WToE: Learning When to Explore in Multiagent Reinforcement Learning

Shaokang Dong, Hangyu Mao, Shangdong Yang, Shengyu Zhu, Wenbin Li, Jianye Hao, *Member, IEEE*, and Yang Gao, *Senior Member, IEEE*

***Abstract*—Existing multiagent exploration works focus on *how to explore* in the fully cooperative task, which is insufficient in the environment with nonstationarity induced by agent interactions. To tackle this issue, we propose When to Explore (WToE), a simple yet effective variational exploration method to learn WToE under nonstationary environments. WToE employs an interaction-oriented adaptive exploration mechanism to adapt to environmental changes. We first propose a novel graphical model that uses a latent random variable to model the step-level environmental change resulting from interaction effects. Leveraging this graphical model, we employ the supervised variational auto-encoder (VAE) framework to derive a short-term inferred policy from historical trajectories to deal with the nonstationarity. Finally, agents engage in exploration when the short-term inferred policy diverges from the current actor policy. The proposed approach theoretically guarantees the convergence of the Q-value function. In our experiments, we validate our exploration mechanism in grid examples, multiagent particle environments and the battle game of MAgent environments. The results demonstrate the superiority of WToE over multiple baselines and existing exploration methods, such as MAEXQ, NoisyNets, EITI, and PR2.**

***Index Terms*—Efficient exploration, multiagent, reinforcement learning.**

Manuscript received 8 July 2023; revised 5 October 2023; accepted 27 October 2023. Date of publication 21 November 2023; date of current version 23 July 2024. This work was supported in part by the National Natural Science Foundation of China under Grant 62192783, Grant 62106100, Grant 62206133, and Grant 92370132; in part by the Science and Technology Innovation 2030 New Generation Artificial Intelligence Major Project under Grant 2018AAA0100905; in part by the Natural Science Foundation of Jiangsu Province under Grant BK20221441; in part by the Primary Research and Development Plan of Jiangsu Province under Grant BE2021028; in part by the Shenzhen Fundamental Research Program under Grant 2021Szvup056; and in part by the Research Fund of Guangxi Key Lab of Multi-Source Information Mining and Security under Grant MIMS22-01. This article was recommended by Associate Editor B. Lei. *(Corresponding author: Wenbin Li.)*

Shaokang Dong, Wenbin Li, and Yang Gao are with the State Key Laboratory for Novel Software Technology, Nanjing University, Nanjing 210023, China (e-mail: shaokangdong@smail.nju.edu.cn; liwenbin@nju.edu.cn; gaoy@nju.edu.cn).

Hangyu Mao is with the Smart City Group, SenseTime Research, Beijing 100084, China (e-mail: maohangyu@sensetime.com).

Shangdong Yang is with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing 210023, China, and also with the Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University, Guilin 541004, China (e-mail: sdyang@njupt.edu.cn).

Shengyu Zhu is with the Noah's Ark Lab, Huawei Technologies, Beijing 100085, China (e-mail: zhushyu@outlook.com).

Jianye Hao is with the College of Intelligence and Computing, Tianjin University, Tianjin 300350, China, and also with the Noah's Ark Lab, Huawei Technologies, Beijing 100085, China (e-mail: jianye.hao@tju.edu.cn).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCYB.2023.3328732.

Digital Object Identifier 10.1109/TCYB.2023.3328732

I. Introduction

REINFORCEMENT learning has made great achievements in many fields, including games [1], [2], robotics [3], medical health [4], and transportation [5]. Moreover, numerous real-world applications necessitate the collaborative learning of multiple agents to solve tasks collectively. Unfortunately, efficiently learning an optimal policy in multiagent scenarios poses challenges stemming from the exponentially expanding joint action space, delayed and sparse feedback, consensus problems [6], nonstationary environments, and complex competitive/cooperative relationships [7]. Consequently, given its nature as a trial-and-error method, reinforcement learning agents should engage in exploration in unfamiliar states and reuse the knowledge in frequently visited states, which gives rise to a long-standing dilemma of effectively managing the tradeoff between exploration and exploitation.

Notably, most of the existing multiagent exploration works focus on *how to explore* in fully cooperative tasks [8]. For example, LIIR [9], MAVEN [10], and EITI & EDTI [11] mainly address the selection of appropriate exploratory actions (e.g., randomly, individually motivated, and influence-based) to maintain a balance with learning objectives. Nevertheless, the aforementioned methods focus primarily on how to explore, disregarding the critical issue of When to Explore (WToE) in nonstationary environments caused by agent interactions. In fact, the problem of WToE in single-agent reinforcement learning has been studied in [12]. The work of [12] considers four choices of temporal granularity for exploration, including step-level, experiment-level, episode-level, and intraepisodic. Furthermore, it introduces two "WToE" mechanisms, namely, blind switching and informed switching. However, it does not address the nonstationary problem caused by agent interactions in the complex multiagent setting.

Learning WToE can achieve higher efficiency in multiagent reinforcement learning (MARL). Fig. 1 illustrates this concept with a specific example, in which there are two agents in a grid environment and the optimal policy aims to find the cake as soon as possible. In most cases, as depicted in the left grid, agents do not require extensive exploration because their policies do not influence each other and the environment keeps stationary. However, in situations akin to the scenario depicted in the right grid, agents should engage in more exploration to adapt to the nonstationary environment caused by interactions between agents and prevent collisions.

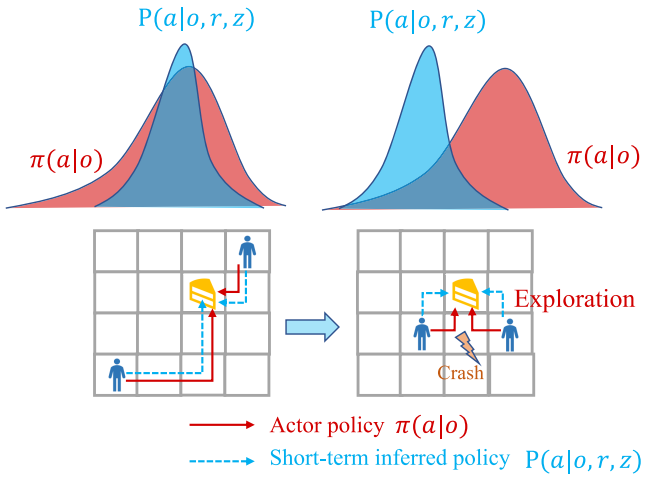


Fig. 1. In the left grid, the actor policies of two agents have not interfered, so the short-term inferred policy is the same as the actor policy and the environment keeps stationary. In the right grid, if two agents maintain the initial actor policy, they may collide with each other. When the two policies differ significantly, the agent should engage in more exploration.

Nevertheless, the simple *WToE* methods are still based on randomness, time decay, or manual switching of exploration modes [12]. Therefore, learning WToE under the nonstationary environment caused by agent interactions remains an open research problem.

In this article, we propose *WToE*, a simple yet effective method to determine WToE in a general multiagent setting. Drawing inspiration from adversarial learning and probabilistic inference literature, WToE formulates an off-policy MARL algorithm capable of addressing challenges posed by nonstationarity in exploration. Specifically, we model environmental changes resulting from interactions as a latent random variable z, and utilize a variational auto-encoder (VAE) to formulate the distribution of z. The decoder of the VAE produces the short-term inferred policy, a supervised policy incorporating short-term transition trajectories and instantaneous z to adapt to the nonstationary environment. Meanwhile, the actor policy $\pi(a|o)$, which disregards the latent random variable z, tends to tackle all the interaction cases moderately, since it marginalizes the latent variable z. Finally, agents intensify exploration when the short-term inferred policy diverges noticeably from the actor policy.

In experiments, we evaluate the performance of our exploration mechanism in Grid examples, multiagent particle environments (MPEs) and the Battle game of MAgent environments. The results demonstrate the superiority of WToE over multiple baselines and existing exploration methods. Moreover, we validate the rapid adaptability of the short-term inferred policy to environmental changes caused by agent interactions. In addition, we find that the 2-norm of z is directly proportional to the intensity of interactions. The principal contributions of this article are summarized as follows.

1) We propose a novel graphical model of MARL, which can deal with the between-step nonstationary problem.
2) We devise a VAE framework to infer the short-term policy capable of quantifying the extent of environmental change caused by intricate agent interactions.
3) We devise an adaptive exploration mechanism that relies on the dissimilarity between the actor policy and short-term inferred policy, while also provide the theoretical convergence guarantee of the Q-value function.
4) We demonstrate empirical performance through three typical benchmarks with existing algorithms and validate the effectiveness of our exploration mechanism.

The remainder of this article is organized as follows. In Section II, we discuss the related work. Then, we will introduce the preliminaries in Section III. Our approach will be illustrated in Section IV. Finally, we report our experimental results in Section V and conclude in Section VI.

II. Related Work

Single-Agent-Oriented Exploration: The bonus-based reward shaping is an intuitive exploration method for single-agent RL. For example, upper confidence bound (UCB) [13] encourages the exploration of less frequented actions in model-based reinforcement learning. This concept extends to model-free Q-learning methods contributing to more sample efficiency [14]. However, efficient methods [15], [16], [17] based on posterior sampling cannot be easily applied in large-scale and continuous environments. To address this, the count-based methods [18], [19] direct the agent toward states seldom visited before. In continuous state space, pseudo-counts methods [20], [21], [22] are proposed to estimate the state density instead of counts. Similar to count-based methods, curiosity-driven methods, such as ICM [23] and RND [24], encourage exploration based on the agent's curiosity. These methods determine the intrinsic reward according to the error of the forward model between the embedding state and the predicted estimate. However, the above methods concentrate more on *how to explore* rather than *WToE*. Currently, the simple yet effective *WToE* methods are still random (e.g., NoisyNets [25]), time-decay exploration (e.g., ϵ-greedy, turned-greedy [26], or exploration-exploitation [27]) or manually switching exploration modes [12]. More specifically, the work [12] devises different exploration modes and demonstrates the effects on the experimental performance.

