Unveiling Decision Intention for Cooperative Multi-Agent Reinforcement Learning

Zeren Zhang
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of AutomationChinese Academy of Sciences
Beijing, China
School of Artificial Intelligence, University of Chinese Academy of Sciences
Beijing, China
zhangzeren2021@ia.ac.cn

Zhiwei Xu
School of Artificial Intelligence, Shandong University
Jinan, Shandong, China
zhiwei_xu@sdu.edu.cn

Guangchong Zhou
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of AutomationChinese Academy of Sciences
Beijing, China
School of Artificial Intelligence, University of Chinese Academy of Sciences
Beijing, China
zhouguangchong2021@ia.ac.cn

Dapeng Li
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of AutomationChinese Academy of Sciences
Beijing, China
School of Artificial Intelligence, University of Chinese Academy of Sciences
Beijing, China
lidapeng2020@ia.ac.cn

Bin Zhang
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of AutomationChinese Academy of Sciences
Beijing, China
School of Artificial Intelligence, University of Chinese Academy of Sciences
Beijing, China
zhangbin2020@ia.ac.cn

Guoliang Fan
The Key Laboratory of Cognition and Decision Intelligence for Complex Systems, Institute of AutomationChinese Academy of Sciences
Beijing, China
guoliang.fan@ia.ac.cn

ABSTRACT

For cooperative multi-agent reinforcement learning, various methods have been proposed to enhance the collaborative strategy capabilities of agents. However, when agents make decisions, humans have no knowledge of their subsequent decision-making intentions or sub-goals. This lack of understanding hinders human comprehension of agent strategies and further research on agents. Currently, there are limited relevant studies. To address this problem, we propose a novel framework which can generate the decision intention of agents. We first formalize this problem and use states crucial to the task to express the decision intentions of agents. Then, we introduce the polarization index to measure the importance of states and select them for training. Finally, we learn the decision intentions through a diffusion model with rapid generation capability and generate them during the decision-making process. This study sheds light on the problem of agent decision intention and enhances the transparency of agent strategies, facilitating deeper research on agents. The experimental results demonstrate the effectiveness of our approach.

KEYWORDS

Diffusion model; Multi-agent reinforcement learning; Decision intention

ACM Reference Format:
Zeren Zhang, Zhiwei Xu, Guangchong Zhou, Dapeng Li, Bin Zhang, and Guoliang Fan. 2025. Unveiling Decision Intention for Cooperative Multi-Agent Reinforcement Learning. In *Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Detroit, Michigan, USA, May 19 – 23, 2025*, IFAAMAS, 10 pages.

1 INTRODUCTION

In recent years, multi-agent reinforcement learning (MARL) has made remarkable advancements and drawn widespread attention across various domains including game AI [2, 34], robots [22], and traffic control [41]. In contrast to single-agent tasks, multi-agent tasks pose additional challenges. For instance, agents in multi-agent settings often face the problem of limited observability, perceiving only local information of the environment. To mitigate the negative impact of these challenges, many studies follow the paradigm of Centralized Training with Decentralized Execution (CTDE) [17], where agents receive global information only during the training

Proc. of the 24th International Conference on Autonomous Agents and Multiagent Systems (AAMAS 2025), Y. Vorobeychik, S. Das, A. Nowé (eds.), May 19 – 23, 2025, Detroit, Michigan, USA. © 2025 International Foundation for Autonomous Agents and Multiagent Systems (www.ifaamas.org).

phase while relying on their local observations during execution. Such methods can be broadly categorized into algorithms based on Actor-Critic (AC) architecture [8, 40] and those based on value decomposition [24, 32]. In MARL, a key research focus lies in fully cooperative tasks, where all agents are required to cooperate to achieve specified objectives.

