchi, Distributed multi-agent-based protection scheme for transient stability enhancement in power systems, Int. J. Emerg. Electr. Power Syst. 16 (2) (2015) 117–129.

[147] A. Prasad, I. Dusparic, Multi-agent deep reinforcement learning for zero energy communities, in: Proceedings of the 2019 IEEE PES Innovative Smart Grid Technologies Europe, 2019, http://dx.doi.org/10.1109/ISGTEurope.2019.8905628.

[148] Y. Sui, S. Song, A multi-agent reinforcement learning framework for lithium-ion battery scheduling problems, Energies 13 (8) (2020) 1982.

[149] W. Zhang, H. Liu, F. Wang, T. Xu, H. Xin, D. Dou, H. Xiong, Intelligent electric vehicle charging recommendation based on multi-agent reinforcement learning, in: Proceedings of the Web Conference, 2021, pp. 1856–1867.

[150] Y. Zhang, Q. Yang, D. An, D. Li, Z. Wu, Multistep multiagent reinforcement learning for optimal energy schedule strategy of charging stations in smart grid, IEEE Trans. Cybern. 53 (7) (2023) 4292–4305.

[151] P.R.J. Tillotson, Q.H. Wu, P.M. Hughes, Multi-agent learning for routing control within an Internet environment, Eng. Appl. Artif. Intell. 17 (2) (2004) 179–185.

[152] B. Pandey, Adaptive Learning for Mobile Network Management (Master's thesis), School of Science, Aalto University, Espoo, Finland, 2016.

[153] H.K. Mousavi, M. Nazari, M. Takáč, N. Motee, Multi-agent image classification via reinforcement learning, in: Proceedings of the 2019 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2019, pp. 5020–5027.

[154] H.K. Mousavi, G. Liu, W. Yuan, M. Takáč, H. Muñoz-Avila, N. Motee, A layered architecture for active perception: Image classification using deep reinforcement learning, 2019, arXiv:1909.09705.

[155] X. Kong, B. Xin, Y. Wang, G. Hua, Collaborative deep reinforcement learning for joint object search, in: Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 1695–1704.

[156] I. Qaffou, Adaptive image processing using multi-agent reinforcement learning, in: Advanced Intelligent Systems for Sustainable Development, in: Advances in Intelligent Systems and Computing, vol. 1418, Springer, Cham, 2022, pp. 499–507.

[157] R. Jain, P.R. Panda, S. Subramoney, Cooperative multi-agent reinforcement learning-based co-optimization of cores, caches, and on-chip network, ACM Trans. Architect. Code Optim. 14 (4) (2017) 1–25, Article 32.

[158] C. Jiang, R. Yang, Q. Xu, H. Yao, T.-Y. Ho, B. Yuan, A cooperative multiagent reinforcement learning framework for droplet routing in digital microfluidic biochips, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 42 (9) (2023) 3007–3020.

[159] T.-C. Liang, J. Zhou, Y.-S. Chan, T.-Y. Ho, K. Chakrabarty, C.-Y. Lee, Parallel droplet control in MEDA biochips using multi-agent reinforcement learning, in: Proceedings of the 38th International Conference on Machine Learning, in: Proceedings of Machine Learning Research, vol. 139, 2021, pp. 6588–6599.

[160] V. Stephan, K. Debes, H.-M. Gross, F. Wintrich, H. Wintrich, A reinforcement learning based neural multi-agent-system for control of a combustion process, in: Proceedings of the IEEE-INNS-ENNS International Joint Conference on Neural Networks, 2000, pp. 217–222.

[161] K. Chen, H. Wang, B. Valverde-Pérez, S. Zhai, L. Vezzaro, A. Wang, Optimal control towards sustainable wastewater treatment plants based on multi-agent reinforcement learning, Chemosphere 279 (2021) 130498.

[162] Z. He, K.P. Tran, S. Thomassey, X. Zeng, J. Xu, C. Yi, Multi-objective optimization of the textile manufacturing process using deep-Q-network based multi-agent reinforcement learning, J. Manuf. Syst. 62 (2022) 939–949.