Another category of existing methods employs variational information maximizing techniques to facilitate exploration [28], [29], [30], [31], [32]. For example, VIREL [29] incorporates the uncertainty in the optimality of the action-value function to drive the exploration. VIME [30] maximizes the information gain concerning the agent's beliefs about environmental dynamics to promote exploration. IPPDO [31] proposes an inference-based method to expedite and stabilize parameter distribution learning, thereby enhancing the agent's exploration capability. VDM [32] explicitly models the multimodality and stochasticity of environmental dynamics using variational inference for efficient exploration. However, these single-agent-oriented methods face infeasibility or inefficiency in multiagent settings due to scalability challenges, as the joint-action space grows exponentially with the number of agents. Furthermore, the aforementioned methods do not

account for the high-dynamic and nonstationary problems of multiagent environments.

Multiagent-Oriented Exploration: Compared with the success of the single-agent field, the progress of the multiagent field is relatively less energetic. In the multiagent setting, the exploration methods focus more on how to guide the agents to cooperate efficiently. For example, LIIR [9] learns the individual reward and devises a centralized critic to address the credit assignment problem. MAVEN [10] hybridizes the value and policy-based methods by introducing a latent space for hierarchical control. The value-based agents condition their behavior on the shared latent variable which generates the different network parameters to diversify the Q-value function for each agent to achieve committed, temporally extended, and efficient exploration. Besides that, the work of EITI & EDTI [11] is proposed to deal with the transition-dependent multiagent problem. EITI employs the mutual information metric to measure the influence of transition dynamics, while EDTI devises an intrinsic reward to measure the influence of one agent's policy on the rewards of other agents. The works [33], [34] introduce flexible exploration strategies for model learning and knowledge sharing in stochastic environments. Nevertheless, the above methods focus on how to explore the environment in fully cooperative tasks and neglect the problem of *WToE* resulting from the impact of the nonstationary environment.

Recently, some works gradually pay close attention to the *WToE* problem [12], [35], [36]. The work of [12] has summarized four choices of temporal granularity for exploratory periods, such as step-level, experiment-level, episode-level, and intraepisodic. That work also analyzes the importance of exploration granularity and switching mechanism (when exactly to start and when to stop an exploratory period, for example, blind and informed switching) in the experiment. In short, our method belongs to intraepisodic exploration methods and devises an adaptive informed trigger signal to determine WToE.

Graphical Model: Moreover, the applications of graphical models [37], [38], [39], [40] may help to address the nonstationary, dynamic or uncertainty problem in the exploration process. For example, PR2 [40] utilizes the variational inference to approximate the other agents' conditional policies in the multiagent setting, which can help the self-agent to make an appropriate decision. The introduction of graphical models in the works of [38] and [39] establishes a basic inference framework and dynamic parameter MDP (DP-MDP). The basic inference framework [38] derives the maximum entropy RL method and LILAC [39] explicitly considers between-episode nonstationarity. However, they cannot deal with the high-dynamic *between-step* nonstationary problem.

III. Preliminaries

In this section, we first introduce the necessary notations and definitions of MARL. Then, we will illustrate some existing graphical models to deal with the nonstationary problem. Finally, we will introduce the application of variational inference in reinforcement learning.

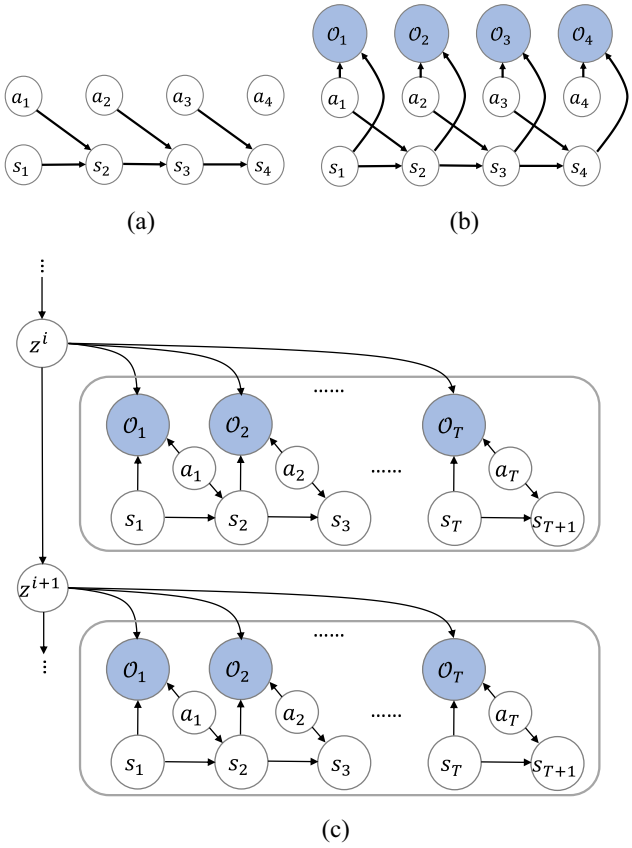


Fig. 2. Three types of graphical models in reinforcement learning. (a) Basic graphical model. (b) Graphical model with optimality. (c) Between-episode dynamic graphical model with optimality.

A. Notation

We consider a tuple $\langle N, S, O, A, T, R\rangle$ for a partially observable stochastic game (POSG) [41], [42], where N is the number of agents, S denotes the global state space in the environment, O is the observation space, and R is the reward function. The action set $A = \{A_1, A_2, \ldots, A_N\}$ represents the actions of all agents in one time step, and $T : S \times A \times S \to [0, 1]$ is the state transition probability. Following the partially observable property, each agent i can only receive a local observation $o_i \in O$ containing the partial information of the global state S. The agent i chooses its action $a_i \in A_i$ based on the actor policy $\pi_i(a_i|o_i; \theta_i)$ which is parameterized by θ_i and conditioned on the local observation o_i. The individual reward for each agent $r_i(s, a_i, a_{-i})$ depends on the global state and actions of all the agents ($-i$ denotes the set of agents except i), and $r_i \in R : S\times A\times S \to \mathbb{R}$. The rewards are not shared by each agent, therefore, it is a general multiagent setting rather than a fully cooperative setting. The ultimate goal for each agent is to maximize the expected return as $J_i(\pi_i(\theta_i)) = \mathbb{E}_{\pi_i}[\sum_{t=0}^{\infty} \gamma^t r_i^t]$, where $\gamma \in [0, 1]$ is the discount factor.

B. Graphical Model

There are three typical graphical models for optimal control in single-agent reinforcement learning. The first model in Fig. 2(a) is a basic graphical model for a Markov decision process (MDP).

The second model in Fig. 2(b) is introduced to incorporate the concept of rewards or costs through optimality variables $\mathcal{O}_t$ [38]. The optimality variable is represented as a binary random variable, where $\mathcal{O}_t = 1$ indicates that the action a_t adheres to the optimal policy at time step t, and otherwise a_t is nonoptimal. Therefore, the probability of the optimality variable corresponding to the fundamental MDP element ($r \in [-1, 0]$) is defined as

$$p(\mathcal{O}_t = 1|s_t, a_t) = \exp(r(s_t, a_t)). \tag{1}$$

In this model, the probability of observing a given trajectory τ under the optimal policy is as follows:

$$\begin{aligned} p(\tau) &= p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t)p(\mathcal{O}_t = 1|s_t, a_t) \\ &= \left[p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t)\right]\exp\left(\sum_{t=1}^{T} r(s_t, a_t)\right). \end{aligned} \tag{2}$$

Moreover, the probability of observing the same trajectory τ under the learning policy $\pi(a_t|s_t)$ is

$$\hat{p}(\tau) = p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t)\pi(a_t|s_t). \tag{3}$$

Minimizing the Kullback–Leibler divergence (KL divergence) $D_{\mathrm{KL}}(\hat{p}(\tau)\|p(\tau))$ of the probability under learning policy $\pi(a_t|s_t)$ and optimal policy can help the learning policy converge to the optimal policy. Therefore, the learning objective is to maximize

$$\begin{aligned} -D_{\mathrm{KL}}\big(\hat{p}(\tau)\|p(\tau)\big) &= \mathbb{E}_{\tau\sim\hat{p}(\tau)}\big[\log p(\tau) - \log\hat{p}(\tau)\big] \\ &= \mathbb{E}_{\tau\sim\hat{p}(\tau)}\left[\sum_{t=1}^{T} r(s_t, a_t) - \log\pi(a_t|s_t)\right] \\ &= \sum_{t=1}^{T}\mathbb{E}_{\tau\sim\hat{p}(\tau)}[r(s_t, a_t) + \mathcal{H}(\pi(a_t|s_t))]. \end{aligned} \tag{4}$$

Namely, the objective, is still to maximize reward and policy entropy $\mathcal{H}(\pi(a_t|s_t))$, which is the so-called soft actor–critic algorithm [43].

The third model [Fig. 2(c)] is introduced to deal with the between-episode nonstationary problem. It regards each episode as an instance of a stationary MDP, while different episodes are considered as different tasks parameterized by the latent variables z^i (the superscript i represents different episodes, contrasting with the subscript t for various steps in the model). Within this formulation, the joint probability distribution of trajectories gathered from M episodes with latent variables is defined as

$$p\left(z^{1:M}, \tau^{1:M}\right) = p\left(z^1\right)p\left(\tau^1|z^1\right)\prod_{i=1}^{M-1} p\left(z^{i+1}|z^i\right)p\left(\tau^{i+1}|z^{i+1}\right). \tag{5}$$

In each episode, given the latent variable z, the prior probability of each trajectory τ is

$$p(\tau|z) = \left[p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t; z)\right]\exp\left(\sum_{t=1}^{T} r(s_t, a_t; z)\right). \tag{6}$$

Similar to the (4), the learning objective under the latent variable z is to maximize

$$\begin{aligned} &-D_{\mathrm{KL}}\big(\hat{p}(\tau; z)\ \|p(\tau)\big) \\ &= \mathbb{E}_{\tau\sim\hat{p}(\tau;z)}\big[\log p(\tau) - \log\hat{p}(\tau; z)\ \big] \\ &= \mathbb{E}_{\tau\sim\hat{p}(\tau;z)}\left[\sum_{t=1}^{T} r(s_t, a_t) - \log\pi(a_t|s_t; z)\right] \\ &= \sum_{t=1}^{T}\mathbb{E}_{\tau\sim\hat{p}(\tau;z)}[r(s_t, a_t) + \mathcal{H}(\pi(a_t|s_t; z))]. \end{aligned} \tag{7}$$