The majority of these studies are centered on enhancing the decision-making capabilities of agents. However, the behaviors of agents are unpredictable from the human perspective, leaving people in the dark about the agents' future decisions. This uncertainty hinders human comprehension of agent strategies and further research on agents. In this paper, intention is not used as a technical term but refers to certain information that promotes understanding agents' future decision-making behaviors. Some existing studies address intention or goal recognition problems [1, 5], but they focus on classical planning, and assume a known set of possible goals or even domain knowledge of the state transition function [19], which makes them unsuitable for multi-agent reinforcement learning frameworks. Currently, there has been limited research on this issue. Some studies indirectly related to this problem aim to enhance the decision-making performance of agents by inferring the current global state information [38, 42] or forecasting the subsequent state information [39]. However, these works suffer from several limitations. Firstly, they infer implicit embeddings that cannot be understood by humans. Secondly, they can only infer features for the current or the next few time steps.

By leveraging diverse datasets, generative models have made substantial strides in the domains of vision and language [3, 23, 25, 26]. Some studies utilize diffusion models [13, 20, 29] to predict future events in scenarios such as traffic or basketball games [9, 11, 18]. These models typically forecast multiple possible motion trajectories for the next few time steps to approximate the actual future trajectories. They rely exclusively on external information and cannot leverage the intrinsic information of individuals (e.g., decision strategies, personal considerations), which often plays a pivotal role in shaping future events.

Inspired by the aforementioned studies, we aim to unveil decision intentions within the context of MARL, which is able to utilize the internal information of agents and generate content that encompasses a broader range of state information compared to motion trajectories. We represent the decision intentions or sub-goals in the form of global states, making the agents' future decisions more understandable and predictable for humans. Since decision intentions are not limited to the next few time steps, we aim to overcome this limitation. In this paper, we formalize the intentions of agents and propose the polarization index based on action-value function, a mechanism to identify appropriate states for expressing the intentions. Then we introduce a dynamic weighting method to achieve step-by-step intention generation. Finally, we propose a novel framework for Unveiling Decision Intention (UDI) In Cooperative Multi-Agent Reinforcement Learning. UDI is based on a diffusion model with rapid generation capability to infer the decision intentions of agents utilizing their internal information.

We demonstrated the effectiveness of polarization index and UDI through an intuitive grid environment. Additionally, we tested the performance of our method on complex tasks under the widely applicable SMAC environment. Since there is currently limited relevant research, our method sheds light on the issue of agent decision intention, enhances the transparency of agent strategies, and facilitates deeper research into agents.

2 RELATED WORK

There are existing studies on intention or goal recognition that address this problem in both single-agent and multi-agent tasks. Some approaches [1, 5] measure the sequence of observations against a set of Q-functions using a distance metric to solve the goal recognition task. However, they focus on classical planning, and assume a predefined set of possible goals or even prior domain knowledge of the state transition function [19], which makes them unsuitable for multi-agent reinforcement learning frameworks. some other related methods are not fully applicable to this setting, such as those designed for scenarios with only one or two agents [10, 35], fully observable [33], and stationary environments [12], or methods that rely on a predefined set of generated content [6].

In the field of MARL, some studies focus on enabling agents to infer global state information for the current time step or the next few time steps under partial observability conditions. For instance, CTDS [42] and PTDE [4] employ knowledge distillation techniques. During training, they utilize global state information and perform distillation. During execution, agents infer global state information under local observation conditions. MBVD [39], as a model-based method, constructs an imagination module through variational inference, which is then utilized to infer information for the next several time steps. MASER [16], as a goal-based method, generates subgoals for agents from the experience replay buffer by considering Q-values. However, these methods acquire global state information implicitly or through embeddings, precluding the interpretability of their behaviors by humans. In addition, the primary objective of these studies is to enhance the strategic capabilities of agents, and they cannot predict future states and make them understandable to humans. The recognition of agents' decision intentions or sub-goals, which enables humans to understand their future decisions, has not yet been addressed in existing research.