[163] Q. Meng, P.D. Anandan, C.D. Rielly, B. Benyahia, Multi-agent reinforcement learning and RL-based adaptive PID control of crystallization processes, Comput. Aided Chem. Eng. 52 (2023) 1667–1672.

[164] J. Li, T. Yu, B. Yang, A data-driven output voltage control of solid oxide fuel cell using multi-agent deep reinforcement learning, Appl. Energy 304 (2021) 117541.

[165] Á. Sass, A. Kummer, J. Abonyi, Multi-agent reinforcement learning-based exploration of optimal operation strategies of semi-batch reactors, Comput. Chem. Eng. 162 (2022) 107819.

[166] Y. Yue, L. Samavedham, Multi-agent reinforcement learning for process control: Exploring the intersection between fields of reinforcement learning, control theory, and game theory, Can. J. Chem. Eng. 101 (11) (2023) 6227–6239.

[167] Y.-C. Choi, H.-S. Ahn, The bio-insect and artificial robots interaction based on multi-agent reinforcement learning, in: Proceedings of the ASME/IEEE 2009 International Conference on Mechatronic and Embedded Systems and Applications, in: Proceedings of the ASME 2009 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, vol. 3, 2009, pp. 9–15.

[168] N. Vaughan, Multi-agent reinforcement learning for swarm retrieval with evolving neural network, in: Biomimetic and Biohybrid Systems, in: Lecture Notes in Computer Science, vol. 10928, Springer, Cham, 2018, pp. 522–526.

[169] X. Wang, S. Liu, Y. Yu, S. Yue, Y. Liu, F. Zhang, Y. Lin, Modeling collective motion for fish schooling via multi-agent reinforcement learning, Ecol. Model. 477 (2023) 110259.

[170] X. Yu, W. Wu, P. Feng, Y. Tian, Swarm inverse reinforcement learning for biological systems, in: Proceedings of the 2021 IEEE International Conference on Bioinformatics and Biomedicine, 2021, pp. 274–279.

[171] M. Kouzehgar, M. Meghjani, R. Bouffanais, Multi-agent reinforcement learning for dynamic ocean monitoring by a swarm of buoys, in: Proceedings of the Global Oceans 2020: Singapore-U.S. Gulf Coast, 2020, http://dx.doi.org/10.1109/IEEECONF38699.2020.9389128.

[172] J. Yamada, J. Shawe-Taylor, Z. Fountas, Evolution of a complex predator-prey ecosystem on large-scale multi-agent deep reinforcement learning, in: Proceedings of the 2020 International Joint Conference on Neural Networks, 2020, http://dx.doi.org/10.1109/IJCNN48605.2020.9206765.

[173] P. Sunehag, G. Lever, S. Liu, J. Merel, N. Heess, J.Z. Leibo, E. Hughes, T. Eccles, T. Graepel, Reinforcement learning agents acquire flocking and symbiotic behaviour in simulated ecosystems, in: Proceedings of the ALIFE 2019: The 2019 Conference on Artificial Life, 2019, pp. 103–110.

[174] M. Sabzevari, S. Szedmak, M. Penttilä, P. Jouhten, J. Rousu, Strain design optimization using reinforcement learning, PLoS Comput. Biol. 18 (6) (2022) e1010177.

[175] M. Camara, O. Bonham-Carter, J. Jumadinova, A multi-agent system with reinforcement learning agents for biomedical text mining, in: Proceedings of the 6th ACM Conference on Bioinformatics, Computational Biology and Health Informatics, 2015, pp. 634–643.

[176] K. Kasseroller, F. Thaler, C. Payer, D. Štern, Collaborative multi-agent reinforcement learning for landmark localization using continuous action space, in: Information Processing in Medical Imaging, in: Lecture Notes in Computer Science, vol. 12729, 2021, pp. 767–778.

[177] X. Liao, W. Li, Q. Xu, X. Wang, B. Jin, X. Zhang, Y. Wang, Y. Zhang, Iteratively-refined interactive 3D medical image segmentation with multi-agent reinforcement learning, in: Proceedings of the 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2020, pp. 9394–9402.