C. Variational Inference

Variational inference is a valuable and tractable method for approximating the posterior $p(\theta|\mathcal{D})$ pertaining to a dataset $\mathcal{D}$. Specifically, the posterior can be estimated using an alternative distribution $q(\theta; \phi)$, parameterized by ϕ, by minimizing the KL divergence $D_{\mathrm{KL}}[q(\theta; \phi)\|p(\theta|\mathcal{D})]$. As in prior works, it is often more convenient to maximize the evidence lower bound (ELBO) as follows:

$$L\big[q(\theta; \phi), \mathcal{D}\big] = \mathbb{E}_{\theta\sim q(\cdot;\phi)}\big[\log p(\mathcal{D}|\theta)\big] - D_{\mathrm{KL}}\big[q(\theta; \phi)\|p(\theta)\big]. \tag{8}$$

In single-agent reinforcement learning, variational inference can be used to approximate the policy with another policy distribution $q(a_t|s_t)$. For example, in the graphical model with optimality variables of Fig. 2(b), the variational ELBO for $\mathcal{O}_{1:T} = 1$ is given by

$$\begin{aligned} &\log p(\mathcal{O}_{1:T} = 1) \\ &= \log\int\int p(\mathcal{O}_{1:T} = 1, s_{1:T}, a_{1:T})ds_{1:T}da_{1:T} \\ &= \log\int\int p(\mathcal{O}_{1:T} = 1, s_{1:T}, a_{1:T})\frac{q(s_{1:T}, a_{1:T})}{q(s_{1:T}, a_{1:T})}ds_{1:T}da_{1:T} \\ &= \log\mathbb{E}_{q(s,a)}\left[\frac{p(\mathcal{O}_{1:T} = 1, s_{1:T}, a_{1:T})}{q(s_{1:T}, a_{1:T})}\right] \\ &\geq \mathbb{E}_{q(s,a)}\big[\log p(\mathcal{O}_{1:T} = 1, s_{1:T}, a_{1:T}) - \log q(s_{1:T}, a_{1:T})\big] \\ &= \mathbb{E}_{q(s,a)}\left[\sum_{t=1}^{T} r(s_t, a_t) - \log q(a_t|s_t)\right] \end{aligned} \tag{9}$$

which is in the same form as the maximum entropy reinforcement learning (4).

In the graphical model of Fig. 2(c), the ELBO objective can be decomposed into two components $\log p(\tau^{1:i-1}) + \log p(\mathcal{O}^i_{1:T} = 1|\tau^{1:i-1})$ [39]. Variational inference is used to approximate the distribution $q(z^i|\tau^i)$. Therefore, the variational lower bound of the first term can be expressed as follows:

$$\log p\left(\tau^{1:i-1}\right) \geq \mathbb{E}_q\left[\sum_{i'=1}^{i}\sum_{t=1}^{T}\log p\left(s_{t+1}, r_t|s_t, a_t; z^{i'}\right) - D_{\mathrm{KL}}\left(q\left(z^{i'}|\tau^{i'}\right)\|p\left(z^{i'}|z^{i'-1}\right)\right)\right] = \mathcal{L}_{\mathrm{rep}}. \tag{10}$$

In this case, the lower-bound $\mathcal{L}_{\mathrm{rep}}$ represents an unsupervised representation learning objective in a sequential latent variable model [39].

For the second term

$$\begin{aligned}
&\log p\left(\mathcal{O}^i_{1:T} = 1|\tau^{1:i-1}\right)\\
&= \log \int p\left(\mathcal{O}^i_{1:T} = 1, z^i|\tau^{1:i-1}\right)dz^i\\
&= \log \int p\left(\mathcal{O}^i_{1:T} = 1|z^i\right)p\left(z^i|\tau^{1:i-1}\right)dz^i\\
&\geq \mathbb{E}_{p\left(z^i|\tau^{1:i-1}\right)}\left[\log p\left(\mathcal{O}^i_{1:T} = 1|z^i\right)\right]\\
&\geq \mathbb{E}\left[\sum_{i=1}^{T} r\left(s_t, a_T; z^i\right) - \log \pi\left(a_t|s_t, z^i\right)\right] = \mathcal{L}_{RL}
\end{aligned} \tag{11}$$

which optimizes both rewards and entropy with the same form as the maximum entropy reinforcement learning (4).

Moreover, in the multiagent setting, variational inference can be utilized to model other agents' policies. For example, the opponent unobservable conditional policy $\rho^{-i}_{\phi^{-i}}(a_{-i}|s, a_i)$ can be inferred through the optimization-based method [40] as

$$\rho^{-i}_{\phi^{-i}}(a_{-i}|s, a_i) = \frac{1}{Z}\exp\left(Q^i_{\pi_\theta}(s, a_i, a_{-i}) - Q^i_{\pi_\theta}(s, a_i)\right) \tag{12}$$

where Z is the normalized term and $Q^i_{\pi_\theta}(s, a_i) = \log \int_{a_{-i}} \exp(Q^i_{\pi_\theta}(s, a_i, a_{-i}))\mathrm{d}a_{-i}$.

IV. Methodology

Exploration plays a crucial role in MARL. However, as previously mentioned, current multiagent exploration methods only consider how to explore. In this section, we propose our method WToE in the *general multiagent setting with partial observation*, where each agent is solely concerned with maximizing its individual reward. The core concept behind WToE is to reserve two distinct policies for each agent, one is the actor policy for decision making, and the other is a short-term inferred policy to capture the environmental change. When a significant disparity arises between these two policies, the agent should engage in exploration. Specifically, we first introduce a novel dynamic graphical model with a latent random variable z to address the *between-step* nonstationary problem. Then we can define an optimal short-term policy with the input of short-term transition trajectories and instantaneous z to capture the environmental change. We further propose a supervised VAE framework to approximate the optimal short-term policy. Finally, we provide the theoretical convergence guarantee of the Q-value function.

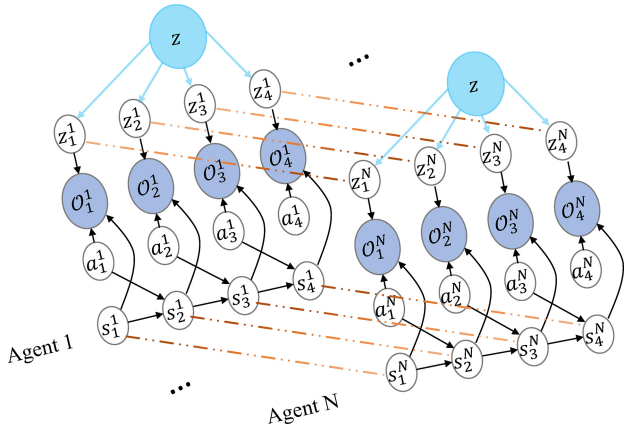


Fig. 3. Between-step dynamic graphical model with optimality in MARL.

A. Dynamic Graphical Model

In the multiagent setting, the environmental change caused by agent interactions plays a pivotal role in shaping adaptive exploration mechanisms. Therefore, it is necessary to incorporate a *between-step* dynamic graphical model that represents the interaction effect, as depicted in Fig. 3. The latent random variable z is used to model the environmental change caused by agent interactions. Specifically, z^k_t represents the interaction effect on agent k at time step t. In contrast to the graphical model of the single-agent setting shown in Fig. 2(c), our proposed model is tailored to address the challenges posed by the partially observable multiagent setting, effectively handling *between-step* nonstationarity caused by agent interactions. Furthermore, in the experiment section, we can find the 2-norm of z is directly proportional to the intensity of interaction, thus validating the fundamental mechanism of WToE.

B. Optimal Policy

Based on our defined dynamic graphical model, the probability of a trajectory τ given latent variable z for each agent can be represented as

$$\begin{aligned}
p(\tau|z) &= p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t, z_t)p(\mathcal{O}_t = 1|s_t, a_t)\\
&= \left[p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t, z_t)\right]\exp\left(\sum_{t=1}^{T} r(s_t, a_t|z_t)\right).
\end{aligned} \tag{13}$$

The goal is to fit the policy $\pi(a_t|s_t, z_t)$ such that the trajectory distribution

$$\hat{p}(\tau|z) = p(s_1)\prod_{t=1}^{T} p(s_{t+1}|s_t, a_t, z_t)\pi(a_t|s_t, z_t) \tag{14}$$

matches the distribution in (13). Therefore, the optimization objective is to maximize

$$\begin{aligned}
&-D_{\mathrm{KL}}\left(\hat{p}(\tau|z)\,\|p(\tau|z)\right)\\
&= \mathbb{E}_\tau\left[\log p(s_1) + \sum_{t=1}^{T}(\log p(s_{t+1}|s_t, a_t, z_t) + r(s_t, a_t|z_t))\right.
\end{aligned}$$

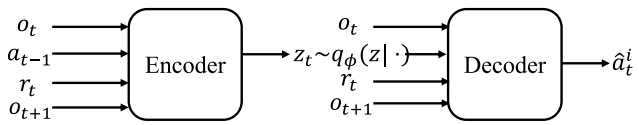


Fig. 4. VAE framework for inferred policy involves online processing of past trajectories comprising observations, actions, and rewards through a recurrent neural network (RNN) to generate the posterior $z \sim q_\phi(z|\cdot)$. Subsequently, this posterior undergoes training by a decoder tasked with predicting the current action based on the current observation and reward.

$$
\begin{aligned}
&- \log p(s_1) - \sum_{t=1}^{T} (\log p(s_{t+1}|s_t, a_t, z_t) + \log \pi(a_t|s_t, z_t)) \Bigg] \\
&= \mathbb{E}_\tau \left[\sum_{t=1}^{T} r(s_t, a_t|z_t) - \log \pi(a_t|s_t, z_t) \right]. \qquad (15)
\end{aligned}
$$

The objective is similar to the maximum entropy reinforcement learning methods, leading to the theoretical derivation of the optimal policy $\pi(a|s, z)$. This optimal policy considers the influence of z to effectively model the nonstationary environment at the step level, thereby mitigating the nonstationary issue. However, directly obtaining the optimal policy $\pi(a|o, z)$ for all instantaneous z poses a challenging task. In the general case, the actor policy $\pi(a|o)$ ignores the latent random variable z and tends to tackle all interaction cases moderately, since it marginalizes the latent variable ($\pi(a|o) = \int \pi(a|o, z)\, dz$). Therefore, in the subsequent section, we will employ a VAE to approximate a short-term policy aimed at capturing environmental changes.