Currently, some studies utilize the diffusion model to predict the trajectories of pedestrians or players in scenarios such as traffic or basketball games. Many methods formulate trajectory forecasting as a sequential prediction problem , and focus on modeling this social interaction. Some methods apply Generative Adversarial Network (GAN) structures to capture multi-modality and generate future trajectory distributions [7, 31]. Another approaches adopt encoder-decoder architectures, leveraging VAE-based techniques to learn the distribution through variational inference [15, 37]. Inspired by non-equilibrium thermodynamics, diffusion models possess strong feature representation capabilities and demonstrate a good fit for complex data distributions. Some approaches have applied these models to motion trajectory prediction problems, achieving better performance compared to the aforementioned methods [9, 11, 18]. The common points of these studies are: 1) They only predict the motion trajectories, and just for the next several time steps. 2) They generate multiple trajectories and minimize the minimum discrepancy between these trajectories and the future ground truth. 3) They only rely on external information for prediction. Given the reasons above, directly applying the aforementioned methods to the

MARL framework is not feasible. However, we can draw inspiration from the proven capabilities of diffusion models to design a novel framework.

3 PRELIMINARIES

3.1 Decentralized Partially Observable Markov Decision Process (Dec-POMDP)

The multi-agent collaborative sequential decision-making problem can be modeled by the decentralized partially observable Markov decision process (Dec-POMDP). The Dec-POMDP can be defined as a tuple $\mathcal{G} = \langle \mathcal{A}, \mathcal{S}, \mathcal{U}, \mathcal{Z}, \mathcal{P}, O, r, \gamma \rangle$ [21]. $a \in \mathcal{A} := \{1, 2, \ldots, n\}$ denotes the set of agents. $s \in \mathcal{S}$ denotes the global state of the environment. $u^a \in \mathcal{U}$ denotes the action of each agent, and $\boldsymbol{u} \in \boldsymbol{\mathcal{U}} \equiv \mathcal{U}^n$ denotes the joint action. $\mathcal{P} : \mathcal{S} \times \boldsymbol{\mathcal{U}} \times \mathcal{S} \to [0, 1]$ denotes the state transition function. Each agent receives its local observation $z^a \in \mathcal{Z}$ provided by the observation function $O : \mathcal{S} \times \mathcal{A} \to \mathcal{Z}$. $r : \mathcal{S} \times \boldsymbol{\mathcal{U}} \to \mathbb{R}$ denotes the shared reward function, and γ is the discount factor. The goal of all agents is to maximize the discount return $R_t = \sum_{m=0}^{\infty} \gamma^m r_{t+m}$.

Current methods typically enable agents to make decisions based on their historical trajectories rather than current local observations to mitigate the negative impact of partial observability. In value-based approaches, the joint state value function and state-action value function are defined as:

$$V_{tot}^{\boldsymbol{\pi}}(s) = \mathbb{E}_{\boldsymbol{\pi}}[R_t|s] \quad Q_{tot}^{\boldsymbol{\pi}}(s, \boldsymbol{u}) = \mathbb{E}_{\boldsymbol{\pi}}[R_t|s, \boldsymbol{u}]$$

where $\boldsymbol{\pi} = \{\pi_1, \pi_2, \ldots, \pi_n\}$ denotes the joint strategies of agents.