[178] T.M. Nguyen, T.P. Quinn, T. Nguyen, T. Tran, Counterfactual explanation with multi-agent reinforcement learning for drug target prediction, 2021, arXiv:2103.12983v2.

[179] P.M. Scheikl, B. Gyenes, T. Davitashvili, R. Younis, A. Schulze, B.P. Müller-Stich, G. Neumann, M. Wagner, F. Mathis-Ullrich, Cooperative assistance in robotic surgery through multi-agent reinforcement learning, in: Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2021, pp. 1859–1864.

[180] T.R. Rajesh, S. Rajendran, Intelligent multi-agent reinforcement learning based disease prediction and treatment recommendation model, in: Proceedings of the 2022 International Conference on Augmented Intelligence and Sustainable Systems, 2022, pp. 216–221.

[181] T.R. Rajesh, S. Rajendran, M. Alharbi, Penguin search optimization algorithm with multi-agent reinforcement learning for disease prediction and recommendation model, J. Intell. Fuzzy Systems 44 (5) (2023) 8521–8533.

[182] M.S. Hajar, H.K. Kalutarage, M.O. Al-Kadri, 3R: A reliable multi agent reinforcement learning based routing protocol for wireless medical sensor networks, Comput. Netw. 237 (2023) 110073.

[183] M. Jaloli, M. Cescon, Basal-bolus advisor for type 1 diabetes (T1D) patients using multi-agent reinforcement learning (RL) methodology, Control Eng. Pract. 142 (2024) 105762.

[184] H. Allioui, M.A. Mohammed, N. Benameur, B. Al-Khateeb, K.H. Abdulkareem, B. Garcia-Zapirain, R. Damaševičius, R. Maskeliūnas, A multi-agent deep reinforcement learning approach for enhancement of COVID-19 CT image segmentation, J. Personalized Med. 12 (2022) 309.

[185] D. Sharma, A. Shah, C. Gopalappa, A multi-agent reinforcement learning framework for evaluating the U.S. ending the HIV epidemic plan, 2023, arXiv:2311.00855v2.

[186] J. Pérolat, J.Z. Leibo, V. Zambaldi, C. Beattie, K. Tuyls, T. Graepel, A multi-agent reinforcement learning model of common-pool resource appropriation, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 3646–3655.

[187] D.B. Noureddine, A. Gharbi, S.B. Ahmed, Multi-agent deep reinforcement learning for task allocation in dynamic environment, in: Proceedings of the 12th International Conference on Software Technologies, 2017, pp. 17–26.

[188] B. Han, C. Arndt, Budget allocation as a multi-agent system of contextual & continuous bandits, in: Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining, 2021, pp. 2937–2945.

[189] Z. Huang, F. Tanaka, MSPM: A modularized and scalable multi-agent reinforcement learning-based system for financial portfolio management, PLoS One 17 (2) (2022) e0263689.

[190] A. Shavandi, M. Khedmati, A multi-agent deep reinforcement learning framework for algorithmic trading in financial markets, Expert Syst. Appl. 208 (2022) 118124.

[191] C. Ma, J. Zhang, Z. Li, S. Xu, Multi-agent deep reinforcement learning algorithm with trend consistency regularization for portfolio management, Neural Comput. Appl. 35 (2023) 6589–6601.

[192] J.Z. Leibo, V. Zambaldi, M. Lanctot, J. Marecki, T. Graepel, Multi-agent reinforcement learning in sequential social dilemmas, in: Proceedings of the 16th Conference on Autonomous Agents and Multiagent Systems, 2017, pp. 464–473.

[193] M. Kleiman-Weiner, M.K. Ho, J.L. Austerweil, M.L. Littman, J.B. Tenenbaum, Coordinate to cooperate or compete: Abstract goals and joint intentions in social interaction, in: Proceedings of the 38th Annual Conference of the Cognitive Science Society, 2016, pp. 1679–1684.

[194] E.M. de Cote, A. Lazaric, M. Restelli, Learning to cooperate in multi-agent social dilemmas, in: Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems, 2006, pp. 783–785.

[195] A. Lerer, A. Peysakhovich, Maintaining cooperation in complex social dilemmas using deep reinforcement learning, 2018, arXiv:1707.01068v4.