C. Learning the Short-Term Inferred Policy

Without loss of generality, in the partially observable multiagent setting, we opt for the convention of substituting the term "state s" with "observation o" for each agent. In the methodology, we utilize the VAE framework to approximate the distribution of the latent variable z and the short-term policy $p(a|o, r, z)$. The predicted action $\hat{a}$ is sampled from $p(a|o, r, z)$, and subsequently compared with the actual action sampled from the actor policy $\pi(a|o)$ to facilitate the training of the VAE.

The detailed VAE framework is depicted in Fig. 4. When agent k is at time step t, we encode the trajectory data $\{o_t^k, a_{t-1}^k, r_t^k, o_{t+1}^k\}$ to derive the current posterior $q_\phi(z^k|\cdot)$. Subsequently, we decode the current trajectory, including $\{o_t^k, r_t^k, o_{t+1}^k\}$, to predict the action $\hat{a}$. At time step t, the objective of our VAE model is to maximize the log-likelihood function concerning the predicted action $\hat{a}_t$, denoted as $\mathbb{E}_{\hat{a}_t \in \hat{\tau}_t}[\log p(\hat{a}_t = a_t^k|o_t^k, r_t^k)]$. For simplicity, we denote $\log p(\hat{a}_t = a_t^k|o_t^k, r_t^k)$ by $\log p(\hat{a}_t^k|o_t^k, r_t^k)$. To optimize this objective, we aim to maximize its ELBO with a learned posterior distribution $q_\phi(z|o, r)$. For each agent k, the variational ELBO at time step t is calculated as follows:

$$
\begin{aligned}
&\log p\left(\hat{a}_t^k|o_t^k, r_t^k\right) \\
&= \mathbb{E}\left[\log \int p\left(\hat{a}_t^k, z_t^k|o_t^k, r_t^k\right) \frac{q_\phi\left(z_t^k|o_t^k, r_t^k\right)}{q_\phi\left(z_t^k|o_t^k, r_t^k\right)} dz_t^k \right] \\
&= \mathbb{E}\left\{ \log \mathbb{E}_q \left[\frac{p\left(\hat{a}_t^k, z_t^k|o_t^k, r_t^k\right)}{q_\phi\left(z_t^k|o_t^k, r_t^k\right)} \right] \right\} \\
&\geq \mathbb{E}_{q_\phi}\left\{ \log \left[\frac{p\left(\hat{a}_t^k, z_t^k|o_t^k, r_t^k\right)}{q_\phi\left(z_t^k|o_t^k, r_t^k\right)} \right] \right\} \\
&= \mathbb{E}_{q_\phi}\left[\log p\left(\hat{a}_t^k|o_t^k, r_t^k, z_t^k\right) + \log \frac{p\left(z_t^k|o_t^k, r_t^k\right)}{q_\phi\left(z_t^k|o_t^k, r_t^k\right)} \right] \\
&= \mathbb{E}_{q_\phi}\Big[\log p\left(\hat{a}_t^k|o_t^k, r_t^k, z_t^k\right) \\
&\qquad - D_{\mathrm{KL}}\left(q_\phi\left(z_t^k|o_t^k, r_t^k\right) \| p\left(z_t^k|o_t^k, r_t^k\right)\right) \Big] \\
&= -\mathcal{J}_{\mathrm{ELBO}}^k. \qquad (16)
\end{aligned}
$$

Here, the first term is commonly referred to as the reconstruction loss associated with the predicted action. Moreover, the KL divergence between the variational posterior $q_\phi(z_t^k|o_t^k, r_t^k)$ and the prior over the embedding $p(z_t^k|o_t^k, r_t^k)$ is termed as the approximation loss. Through the following experiment section, we can find the 2-norm of z in the VAE is directly proportional to the intensity of interaction.

D. When to Explore

Given the actor policy and the short-term inferred policy in Section IV-C, we utilize both policies to determine WToE in this section. Specifically, the actor policy $\pi(a|o)$ conditions only on agents' observations and tends to tackle all interaction cases moderately by marginalizing the latent variable z. Meanwhile, the short-term inferred policy $p(a|o, r, z)$ takes into account the short-term transition trajectories and instantaneous value of z as inputs. If the short-term inferred policy closely aligns with the actor policy, it implies a stationary environment. Otherwise, the short-term inferred policy considering the instantaneous z may output a distinguishable action. Therefore, the discrepancy between the actor policy and short-term inferred policy serves as an indicator of the degree of environmental change, which can guide agents to adapt their exploration strategies accordingly. In Fig. 1, when the two agents are close as shown in the right grid, they may have collision interactions that may affect the original reward distribution for each agent. In this scenario, the short-term inferred policy can rapidly capture the environmental change and diverge from the initial actor policy. Therefore, the exploration intensity of WToE is proportional to the discrepancy between the actor policy $\pi(a|o)$ and short-term inferred policy $p(a|o, r, z)$. Here, the measure of discrepancy between the two policies is determined by the action error between their output samples.

The ultimate policy takes into account the actor output as well as the discrepancy between the action and short-term predicted action at the last step. Specifically, we introduce a Gaussian noise $\mathcal{N}(0, |a_{t-1}^k - \hat{a}_{t-1}^k|)$ with a variance equal to the discrepancy between the actions (step 8, Algorithm 1). Besides that, the critic's objective for agent k is

$$
\mathcal{J}_{Q^k} = \mathbb{E}_\pi \left[\sum_{t=0}^{\infty} \gamma^t \left(r_t^k + \alpha H\left(\pi_t^k\left(a_t^k|o_t^k\right)\right)\right) \right] \qquad (17)
$$

Algorithm 1: WToE

Input: the number of agents N, learning rates $\alpha_Q, \alpha_\pi, \alpha_\phi$.
Output: the actor and critic of each agent.

1 initialize replay buffer D^k, value function Q^k.
2 **foreach** *episode* i **do**
3 Initialize global state $s, t = 0$.
4 **foreach** *step* t **do**
5 **foreach** *agent* $k \in 0, 1, 2, ..., N$ **do**
6 Get the observation o_t^k.
7 Get actor and VAE output a_{t-1}^k, $\hat{a}_{t-1}^k$.
8 $a_t^k \leftarrow \pi(o_t^k) + \mathcal{N}(0, |a_{t-1}^k - \hat{a}_{t-1}^k|)$.
9 Store $(o_t^k, a_t^k, r_t^k, o_{t+1}^k)$ in D^k.
10 **if** *training* **then**
11 Update critic $\theta_{Q^k} \leftarrow \theta_{Q^k} - \alpha_Q \nabla_{\theta_{Q^k}} \mathcal{J}_{Q^k}$.
12 Update actor $\theta_{\pi^k} \leftarrow \theta_{\pi^k} - \alpha_\pi \nabla_{\theta_{\pi^k}} \mathcal{J}_{\pi^k}$.
13 Update VAE $\phi^k \leftarrow \phi^k - \alpha_\phi \nabla_{\phi^k} \mathcal{J}^k_{\text{ELBO}}$.
14 **end**
15 **end**
16 $s \leftarrow s', t \leftarrow t + 1$.
17 **end**
18 **end**

where the information used to train the critic can be local $Q(o^k, a^k)$ or global $Q(s, a^k, a_{-k})$ information, depending on the specific task. Therefore, the actor gradient for any agent k is defined as (critic with local or global information)

$$\nabla_{\theta_{\pi^k}} \mathcal{J}_{\pi^k} = \mathbb{E}\Big[\nabla_{\theta_{\pi^k}} \log \pi^k\Big(a^k|o^k\Big) Q^k\Big(o^k, a^k\Big)\Big]$$
$$\nabla_{\theta_{\pi^k}} \mathcal{J}_{\pi^k} = \mathbb{E}\Big[\nabla_{\theta_{\pi^k}} \log \pi^k\Big(a^k|o^k\Big) Q^k\Big(s, a^k, a^{-k}\Big)\Big]. \quad (18)$$

Together with the VAE objective in (16), the gradient update is defined in steps 11 to 13, Algorithm 1.

Therefore, the overall objective function for each agent k is to maximize

$$\mathcal{J}^k = \mathbb{E}\Bigg[\sum_{t=0}^{\infty} \gamma^t \Bigg(\underbrace{r_t^k + \alpha H\Big(\pi_t^k\Big(a_t^k|o_t^k\Big)\Big)}_{①} - \underbrace{\beta D_{\text{KL}}\Big(q_\phi\Big(z_t^k|o_t^k, r_t^k\Big)\|p\Big(z_t^k|o_t^k, r_t^k\Big)\Big)}_{②} - \underbrace{\eta D_{\text{KL}}\Big(\pi_t^k\Big(a_t^k|o_t^k\Big)\|p\Big(a_t^k|o_t^k, r_t^k, z_t^k\Big)\Big)}_{③}\Bigg)\Bigg] \quad (19)$$

where $p(a_t^k|o_t^k, r_t^k, z_t^k)$ is the short-term inferred policy and $q_\phi(a_t^k|o_t^k, r_t^k, z_t^k)$ is approximated by VAE, α, β, η are the parameters ranging from 0 to 1. The term ① is in the same form as maximum entropy reinforcement learning methods, and the term ② is utilized to train the VAE framework, and the term ③ is regularized to make actor policy and short-term policy consistent. Finally, we will provide the convergence guarantee of the Q-value function under our exploration mechanism in discrete action space as the following Theorem 1.

Theorem 1: Let $M =< N, S, O, A, T, R >$ be an N-agent POSG. The update of Q value function for agent k in the training process is defined as (to make it easier to read, we omit all the superscripts k, for example, $\{o_t^k, a_t^k, r_t^k, z_t^k\} \rightarrow \{o_t, a_t, r_t, z_t\}$)

$$Q_\pi \coloneqq r_0 + \mathbb{E}\Bigg[\sum_{t=1}^{\infty} \gamma^t (r_t + \alpha H(\pi_t(a_t|o_t)) - \beta D_{\text{KL}}\big(q_{\phi_t}(z_t|o_t, r_t)\|p(z_t|o_t, r_t)\big) - \eta D_{\text{KL}}(\pi_t(a_t|o_t)\|p(a_t|o_t, r_t, z_t)))\Bigg]. \quad (20)$$

The update of policy in discrete action space is calculated as

$$\pi_{t+1} \coloneqq \frac{\exp\Big(\frac{1}{\alpha+\eta} Q_{\pi_t}\Big)}{\sum_{a_k} \exp\Big(\frac{1}{\alpha+\eta} Q_{\pi_t}\Big)}. \quad (21)$$

Then we have $Q_{\pi_{t+1}} \geq Q_{\pi_t}$ and Q_{π_t} converges as $t \rightarrow \infty$.