3.2 Diffusion Models

As a generative model, a diffusion model [13, 28, 30] consists of a pre-defined forward noising process and a trainable denoising process. For $x^0 \sim q(x)$, the forward noising process fixed to a Markov chain adds Gaussian noise step by step to generate $\{x^1, x^2, \ldots, x^K\}$ by $q(x^{k+1}|x^k) := \mathcal{N}(x^{k+1}; \sqrt{\alpha_{k+1}}x^k, (1 - \alpha_{k+1})\boldsymbol{I})$, where $\mathcal{N}(\mu, \Sigma)$ denotes a Gaussian distribution with mean μ and variance Σ, $\alpha_k \in \mathbb{R}$ is a pre-defined variance schedule. When K is sufficiently large, x^K can be regarded as following an isotropic Gaussian distribution. We use the superscript k to denote the diffusion step, distinguishing it from the time step in RL. During the denoising process, an initial noise $x^K \sim \mathcal{N}(0, \boldsymbol{I})$ is sampled, and progressively denoised by $p_\theta(x^{k-1}|x^k) := \mathcal{N}(x^{k-1}; \mu_\theta(x^k, k), \Sigma_k)$. The diffusion models can be trained by maximizing the evidence lower bound (ELBO): $\mathbb{E}_{x^0}[\log p_\theta(x^0)] \geq \mathbb{E}_q\left[\log \frac{p_\theta(x^{0:K})}{q(x^{1:K}|x^0)}\right]$, or optimizing a simplified surrogate loss [30]:

$$\mathcal{L}(\theta) = \mathbb{E}_{k\sim[1,K], x^0\sim q, \epsilon\sim\mathcal{N}(0,\boldsymbol{I})}\left[\left\|\epsilon - \epsilon_\theta(x^k, k)\right\|^2\right] \tag{1}$$

where $\epsilon_\theta(x^k, k)$ is the predicted noise parameterized through a deep neural network, denoting the noise added to the x^0 to produce x^k. The mean $\mu_\theta(x^k, k)$ can be directly obtained from $\epsilon_\theta(x^k, k)$ by:

$$\mu_\theta(x^k, k) = \frac{1}{\sqrt{\alpha_k}}x^k - \frac{1 - \alpha_k}{\sqrt{1 - \bar{\alpha}_k}\sqrt{\alpha_k}}\epsilon_\theta(x^k, k) \tag{2}$$

where $\bar{\alpha}_k = (\alpha_k \cdots \alpha_1)^2$.

However, the Markov assumption leads to a generation process requiring K steps, resulting in significant computational costs. DDIM [29] removes the Markov assumption, significantly reducing the number of generation steps while keeping the training process unchanged. The generation process of DDIM is as follows:

$$\begin{aligned} x^{prev} =& \sqrt{\bar{\alpha}_{prev}}\left(\frac{x^k - \sqrt{1 - \bar{\alpha}_k}\epsilon_\theta(x^k, k)}{\sqrt{\bar{\alpha}_k}}\right) \\ &+ \sqrt{1 - \bar{\alpha}_{prev} - \sigma_k^2}\epsilon_\theta(x^k, k) + \sigma_k^2\epsilon \end{aligned} \tag{3}$$

where x^{prev} and x^k can have intervals of multiple steps between them, indicating that the denoising process does not need to be performed step by step. σ_k is manually specified. When $\sigma_k = 0$, the generated results are deterministic, eliminating the influence of randomness.

3.3 Guided Diffusion

If it's necessary to generate different data under various conditions using diffusion models, one approach is classifier-based guidance [20]. It trains an additional classifier $p_\phi(y|x^k)$ on noisy samples, and the generation process and be guided with the gradients from the classifier. Another more flexible method is classifier-free guidance [14] without the extra classifier. It learns both a conditional $\epsilon_\theta(x^k, y, k)$ and an unconditional $\epsilon_\theta(x^k, k)$ model. The perturbed noise $\epsilon_\theta(x^k, k) + \omega(\epsilon_\theta(x^k, y, k) - \epsilon_\theta(x^k, k))$ is used to guide the generation process, where ω is referred to as the guidance scale.

4 METHOD

In this section, we introduce our method for determining agent decision intentions or sub-goals and the UDI framework for generating these intentions. Firstly, we provide our formalization of the agent decision intention. We represent the decision intentions in the form of global states, making the agents' future decisions more understandable and predictable for humans. Subsequently, we introduced a quantitative metric, namely polarization index, to measure and extract the representative states for training. Finally, we elaborated on our intention generation model. The UDI framework is depicted in Figure 1.