[196] J. Jin, C. Song, H. Li, K. Gai, J. Wang, W. Zhang, Real-time bidding with multi-agent reinforcement learning in display advertising, in: Proceedings of the 27th ACM International Conference on Information and Knowledge Management, 2018, pp. 2193–2201.

[197] M. Karpe, J. Fang, Z. Ma, C. Wang, Multi-agent reinforcement learning in a realistic limit order book market simulation, in: Proceedings of the 1st ACM International Conference on AI in Finance, 2021, Article 30, 1–7.

[198] S. Ganesh, N. Vadori, M. Xu, H. Zheng, P. Reddy, M. Veloso, Reinforcement learning for market making in a multi-agent dealer market, 2019, arXiv:1911.05892.

[199] F.-F. He, C.-T. Chen, S.-H. Huang, A multi-agent virtual market model for generalization in reinforcement learning based trading strategies, Appl. Soft Comput. 134 (2023) 109985.

[200] J.W. Lee, J. O, A multi-agent Q-learning framework for optimizing stock trading systems, in: Proceedings of the 13th International Conference on Database and Expert Systems Applications, in: Lecture Notes in Computer Science, vol. 2453, Springer, Berlin, Heidelberg, 2002, pp. 153–162.

[201] J. Lussange, S. Vrizzi, S. Bourgeois-Gironde, S. Palminteri, B. Gutkin, Stock price formation: Precepts from a multi-agent reinforcement learning model, Comput. Econ. 61 (2023) 1523–1544.

[202] J.W. Lee, J. Park, J. O, J. Lee, E. Hong, A multiagent approach to Q-learning for daily stock trading, IEEE Trans. Syst. Man Cybern. A 37 (6) (2007) 864–877.

[203] J. Lussange, I. Lazarevich, S. Bourgeois-Gironde, S. Palminteri, B. Gutkin, Modelling stock markets by multi-agent reinforcement learning, Comput. Econ. 57 (2021) 113–147.

[204] U. Pham, Q. Luu, H. Tran, Multi-agent reinforcement learning approach for hedging portfolio problem, Soft Comput. 25 (2021) 7877–7885.

[205] F. Martinez-Gil, M. Lozano, F. Fernández, Emergent behaviors and scalability for multi-agent reinforcement learning-based pedestrian models, Simul. Model. Pract. Theory 74 (2017) 117–133.

[206] S. Qi, S.C. Zhu, Intent-aware multi-agent reinforcement learning, in: Proceedings of the 2018 IEEE International Conference on Robotics and Automation, 2018, pp. 7533–7540.

[207] Z. Fan, X. Li, Y. Li, Multi-agent deep reinforcement learning for online 3D human poses estimation, Remote Sens. 13 (19) (2021) 3995.

[208] Y. Sun, X. Che, N. Zhang, 3D human pose detection using nano sensor and multi-agent deep reinforcement learning, Math. Biosci. Eng. 20 (3) (2023) 4970–4987.

[209] K.R. McKee, E. Hughes, T.O. Zhu, M.J. Chadwick, R. Koster, A.G. Castaneda, C. Beattie, T. Graepel, M. Botvinick, J.Z. Leibo, A multi-agent reinforcement learning model of reputation and cooperation in human groups, 2023, arXiv:2103.04982v2.

[210] M.A. Janssen, R. Holahan, A. Lee, E. Ostrom, Lab experiments for the study of social-ecological systems, Science 328 (5978) (2010) 613–617.

[211] M. Zinkevich, A. Greenwald, M.L. Littman, Cyclic equilibria in Markov games, in: Proceedings of the 18th International Conference on Neural Information Processing Systems, 2005, pp. 1641–1648.

[212] J. Pérolat, B. Piot, B. Scherrer, O. Pietquin, On the use of non-stationary strategies for solving two-player zero-sum Markov games, in: Proceedings of the 19th International Conference on Artificial Intelligence and Statistics, in: Proceedings of Machine Learning Research, vol. 51, 2016, pp. 893–901.