Proof: First, we introduce the following Lemmas 1 and 2 to help prove the theorem. Lemma 1 is about the monotonicity of important terms in the update equation.

Lemma 1: For the important terms in the update equation of the Q value function, we can obtain that

$$\alpha H(\pi_t) - \eta D_{\text{KL}}(\pi_t\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\big[Q_{\pi_t}\big] \leq \alpha H(\pi_{t+1}) - \eta D_{\text{KL}}(\pi_{t+1}\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_{t+1}}\big[Q_{\pi_t}\big]. \quad (22)$$

Proof: According to the update rule of π, we can get that

$$\mathbb{E}_{\pi_t}\big[Q_{\pi_t}\big] = \mathbb{E}_{\pi_t}\Bigg[Q_{\pi_t} - (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg) + (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg)\Bigg]$$
$$= \mathbb{E}_{\pi_t}\Bigg[(\alpha+\eta)\log(\pi_{t+1}) + (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg)\Bigg] \quad (23)$$
$$\Leftrightarrow \quad \alpha H(\pi_t) - \eta D_{\text{KL}}(\pi_t\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\big[Q_{\pi_t}\big]$$
$$= \alpha H(\pi_t) - \eta D_{\text{KL}}(\pi_t\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\Bigg[\alpha\log(\pi_{t+1}) + \eta\log(\pi_{t+1}) + (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg)\Bigg] \quad (24)$$
$$\Leftrightarrow \quad \alpha H(\pi_t) - \eta D_{\text{KL}}(\pi_t\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\big[Q_{\pi_t}\big]$$
$$= \alpha H(\pi_t) - \mathbb{E}_{\pi_t}\big[\eta\log(\pi_t) - \eta\log p(a_t|o_t, r_t, z_t)\big] + \mathbb{E}_{\pi_t}\Bigg[(\alpha+\eta)\log(\pi_{t+1}) + (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg)\Bigg] \quad (25)$$
$$\Leftrightarrow \quad \alpha H(\pi_t) - \eta D_{\text{KL}}(\pi_t\|p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\big[Q_{\pi_t}\big]$$
$$= -(\alpha+\eta)D_{\text{KL}}(\pi_t\|\pi_{t+1}) + \mathbb{E}_{\pi_t}\Bigg[\eta\log p(a_t|o_t, r_t, z_t) + (\alpha+\eta)\log\Bigg(\sum_{a_k}\exp\Big(\frac{Q_{\pi_t}}{\alpha+\eta}\Big)\Bigg)\Bigg]. \quad (26)$$

Therefore, if we regard the left term as a function with variable π_t, we can find that this function achieves the maximum value when π_t equals π_{t+1}. This is because the KL divergence is always positive and only equals 0 as $\pi_t = \pi_{t+1}$ while the other term is a constant (independent with π). Under this condition, we can get the result of Lemma 1 as

$$\begin{aligned}&\alpha H(\pi_t) - \eta D_{\mathrm{KL}}(\pi_t \| p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_t}\left[Q_{\pi_t}\right] \\ &\leq \alpha H(\pi_{t+1}) - \eta D_{\mathrm{KL}}(\pi_{t+1} \| p(a_t|o_t, r_t, z_t)) + \mathbb{E}_{\pi_{t+1}}\left[Q_{\pi_t}\right].\end{aligned} \tag{27}$$

■

Lemma 2: For the VAE framework, we can get that

$$\begin{aligned}&-\beta D_{\mathrm{KL}}\left(q_{\phi_t}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)\right) \\ &\leq -\beta D_{\mathrm{KL}}\left(q_{\phi_{t+1}}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)\right).\end{aligned} \tag{28}$$

Proof: Note that the design of VAE framework in Fig. 4 and (16), we maximize the log-likelihood function about the prediction action $\hat{a}$ as $\mathbb{E}_{\hat{a}\in\hat{\tau}}[\log p(\hat{a}|o, r)\ ]$. Therefore, it is natural that $-\beta D_{\mathrm{KL}}(q_{\phi_t}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)) \leq -\beta D_{\mathrm{KL}}(q_{\phi_{t+1}}(z_t|o_t, r_t) \| p(z_t|o_t, r_t))$ based on the gradient descent. ■

Finally, we prove Theorem 1 based on Lemmas 1 and 2. Recall the update rule of the Q value function, we can easily get that

$$\begin{aligned}Q_{\pi_t} &= r_0 + \mathbb{E}\Bigg[\sum_{t=1}^{\infty} \gamma^t \big(r_t + \alpha H(\pi_t) - \beta D_{\mathrm{KL}}\big(q_{\phi_t}(z_t|o_t, r_t) \| \\ &\qquad p(z_t|o_t, r_t)\big) - \eta D_{\mathrm{KL}}(\pi_t \| p(a_t|o_t, r_t, z_t))\big)\Bigg] \\ &= r_0 + \gamma\big(\alpha H(\pi_t) - \beta D_{\mathrm{KL}}\big(q_{\phi_t}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)\big) \\ &\quad - \eta D_{\mathrm{KL}}(\pi_t \| p(a_t|o_t, r_t, z_t))) \\ &\quad + \gamma \mathbb{E}_{\pi_t}\left[Q_\pi\left(o', a^{(k)'}, a^{(-k)'}\right)\right] \\ &\leq r_0 + \gamma\big(\alpha H(\pi_{t+1}) - \beta D_{\mathrm{KL}}\big(q_{\phi_t}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)\big) \\ &\quad - \eta D_{\mathrm{KL}}(\pi_{t+1} \| p(a_t|o_t, r_t, z_t))) \\ &\quad + \gamma \mathbb{E}_{\pi_{t+1}}\left[Q_\pi\left(o', a^{(k)'}, a^{(-k)'}\right)\right] \\ &\quad \text{(Lemma 1)} \\ &\leq r_0 + \gamma\big(\alpha H(\pi_{t+1}) - \beta D_{\mathrm{KL}}\big(q_{\phi_{t+1}}(z_t|o_t, r_t) \| p(z_t|o_t, r_t)\big) \\ &\quad - \eta D_{\mathrm{KL}}(\pi_{t+1} \| p(a_t|o_t, r_t, z_t))) \\ &\quad + \gamma \mathbb{E}_{\pi_{t+1}}\left[Q_\pi\left(o', a^{(k)'}, a^{(-k)'}\right)\right] \\ &\quad \text{(Lemma 2)} \\ &= Q_{\pi_{t+1}}.\end{aligned} \tag{29}$$

Because the reward is scaled to $[-1, 0]$, the Q value has an upper bound. Therefore, we can get the convergence of Q value as $Q_{\pi_{t+1}} \geq Q_{\pi_t}$ and Q_{π_t} converges as $t \to \infty$. ■

V. Experiments

Our experiments are conducted in three typical environments. The first is a simple discrete grid environment, where our primary focus lies in validating our exploration mechanism. Here, we can evaluate how the discrepancy between the actor policy and inferred policy captures the intensity of interaction or the environmental change. In the second continuous-space environment, we consider the mixed cooperative and competitive MPEs [44] to evaluate the performance of our approach. Furthermore, to assess the scalability of our approach, we conduct the experiment in the Battle game of the MAgent environment [45], which involves a total of 64×2 initialized agents. All experiments are executed on a machine with a 22-core Intel Xeon Gold 6238 CPU @ 2.10 GHz and 8 GeForce RTX 2080 GPUs.[1]

Fig. 5. Grid examples. Left: 2-room environment. Right: 4-room environment.

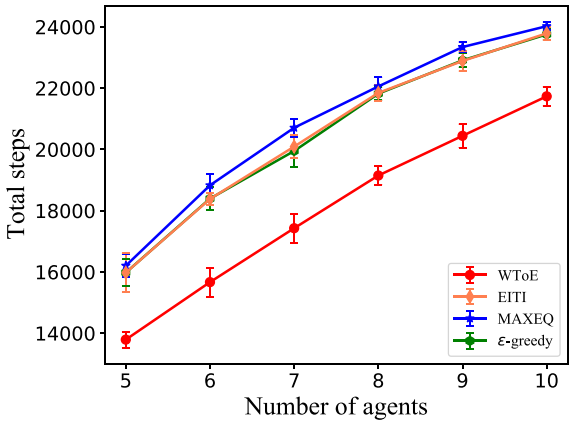


Fig. 6. Total steps are required to complete the task as the number of agents grows in a 2-room environment.

A. Grid Examples

We present two grid examples of multiagent tasks with sparse interaction to explain how WToE works. The first grid example consists of a 7×7 maze with two rooms and a door (Fig. 5 Left). The agents will get punished reward when they have conflicts with each other and only one agent can pass through the door at one time. If two agents collide, both of them will come back to their initial positions before the collision. The optimal policy is to quickly reach the corresponding goal (with the same color) while avoiding collisions. The second, more intricate grid example consists of a 9×5 maze with four rooms and three doors (Fig. 5 Right). As the number of agents grows up to 12, they must engage

[1]The source code can be found at https://github.com/Shaokang-Agent/WToE.

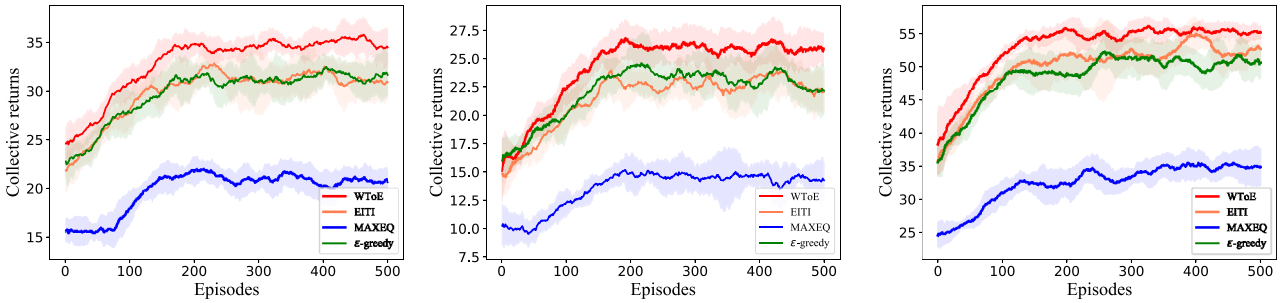


Fig. 7. Return performance of grid examples. From left to right: 1) the collective returns in a 2-room environment (5 agents); 2) the collective returns in a 2-room environment (10 agents); and 3) the collective returns in a 4-room environment (fixed 12 agents).