4.1 Formalization of Agent Decision Intention

When aiming to explicitly demonstrate the decision intention of an agent, it is necessary to first mathematically characterize it and then quantitatively measure it. In this paper, intention is not used as a technical term but refers to certain information that promotes understanding agents' future decision-making behaviors. The intentions of humans in decision-making can be interpreted as the intermediate objectives or sub-goals that need to be achieved in order to accomplish the ultimate task. It represents the critical commonalities across trajectories essential for task completion. We apply this idea to agents and define the decision intention of agent a in time step t as a function of the current state s_t and its policy π_a. Mathematically, this can be expressed as $I(s_t, \pi_a)$, capturing the underlying objectives or sub=goals that the agent seeks to achieve.

To make the decision intention understandable to humans, for each time step, we adopt the most crucial state for task completion in the agent's decision-making process after the current time step to characterize the agent's decision intention. We refer to this state as *decisive state* s_{t_D}. Decisive states obtained at all time steps

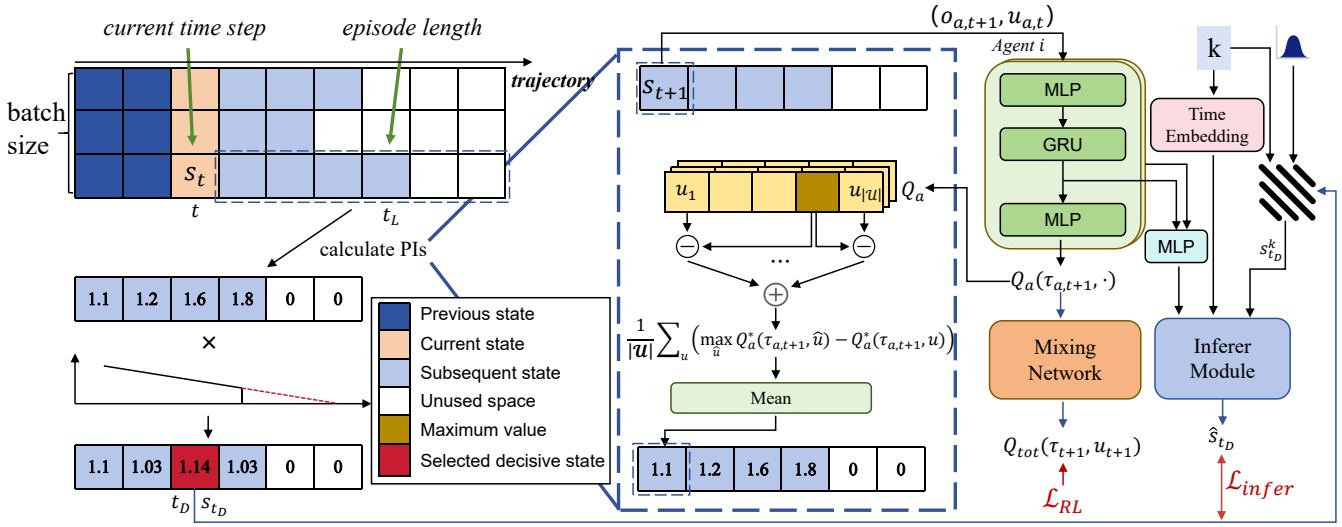


Figure 1: The workflow of UDI. Left: The process of obtaining a decisive state from the complete trajectories of agents. Middle: The computation process of the polarization index of a state in the multi-agent case. Right: The overall architecture of the agent model, including the policy generation part and the decision intention generation part.

form a sequence that illustrates the agent's step-by-step process in achieving the final task. That is:

$$I(s_t, \pi_a) = s_{t_D}, \quad \text{subject to} \quad t_D > t \tag{4}$$

Then we need to quantitatively measure the importance of the states appearing in the decision trajectories. A straightforward approach is to determine state importance based on the rewards obtained by the agent in those states. However, a hand-designed reward function is insufficient to cover all aspects necessary for task completion. The implementation of the aforementioned approach may result in the omission of critical states essential for the successful execution of the task, due to their lack of immediate reward. We attempt to devise an intention-measuring mechanism that can identify crucial states, even if the agent may not receive rewards in those states.