[213] M. Johnson, K. Hofmann, T. Hutton, D. Bignell, The Malmo platform for artificial intelligence experimentation, in: Proceedings of the 25th International Joint Conference on Artificial Intelligence, 2016, pp. 4246–4247.

[214] N. Bard, J.N. Foerster, S. Chandar, N. Burch, M. Lanctot, H.F. Song, E. Parisotto, V. Dumoulin, S. Moitra, E. Hughes, et al., The Hanabi challenge: A new frontier for AI research, Artificial Intelligence 280 (2020) 103216.

[215] M. Samvelyan, T. Rashid, C. Schroeder de Witt, G. Farquhar, N. Nardelli, T.G.J. Rudner, C.-M. Hung, P.H.S. Torr, J. Foerster, S. Whiteson, The StarCraft multi-agent challenge, in: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019, pp. 2186–2188.

[216] G. Papoudakis, F. Christianos, L. Schäfer, S.V. Albrecht, Benchmarking multi-agent deep reinforcement learning algorithms in cooperative tasks, in: Proceedings of NeurIPS 2021 Datasets and Benchmarks Track (Round 1), 2021, https://openreview.net/forum?id=cIrPX-Sn5n.

[217] J. Suarez, Y. Du, P. Isola, I. Mordatch, Neural MMO: A massively multiagent game environment for training and evaluating intelligent agents, 2019, arXiv:1903.00784.

[218] M. Lanctot, E. Lockhart, J.-B. Lespiau, V. Zambaldi, S. Upadhyay, J. Pérolat, S. Srinivasan, F. Timbers, K. Tuyls, S. Omidshafiei, et al., OpenSpiel: A framework for reinforcement learning in games, 2020, arXiv:1908.09453v6.

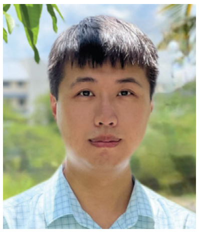
[219] J.K. Terry, B. Black, N. Grammel, M. Jayakumar, A. Hari, R. Sullivan, L. Santos, R. Perez, C. Horsch, C. Dieffendahl, et al., PettingZoo: A standard API for multi-agent reinforcement learning, in: Proceedings of the 35th Conference on Neural Information Processing Systems, in: Advances in Neural Information Processing Systems, vol. 34, 2021, pp. 15032–15043.

[220] H. Kitano, M. Asada, Y. Kuniyoshi, I. Noda, E. Osawa, RoboCup: The robot world cup initiative, in: Proceedings of the 1st International Conference on Autonomous Agents, 1997, pp. 340–347.

[221] S. Liu, G. Lever, J. Merel, S. Tunyasuvunakool, N. Heess, T. Graepel, Emergent coordination through competition, in: International Conference on Learning Representations, 2019, https://openreview.net/forum?id=BkG8sjR5Km.

[222] I. Mordatch, P. Abbeel, Emergence of grounded compositional language in multi-agent populations, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 1495–1502.