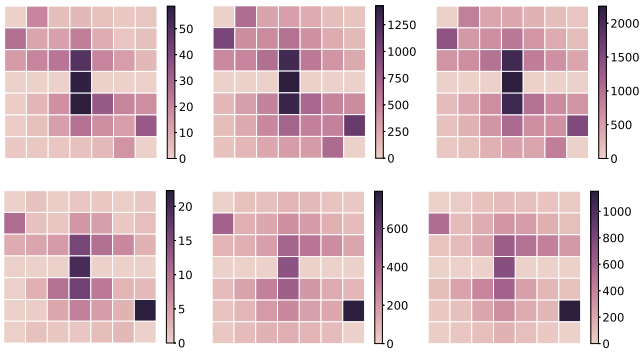


Fig. 8. Heat maps for visited states and policy differences in a 2-room environment (unit: times). (From left to right: 1) visited states in the beginning period; 2) visited states in the middle period; 3) visited states in the ending period; 4) policy differences in the beginning period; 5) policy differences in the middle period; and 6) policy differences in the ending period).

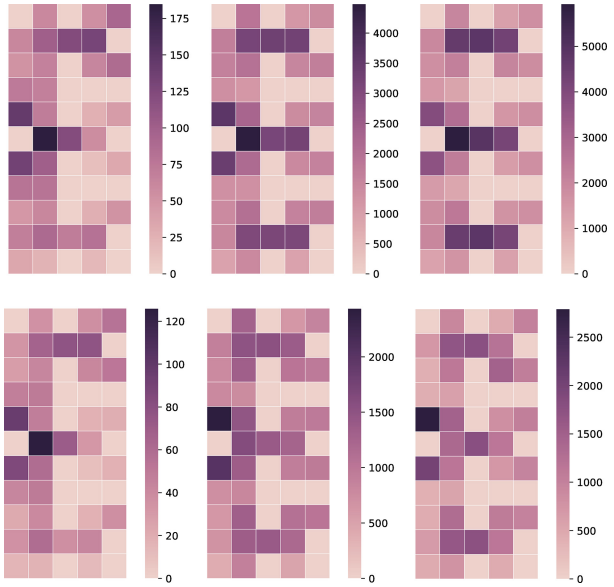


Fig. 9. Heat maps for visited states and policy differences in a 4-room environment (unit: times). (From left to right: 1) visited states in the beginning period; 2) visited states in the middle period; 3) visited states in the ending period; 4) policy differences in the beginning period; 5) policy differences in the middle period; and 6) policy differences in the ending period).

in more exploration in order to successfully navigate through the doors. This allows us to evaluate the effectiveness of our exploration mechanism in accurately identifying regions with a high intensity of interaction. To ensure robustness, we conduct experiments using 50 different random seeds.

In Figs. 6 and 7, it can be observed that the utilization of this exploration mechanism enhances the performance of the baseline ϵ-greedy [the basic graphical model in Fig. 2(a)] and maximum entropy Q-learning (MAXEQ) [38] [the graphical model with optimality in Fig. 2(b)]. Besides that, we also compare WToE with EITI [11]. However, due to the requirement of a centralized value for all agents in EDTI [11] and the unavailability of code for LILAC [39] [the dynamic graphical model in Fig. 2(c)], no comparison has been made with these baselines. Specifically, in Fig. 6, WToE demonstrates the ability to achieve step reductions of approximately 12% to 17% of completing the task as the number of agents varies from 5 to 10. Furthermore, concerning collective returns, as indicated in Fig. 7(1)–(3), WToE exhibits the highest-growth rate and convergent value in both 2-room and 4-room environments. Interestingly, MAXEQ fails to achieve comparable performance in terms of collective returns in these two environments when compared to ϵ-greedy. This result may be attributed to the deterministic nature of the environments and the effectiveness of ϵ-greedy, while MAXEQ excels in stochastic environments.

However, the underlying exploration mechanism of WToE is not directly reflected in the overall performance metrics, such as total steps and collective returns. Therefore, we provide a potential explanation for our exploration approach in Figs. 8 and 9. The vertical axis represents the number of visited states and the times that policy differences occur. We experiment with 500 episodes and subsequently divide them into three stages: 1) beginning period (1–5 episodes); 2) middle period (6–200 episodes); and 3) ending period (201–500 episodes). From Figs. 8 and 9, it is obvious to find that the agents are more concentrated on areas near the doors and goals. This behavior is driven by the intense interactions required to navigate through the doors and reach the goals. We initialize the Q-value for each agent as the single-agent optimal Q-value. Therefore, all the agents are intended to pass through the doors resulting in frequent collisions. The baseline methods require numerous failed attempts to learn how to successfully navigate through the doors, as the long-term stable Q-value function is resistant to modification. However, our short-term inferred policy can quickly capture the environmental change. As shown in Figs. 8 and 9, the discrepancy between actor policy and short-term policy usually happens near the doors and goals, aligning with our motivation to explore areas with

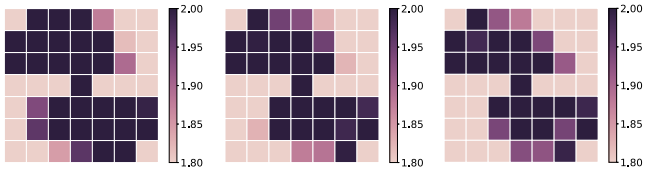


Fig. 10. Heat maps for the 2-norm of "z" in a 2-room environment (left: beginning period, middle: middle period, and right: ending period).

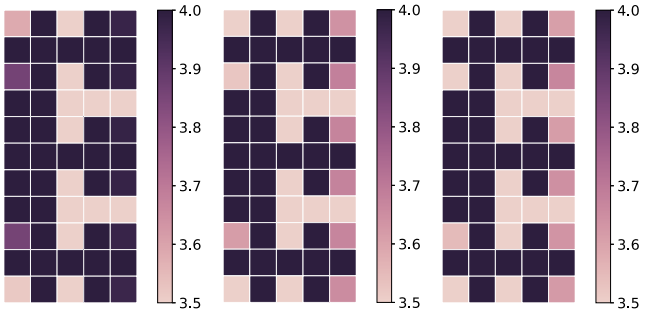


Fig. 11. Heat maps for the 2-norm of "z" in a 4-room environment (left: beginning period, middle: middle period, and right: ending period).

high uncertainty. Besides that, we can find that the policy differences happen more near the goals rather than the doors in the ending period, which indicates that the WToE method has learned well how to pass through the doors in order.

Another concern is about the relationship between the latent variable z and the environmental change. To address this, we conduct an experiment to quantify the 2-norm of z in the grid examples, aiming to demonstrate the intensity of interaction among different agents. In Figs. 10 and 11, we can find that the high values of z tend to aggregate in regions of strong interaction, such as doors and goals. Moreover, as the learning episode progresses, the concentration of these values increases. Hence, it is reasonable to posit that z can effectively capture the intensity of interaction within the environment.

B. Multiagent Particle Environment

We consider the mixed cooperative and competitive environment MPE [44] in the experiments. This includes six basic testing scenarios (two cooperative scenarios and four competitive scenarios).

1) *Cooperative communication* with two agents and three landmarks of different colors. Each agent wants to get to their target landmark, which is known only by the other agent. Therefore, the agents have to learn to communicate the goal of the other agent, and then navigate to their landmark.
2) *Cooperative navigation* is a team cooperative task with three agents and three landmarks. Agents will be rewarded based on how close they are to each landmark. If agents clash with other agents, they will be punished. Therefore, agents must learn to cover all landmarks while avoiding collisions;
3) *Physical deception* with one adversary agent, two good agents, and two landmarks. All agents can observe the location of landmarks and other agents. One of these landmarks is the "target landmark." Good agents will be rewarded based on their distance from the target landmark but will be negatively rewarded if the adversary agent approaches the target landmark. The adversary agent gets a reward based on its distance from the target but does not know which landmark is the target landmark. Therefore, a good agent must learn to "split up," covering all landmarks to deceive the adversary agent.
4) *Covert communication* with two good agents (Alice and Bob) and one adversary agent (Eve). Alice must send a private message to Bob over a public channel. Alice and Bob are rewarded based on how well Bob reconstructs the message, but negatively rewarded if Eve can reconstruct the message. Alice and Bob have a private key (randomly generated at the beginning of each episode), which they must learn to use to encrypt the message.
5) *Keep-away* with one agent, one adversary, and one landmark. The agent is rewarded based on the distance to the landmark. The adversary is rewarded if it is closer to the landmark. Therefore, the adversary learns to push the agent away from the landmark.
6) *Predator-prey* with one prey agent and three predator agents. The prey agent moves faster and tries to avoid being hit by the predator agents.

The critic and actor networks consist of four fully connected layers with 100 units. The batch size is 64, the discounted factor is 0.95, and the learning rates of the critic network and policy network are fixed as $1e-3$ and $1e-4$. The encoder follows a GRU network and the decoder is a 2 fully connected layers network. The batch size for VAE is 32, and the learning rate is fixed as $1e-3$. We run the experiment with 1000 episodes, each of which has 1000 steps. We compare our WToE method with the basic algorithm MADDPG [44], the exploration algorithm NoisyNets [25], and the variational inference algorithm PR2 [40]. Given that competitive tasks involve two camps, we maintain a fixed policy for the adversary agent, which follows a random policy. Conversely, the policies of the good agents are based on our WToE method and other comparable approaches. The specific results are shown in Fig. 12. From the results, we can find that WToE and PR2 exhibit a substantial advantage over the basic MADDPG and NoisyNets on average. Note that, in the *Keep-away* scenario, PR2 is not evaluated due to the presence of only one good agent (PR2 is designed to infer policies of multiple good agents). In summary, apart from the initial substantial fluctuations observed in the *Physical Deception* scenario, WToE consistently outperforms all other methods in terms of collective returns. An intriguing observation is that the WToE method demonstrates rapid convergence and minimal oscillation within the environment, indicative of its efficacy in exploring environmental uncertainties.