At time step t, the importance of the state s_t can be understood as the degree to which deviating from the optimal action at s_t affects the final task. Since the agent's state value reflects the expected return in the future, at this time step, we perturb the original action and define the importance of state s_t as the discrepancy between its state value under the optimal policy when taking random actions only at this time step, and the state value under the optimal action. We refer to this metric as *polarization index*.

Definition 1. *(Polarization Index) For the agent in a task, assuming its optimal policy is known as π^* and its optimal state value function is $V^*(s)$. For any time step t and corresponding state s_t in the agent's decision trajectories, suppose the agent switches to taking random policy only at this time step, resulting in a policy $\tilde{\pi}$ and its corresponding state value function $V^{\tilde{\pi}}(s)$. Then, the polarization index of s_t is given by:*

$$\begin{aligned} PI(s_t) &= V^*(s_t) - V^{\tilde{\pi}}(s_t) \\ &= \frac{1}{|\mathcal{U}|} \sum_u \left(\max_{\hat{u}} Q^*(s_t, \hat{u}) - Q^*(s_t, u) \right) \end{aligned} \tag{5}$$

For detailed derivation, please refer to supplementary materials. When $PI(s_t) = 0$, it implies that at s_t, random actions have no effect on completing the task. In this case, s_t is not considered crucial for task completion. When $Q^*(s_t, u) = 0$ for any u that is not the optimal action, $PI(s_t)$ reaches its maximum, indicating that only the optimal action can accomplish the task, and other actions result in no rewards thereafter. Hence s_t is crucial for task completion and can be seen as a decisive state.

In multi-agent tasks, the above definition can be extended with the joint action, whose space grows exponentially with the increase in the number of agents. Previous research indicates that value decomposition method can achieve implicit credit assignment [43]. Although individual Q-functions are not computed directly, once agents have learned an accurate and effective policy, they can, to some extent, reflect the return benefits of different actions. On the other hand, it has been demonstrated that, these Q-functions can be approximately interpreted as reflecting the evaluation of returns, and thus can be used directly as value functions [16]. It should be noted that PI is also derived based on relative relationships. Therefore, the absolute values of individual Q-functions are not important. As long as they can maintain the same relative relationships among different actions, the calculation results of PI will remain unaffected. We compute PI separately for each agent's individual Q-value, which allows us to assess the importance of states from different agents' perspectives. Then we obtain an overall evaluation by averaging. This approach ensures that the computational

complexity increases linearly with the number of agents. That is:

$$PI(s_t) = \frac{1}{n}\sum_{a}^{n}\frac{1}{|\mathcal{U}|}\sum_{u}\left(\max_{\hat{u}} Q_a^*(\tau_{a,t},\hat{u}) - Q_a^*(\tau_{a,t},u)\right) \quad (6)$$

where $\tau_{a,t}$ represents the historical trajectory of agent a at time step t. The computation process is illustrated in the middle part of Figure 1. When the state with the highest polarization index is located at the end of the decision trajectory, it presents challenges. The time step corresponding to the state is too far away from the current time step, weakening their correlation. This makes prediction extremely challenging and impedes the comprehensibility of intentions. To establish step-by-step predictions, it is essential to strike a balance between the magnitude of the polarization index and the temporal distance between the predicted state and the current state.