[223] J. Panerati, H. Zheng, S.Q. Zhou, J. Xu, A. Prorok, A.P. Schoellig, Learning to fly–a Gym environment with PyBullet physics for reinforcement learning of multi-agent quadcopter control, in: Proceedings of the 2021 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2021, pp. 7512–7519.
[224] M. Behrisch, L. Bieker, J. Erdmann, D. Krajzewicz, SUMO-Simulation of Urban MObility: An overview, in: Proceedings of the 3rd International Conference on Advances in System Simulation, 2011, pp. 55–60.
[225] D. Krajzewicz, J. Erdmann, M. Behrisch, L. Bieker, Recent development and applications of SUMO-Simulation of Urban MObility, Int. J. Adv. Syst. Measur. 5 (3&4) (2012) 128–138.
[226] P. Moritz, R. Nishihara, S. Wang, A. Tumanov, R. Liaw, E. Liang, M. Elibol, Z. Yang, W. Paul, M.I. Jordan, et al., Ray: A distributed framework for emerging AI applications, in: Proceedings of the 13th USENIX Conference on Operating Systems Design and Implementation, 2018, pp. 561–577.
[227] P. Palanisamy, Multi-agent connected autonomous driving using deep reinforcement learning, in: Proceedings of the 2020 International Joint Conference on Neural Networks, 2020, http://dx.doi.org/10.1109/IJCNN48605.2020.9207663.
[228] H. Zhang, S. Feng, C. Liu, Y. Ding, Y. Zhu, Z. Zhou, W. Zhang, Y. Yu, H. Jin, Z. Li, CityFlow: A multi-agent reinforcement learning environment for large scale city traffic scenario, in: The World Wide Web Conference, 2019, pp. 3620–3624.
[229] C. Wu, A. Kreidieh, K. Parvate, E. Vinitsky, A.M. Bayen, Flow: Architecture and benchmarking for reinforcement learning in traffic control, 2017, arXiv:1710.05465.
[230] S. Mohanty, E. Nygren, F. Laurent, M. Schneider, C. Scheller, N. Bhattacharya, J. Watson, A. Egli, C. Eichenberger, C. Baumberger, et al., Flatland-RL: Multi-agent reinforcement learning on trains, 2020, arXiv:2012.05893v2.
[231] F. Martinez-Gil, M. Lozano, F. Fernández, MARL-Ped: A multi-agent reinforcement learning based framework to simulate pedestrian groups, Simul. Model. Pract. Theory 47 (2014) 259–275.
[232] D. Biagioni, X. Zhang, D. Wald, D. Vaidhynathan, R. Chintala, J. King, A.S. Zamzam, PowerGridworld: A framework for multi-agent reinforcement learning in power systems, in: Proceedings of the 13th ACM International Conference on Future Energy Systems, 2022, pp. 565–570.
[233] D. Chen, K. Chen, Z. Li, T. Chu, R. Yao, F. Qiu, K. Lin, PowerNet: Multi-agent deep reinforcement learning for scalable powergrid control, IEEE Trans. Power Syst. 37 (2) (2022) 1007–1017.
[234] D. Byrd, M. Hybinette, T.H. Balch, ABIDES: Towards high-fidelity market simulation for AI research, 2019, arXiv:1904.12066.
[235] Q. Mi, S. Xia, Y. Song, H. Zhang, S. Zhu, J. Wang, TaxAI: A dynamic economic simulator and benchmark for multi-agent reinforcement learning, 2023, arXiv:2309.16307.
[236] S. Sukhbaatar, A. Szlam, G. Synnaeve, S. Chintala, R. Fergus, MazeBase: A sandbox for learning from games, 2016, arXiv:1511.07401v2.
[237] C. Resnick, W. Eldridge, D. Ha, D. Britz, J. Foerster, J. Togelius, K. Cho, J. Bruna, Pommerman: A multi-agent playground, 2022, arXiv:1809.07124v2.
[238] A. Juliani, V.-P. Berges, E. Teng, A. Cohen, J. Harper, C. Elion, C. Goy, Y. Gao, H. Henry, M. Mattar, et al., Unity: A general platform for intelligent agents, 2020, arXiv:1809.02627v2.
[239] Y. Song, A. Wojcicki, T. Lukasiewicz, J. Wang, A. Aryan, Z. Xu, M. Xu, Z. Ding, L. Wu, Arena: A general evaluation platform and building toolkit for multi-agent intelligence, in: Proceedings of the 34th AAAI Conference on Artificial Intelligence, 2020, pp. 7253–7260.
[240] L. Zheng, J. Yang, H. Cai, M. Zhou, W. Zhang, J. Wang, Y. Yu, MAgent: A many-agent reinforcement learning platform for artificial collective intelligence, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, 2018, pp. 8222–8223.
[241] O. Vinyals, T. Ewalds, S. Bartunov, P. Georgiev, A.S. Vezhnevets, M. Yeo, A. Makhzani, H. Küttler, J. Agapiou, J. Schrittwieser, et al., StarCraft II: A new challenge for reinforcement learning, 2017, arXiv:1708.04782.
[242] C. Gao, B. Kartal, P. Hernandez-Leal, M.E. Taylor, On hard exploration for reinforcement learning: A case study in Pommerman, in: Proceedings of the 15th AAAI Conference on Artificial Intelligence and Interactive Digital Entertainment, 2019, pp. 24–30.
[243] E. Todorov, T. Erez, Y. Tassa, MuJoCo: A physics engine for model-based control, in: Proceedings of the 2012 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2012, pp. 5026–5033.
[244] R.S. Sutton, A.G. Barto, Reinforcement Learning: An Introduction (2nd Edition), MIT Press, Cambridge, 2018.
[245] C.-A. Cheng, A. Kolobov, A. Swaminathan, Heuristic-guided reinforcement learning, in: Proceedings of the 35th Conference on Neural Information Processing Systems, 2021, pp. 13550–13563.
[246] T.M. Moerland, J. Broekens, A. Plaat, C.M. Jonker, Model-based reinforcement learning: A survey, Found. Trends Mach. Learn. 16 (1) (2023) 1–118.
[247] W. Sun, N. Jiang, A. Krishnamurthy, A. Agarwal, J. Langford, Model-based RL in contextual decision processes: PAC bounds and exponential improvements over model-free approaches, in: Proceedings of the 32nd Conference on Learning Theory, in: Proceedings of Machine Learning Research, vol. 99, 2019, pp. 2898–2933.
[248] J. García, F. Fernández, A comprehensive survey on safe reinforcement learning, J. Mach. Learn. Res. 16 (42) (2015) 1437–1480.
[249] R.B. Diddigi, D.S.K. Reddy, K.J. Prabuchandran, S. Bhatnagar, Actor-critic algorithms for constrained multi-agent reinforcement learning, in: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019, pp. 1931–1933.
[250] T. Gilovich, D. Griffin, D. Kahneman, Heuristics and Biases: The Psychology of Intuitive Judgment, Cambridge University Press, Cambridge, UK, 2002.
[251] T.R. Colin, T. Belpaeme, A. Cangelosi, N. Hemion, Hierarchical reinforcement learning as creative problem solving, Robot. Auton. Syst. 86 (2016) 196–206.
[252] J.E.T. Taylor, G.W. Taylor, Artificial cognition: How experimental psychology can help generate explainable artificial intelligence, Psychon. Bull. Rev. 28 (2021) 454–475.
[253] J.X. Wang, E. Hughes, C. Fernando, W.M. Czarnecki, E.A. Duéñez-Guzmán, J.Z. Leibo, Evolving intrinsic motivations for altruistic behavior, in: Proceedings of the 18th International Conference on Autonomous Agents and Multiagent Systems, 2019, pp. 683–692.