Moreover, in order to conduct a more comprehensive comparison of the efficacy of WToE, we run different algorithms in two camps rather than employing a random policy for the adversary agent. Because the rewards of agents are unstable when both two camps are concurrent training learnable algorithms, we use the index of normalized

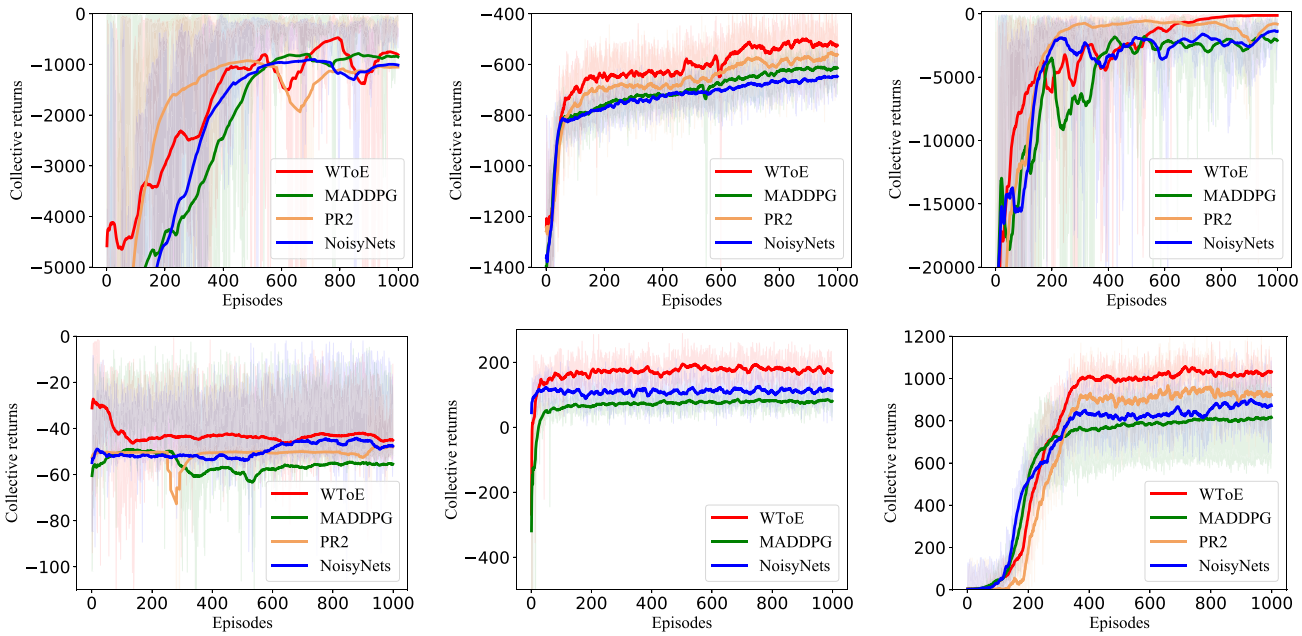


Fig. 12. Collective returns per 1000 steps in MPE. From the left to right and from the top to bottom: 1) *Cooperative communication*; 2) *Cooperative navigation*; 3) *Physical deception*; 4) *Covert communication*; 5) *Keep-away*; and 6) *Predator-prey*.

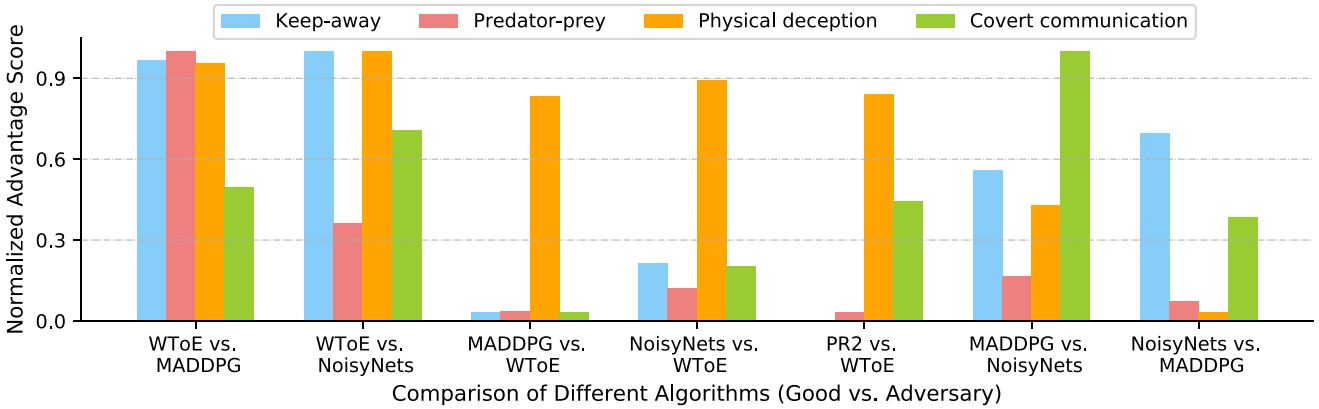


Fig. 13. Performance comparison of WToE, MADDPG, NoisyNets, and PR2. Each bar shows the 0–1 normalized advantage score (good agent reward—adversary agent reward) in the four competitive scenarios of MPE. Higher score is better.

advantage score (good agent reward—adversary agent reward) on the converged period to test the performance. As shown in Fig. 13, we can find that WToE can achieve more advantage scores when compared with other algorithms (in the first and second bars). Moreover, other algorithms tend to perform poorly when compared with WToE (a huge decrease in the third, fourth and fifth bars).

C. Battle Game of MAgent Environment

We consider the Battle game of the MAgent environment [45] to assess the scalability of our method. The environment consists of two teams, namely, the red team and the blue team, each comprising 64 agents. The objective of each team is to eliminate all opposing agents. Agents have the ability to perform actions, such as movement and attacking nearby enemies. The associated rewards for these actions are as follows: -0.005 for a movement action, 0.2 for attacking an enemy, 5 for eliminating an enemy, -0.1 for attacking an empty space, and -0.1 for being attacked or eliminated. The specific visual representation of the environment is depicted in Fig. 14.

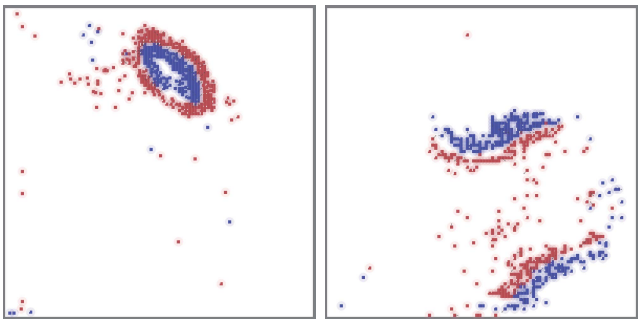


Fig. 14. Scene of the battle game with 64×2 initialed agents.

We incorporate our WToE method into two baselines, namely, independent Q-learning (IQL) and independent actor–critic (IAC). In all algorithms, the batch size is 64,

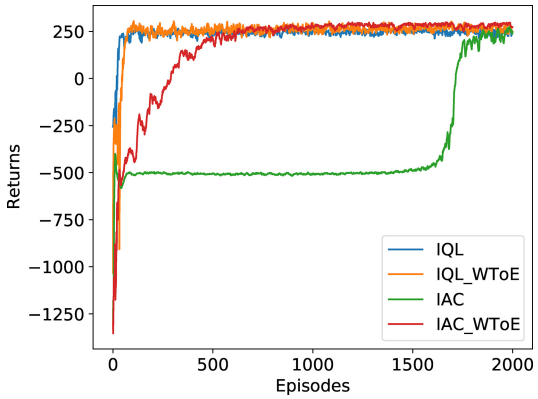


Fig. 15. Performance comparison of WToE, two baselines IQL and IAL. The returns are averaged of two teams per 400 step in each episode.

the discounted factor is 0.95, and the learning rate is fixed as $1e-4$. In IQL, the Q-value networks consist of two CNN layers with 32 filters and three kernel sizes, two fully connected layers with 256 and 32 units. In IAC, the critic and actor networks share 2 fully connected layers with 256 units. Besides that, the critic has an independent fully connected layer to output the value while the actor has an independent fully connected layer to output the action probability. For the VAE component of the WToE method, the encoder follows a GRU network and the decoder is a 2-layer fully connected network. We run the experiment with 2000 episodes, each of which has 400 steps. As shown in Fig. 15, it can be observed that the incorporation of WToE leads to an improved convergence rate and higher-overall returns, especially for the IAC baseline.

VI. Conclusion

In this research article, we propose the WToE method as a solution to address the challenges of exploration and exploitation in the nonstationary multiagent environment. The convergence property of WToE is theoretically proven, and its superiority over multiple baselines is demonstrated through experiments conducted in Grid examples, MPEs and the Battle game with many agents. Our objective is to expand the understanding of the exploration mechanism in the general multiagent setting, encompassing cooperative, competitive, and mixed scenarios. It is important to note that exploration mechanism research is an ongoing endeavor, and in addition to the aforementioned issues, it is crucial to consider how agents can coordinate their exploration efforts. Particularly in large-scale fully cooperative tasks, we think that the current exploration methods are inadequate in effectively allocating exploration tasks among different agents.

References

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

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

[3] J. Kober, J. A. Bagnell, and J. Peters, "Reinforcement learning in robotics: A survey," *Int. J. Robot. Res.*, vol. 32, no. 11, pp. 1238–1274, 2013.

[4] E. O. Neftci and B. B. Averbeck, "Reinforcement learning in artificial and biological systems," *Nat. Mach. Intell.*, vol. 1, no. 3, pp. 133–143, 2019.

[5] A. Haydari and Y. Yilmaz, "Deep reinforcement learning for intelligent transportation systems: A survey," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 1, pp. 11–32, Jan. 2022.

[6] L. Ding, Q.-L. Han, X. Ge, and X.-M. Zhang, "An overview of recent advances in event-triggered consensus of multiagent systems," *IEEE Trans. Cybern.*, vol. 48, no. 4, pp. 1110–1123, Apr. 2018.

[7] T. T. Nguyen, N. D. Nguyen, and S. Nahavandi, "Deep reinforcement learning for multiagent systems: A review of challenges, solutions, and applications," *IEEE Trans. Cybern.*, vol. 50, no. 9, pp. 3826–3839, Sep. 2020.

[8] J. Hao et al., "Exploration in deep reinforcement learning: From single-agent to multiagent domain," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Jan. 19, 2023, doi: 10.1109/TNNLS.2023.3236361.

[9] Y. Du, L. Han, M. Fang, J. Liu, T. Dai, and D. Tao, "LIIR: Learning individual intrinsic reward in multi-agent reinforcement learning," in *Proc. 33rd Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 4405–4416.

[10] A. Mahajan, T. Rashid, M. Samvelyan, and S. Whiteson, "MAVEN: Multi-agent variational exploration," in *Proc. 33rd Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2019, pp. 7611–7622.

[11] T. Wang, J. Wang, Y. Wu, and C. Zhang, "Influence-based multi-agent exploration," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–19.

[12] M. Pislar, D. Szepesvari, G. Ostrovski, D. L. Borsa, and T. Schaul, "When should agents explore?" in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2022, pp. 1–22.

[13] P. Auer, T. Jaksch, and R. Ortner, "Near-optimal regret bounds for reinforcement learning," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2009, pp. 89–96.

[14] C. Jin, Z. Allen-Zhu, S. Bubeck, and M. I. Jordan, "Is Q-learning provably efficient?" in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2018, pp. 4868–4878.

[15] M. J. A. Strens, "A Bayesian framework for reinforcement learning," in *Proc. 17th Int. Conf. Mach. Learn. (ICML)*, 2000, pp. 943–950.

[16] S. Agrawal and R. Jia, "Optimistic posterior sampling for reinforcement learning: Worst-case regret bounds," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 1184–1194.

[17] I. Osband, D. Russo, and B. V. Roy, "(More) efficient reinforcement learning via posterior sampling," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2013, pp. 3003–3011.

[18] A. L. Strehl and M. L. Littman, "An analysis of model-based interval estimation for Markov decision processes," *J. Comput. Syst. Sci.*, vol. 74, no. 8, pp. 1309–1331, Dec. 2008.

[19] R. Devidze, P. Kamalaruban, and A. Singla, "Exploration-guided reward shaping for reinforcement learning under sparse rewards," in *Proc. 36th Conf. Adv. Neural Inf. Process. Syst. (NeurIPS)*, vol. 35, 2022, pp. 5829–5842.

[20] M. Bellemare, S. Srinivasan, G. Ostrovski, T. Schaul, D. Saxton, and R. Munos, "Unifying count-based exploration and intrinsic motivation," in *Proc. 30th Conf. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2016, pp. 1–19.

[21] G. Ostrovski, M. G. Bellemare, A. Oord, and R. Munos, "Count-based exploration with neural density models," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1–14.

[22] S. Lobel, A. Bagaria, and G. Konidaris, "Flipping coins to estimate Pseudocounts for exploration in reinforcement learning," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2023, pp. 1–19.

[23] D. Pathak, P. Agrawal, A. A. Efros, and T. Darrell, "Curiosity-driven exploration by self-supervised prediction," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2017, pp. 1–11.

[24] Y. Burda, H. Edwards, A. Storkey, and O. Klimov, "Exploration by random network distillation," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–16.

[25] M. Fortunato et al., "Noisy networks for exploration," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2018, pp. 1–20.

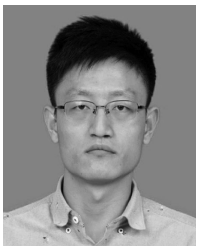
[26] D. A. Martínez, E. Mojica-Nava, K. Watson, and T. Usländer, "Multiagent self-redundancy identification and tuned greedy-exploration," *IEEE Trans. Cybern.*, vol. 52, no. 7, pp. 5744–5755, Jul. 2022.

[27] O. Tutsoy, D. E. Barkana, and K. Balikci, "A novel exploration-exploitation-based adaptive law for intelligent model-free control approaches," *IEEE Trans. Cybern.*, vol. 53, no. 1, pp. 329–337, Jan. 2023.

[28] H. Kim, J. Kim, Y. Jeong, S. Levine, and H. O. Song, "EMI: Exploration with mutual information," in *Proc. 36th Int. Conf. Mach. Learn. (ICML)*, vol. 97, 2019, pp. 3360–3369.
[29] M. Fellows, A. Mahajan, T. G. J. Rudner, and S. Whiteson, "VIREL: A variational inference framework for reinforcement learning," in *Proc. 33rd Conf. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020, pp. 7120–7134.
[30] R. Houthooft, X. Chen, Y. Duan, J. Schulman, F. D. Turck, and P. Abbeel, "VIME: Variational information maximizing exploration," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2017, pp. 1109–1117.
[31] X. Wang, T. Li, Y. Cheng, and C. P. Chen, "Inference-based posteriori parameter distribution optimization," *IEEE Trans. Cybern.*, vol. 52, no. 5, pp. 3006–3017, May 2022.
[32] C. Bai et al., "Variational dynamic for self-supervised exploration in deep reinforcement learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 34, no. 8, pp. 4776–4790, Aug. 2023.
[33] W.-C. Jiang, V. Narayanan, and J.-S. Li, "Model learning and knowledge sharing for cooperative multiagent systems in stochastic environment," *IEEE Trans. Cybern.*, vol. 51, no. 12, pp. 5717–5727, Dec. 2021.
[34] J. Kim, "Cooperative exploration and networking while preserving collision avoidance," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4038–4048, Dec. 2017.
[35] W. Dabney, G. Ostrovski, and A. Barreto, "Temporally-extended ε-greedy exploration," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2021, pp. 1–19.
[36] A. P. Badia et al., "Never give up: Learning directed exploration strategies," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2020, pp. 1–27.
[37] M. Toussaint, "Robot trajectory optimization using approximate inference," in *Proc. Int. Conf. Mach. Learn. (ICML)*, vol. 382, 2009, pp. 1049–1056.
[38] S. Levine, "Reinforcement learning and control as probabilistic inference: Tutorial and review," 2018, *arXiv:1805.00909*.
[39] A. Xie, J. Harrison, and C. Finn, "Deep reinforcement learning amidst lifelong non-stationarity," in *Proc. Lifelong Mach. Learn. Workshop Int. Conf. Mach. Learn. (ICML)*, 2020, pp. 1–14.
[40] Y. Wen, Y. Yang, R. Luo, J. Wang, and W. Pan, "Probabilistic recursive reasoning for multi-agent reinforcement learning," in *Proc. Int. Conf. Learn. Represent. (ICLR)*, 2019, pp. 1–19.
[41] L. P. Kaelbling, M. L. Littman, and A. R. Cassandra, "Planning and acting in partially observable stochastic domains," *Artif. Intell.*, vol. 101, nos. 1–2, pp. 99–134, May 1998.
[42] E. A. Hansen, D. S. Bernstein, and S. Zilberstein, "Dynamic programming for partially observable stochastic games," in *Proc. 19th Nat. Conf. Artif. Intell. (AAAI)*, vol. 4, 2004, pp. 709–715.
[43] T. Haarnoja, A. Zhou, P. Abbeel, and S. Levine, "Soft actor–critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor," in *Proc. Int. Conf. Mach. Learn. (ICML)*, 2018, pp. 1861–1870.
[44] R. Lowe, Y. Wu, A. Tamar, J. Harb, P. Abbeel, and I. Mordatch, "Multi-agent actor–critic for mixed cooperative-competitive environments," in *Proc. Adv. Neural Inf. Process. Syst. (NeurIPS)*, 2020, pp. 6379–6390.
[45] L. Zheng, J. Yang, H. Cai, W. Z. Hang, J. Wang, and Y. Yu "Magent: A many-agent reinforcement learning platform for artificial collective intelligence," in *Proc. AAAI Conf. Artif. Intell. (AAAI)*, vol. 32, 2018, pp. 8222–8223.

Shaokang Dong received the B.S. degree in advanced class from the Huazhong University of Science and Technology, Wuhan, China, in 2018. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Technology, Nanjing University, Nanjing, China.

His research interests include reinforcement learning and multiagent systems.



Hangyu Mao received the Ph.D. degree from Peking University, Beijing, China, in July 2020.

He is currently a Researcher with the SenseTime Research, Beijing. His research interests include multiagent system, reinforcement learning, large language model, and general intelligent decision-making techniques.

Shangdong Yang received the B.S. degree in automation from Southwest Jiaotong University, Chengdu, China, in 2013, and the Ph.D. degree in computer science and technology from Nanjing University, Nanjing, China, in 2020.

He is currently an Assistant Professor with the School of Computer Science, Nanjing University of Posts and Telecommunications, Nanjing, and also a Project Moderator with Guangxi Key Lab of Multi-Source Information Mining and Security, Guangxi Normal University, Guilin, China. His research interests include reinforcement learning and multiagent systems.

Shengyu Zhu received the B.E. degree in electrical engineering from the Beijing Institute of Technology, Beijing, China, in 2010, and the M.S. degree in mathematics and the Ph.D. degree in electrical and computer engineering from Syracuse University, Syracuse, NY, USA, in 2016 and 2017, respectively.

He was with Syracuse University in 2012. He was a Principal Researcher with Huawei Noah's Ark Lab, Huawei Technologies, Beijing. His research interests include causality, machine learning, information theory, and statistical signal processing learning.

Wenbin Li received the Ph.D. degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2019.

He is currently an Assistant Researcher with the Department of Computer Science and Technology, Nanjing University. His research interests include machine learning and computer vision, particularly in metric learning, few-shot learning, and their applications to image classification and image generation.

Jianye Hao (Member, IEEE) received the Ph.D. degree from the Department of Computer Science and Engineering, The Chinese University of Hong Kong, Hong Kong, China, in 2013.

He is currently an Associate Professor with Tianjin University, Tianjin, China, and the Director of the Noah's Ark Decision-making and Reasoning Laboratory, Huawei Technologies, Beijing, China. The research of his team has been successfully applied in domains, such as game artificial intelligence, E-commerce recommendation, network optimization, and supply chain optimization. His research areas focus on reinforcement learning and multiagent systems.

Prof. Hao has received a number of best paper awards, such as ASE2019 and CoRL2020.

Yang Gao (Senior Member, IEEE) received the Ph.D. degree from the Department of Computer Science and Technology, Nanjing University, Nanjing, China, in 2000.

He is currently a Professor with the Department of Computer Science and Technology, Nanjing University. He has authored more than 100 papers in top conferences and journals in and outside of China. His current research interests include artificial intelligence and machine learning.