Zepeng Ning received the Ph.D. degree in control science and engineering from Harbin Institute of Technology, Harbin, China, in 2021. He was also a joint Ph.D. student with the Department of Chemical and Biomolecular Engineering, University of California, Berkeley, CA, USA.

He was a Research Fellow with the School of Chemistry, Chemical Engineering and Biotechnology, Nanyang Technological University, Singapore. He is currently a Schmidt AI in Science Fellow with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore. His research interests include learning-based control, reinforcement learning, and their applications to complex industrial processes.

Lihua Xie received the Ph.D. degree in electrical engineering from the University of Newcastle, Australia, in 1992. Since 1992, he has been with the School of Electrical and Electronic Engineering, Nanyang Technological University, Singapore, where he is currently a professor and Director, Delta-NTU Corporate Laboratory for Cyber-Physical Systems and Director, Center for Advanced Robotics Technology Innovation. He served as the Head of Division of Control and Instrumentation from July 2011 to June 2014. He held teaching appointments in the Department of Automatic Control, Nanjing University of Science and Technology from 1986 to 1989.

Dr Xie's research interests include robust control and estimation, networked control systems, multi-agent networks, localization and unmanned systems. He is an Editor-in-Chief for Unmanned Systems and has served as Editor of IET Book Series in Control and Associate Editor of a number of journals including IEEE Transactions on Automatic Control, Automatica, IEEE Transactions on Control Systems Technology, IEEE Transactions on Network Control Systems, and IEEE Transactions on Circuits and Systems-II. He was an IEEE Distinguished Lecturer (Jan 2012–Dec 2014). Dr Xie is Fellow of Academy of Engineering Singapore, IEEE, IFAC, and CAA.