One approach is to calculate the polarization index for each state in the trajectory and then apply linear decay weights from one to zero for each state based on the length of each episode. However, this approach leads to increasingly severe myopia as the number of time steps increases, with its choices progressively favoring states closer to the current time step (even only the next state). This is due to the increasing dominance of the decaying portion (see supplementary materials for details). We adopt a dynamic weighting method where, at time step t, the weights of states occurring after that time step s_{t+i} decay from 1. That is, $w_{t+i} = 1 - \frac{i-1}{t_L}$, where t_L is the length of this episode. In conclusion, at time step t, the decisive state s_{t_D} can be obtained by:

$$\begin{aligned} t_D &= \arg\max_{\tilde{t}>t} w_{\tilde{t}} PI(s_{\tilde{t}}) \\ &= \arg\max_{\tilde{t}>t}(1-\frac{\tilde{t}-t-1}{t_L})\frac{1}{n}\sum_{a}^{n}\frac{1}{|\mathcal{U}|}\sum_{u}\left(\max_{\hat{u}} Q_a^*(\cdot,\hat{u}) - Q_a^*(\cdot,u)\right) \end{aligned} \quad (7)$$

The complete process of obtaining decisive states from the agents' historical trajectories is depicted on the left part of Figure 1.

4.2 Decision Intention Generation

Benefiting from the powerful generative capacity of the diffusion model, we formulate the decision intention generation process as a standard problem of conditional generative modeling:

$$\max_{\theta} \mathbb{E}_{\tau\sim\mathcal{D}}[\log p_\theta(s_{t_D}|y(\cdot))] \quad (8)$$

where $\mathcal{D}$ denotes the dataset, $y(\cdot)$ represents additional information required for generating the decision intention. As the agent's policy is stored in the form of network parameters, we feed the current local observation and the action input from the previous time step into its individual network and use the hidden variables $\boldsymbol{h}_t = \{h_t^1, h_t^2, \ldots, h_t^n\}$ processed through GRU as conditional variables. This variable encompasses both the policy information and the state information of the agent.

To reduce computational cost, we adopt DDIM to fit $I(s_t, \pi_a)$ and generate s_{t_D}. An additional benefit of this approach is that the generation results of DDIM are deterministic, thus avoiding the influence of randomness on the generated decision intentions. We construct our generative model according to the conditional diffusion process:

$$q\left(s_{t_D}^{k+1}|s_{t_D}^{k}\right), \quad p_\theta\left(s_{t_D}^{prev}|s_{t_D}^{k}, \boldsymbol{h}_t\right) \quad (9)$$

Based on this, we set a decision intention generation part additional to the original agent policy architecture, which includes an inferer model and other modules for processing various inputs. The overall structure is depicted in the right part of Figure 1. We need to develop the model's capability to separate the noise in $s_{t_D}^k$ based on conditional variables. Using either ϵ or $s_{t_D}^0$ as the learning objective can enable the model to acquire this capability. However, through experimental comparison (see Section 5.3), we found that using $s_{t_D}^0$ as the learning objective yields better results than using ϵ for this task. To achieve conditional diffusion process, one approach could be the classifier-based guidance which needs an additional classifier. In contrast, we adopt the more flexible classifier-free guidance approach without the extra classifier. To implement classifier-free guidance, the decision intention is generated by starting with Gaussian noise $s_{t_D}^K$ and denoising $s_{t_D}^k$ into $s_{t_D}^{prev}$ with the model's output $f_\theta(\cdot)$:

$$\hat{s}_{t_D}^0 = f_\theta(x^k, k) + \omega(f_\theta(x^k, y, k) - f_\theta(x^k, k)) \quad (10)$$

We set ω to 1, eliminating the need to learn the generation process without conditional guidance and reducing the cost of model training. The input of inferer model consists of three parts. The first part is the hidden variable that contains both the agent policy information and state information. After obtaining h_t^a from each agent, it is encoded and integrated through an embedding layer composed of an MLP layer and inputted into the inferer module. The second part generates positional encodings based on the noise injection step k, which then undergoes processing through two MLP layers, as in [36]. The third part consists of the noisy decisive state, which is denoised through the inferer module for the training or generation of agent decision intention.

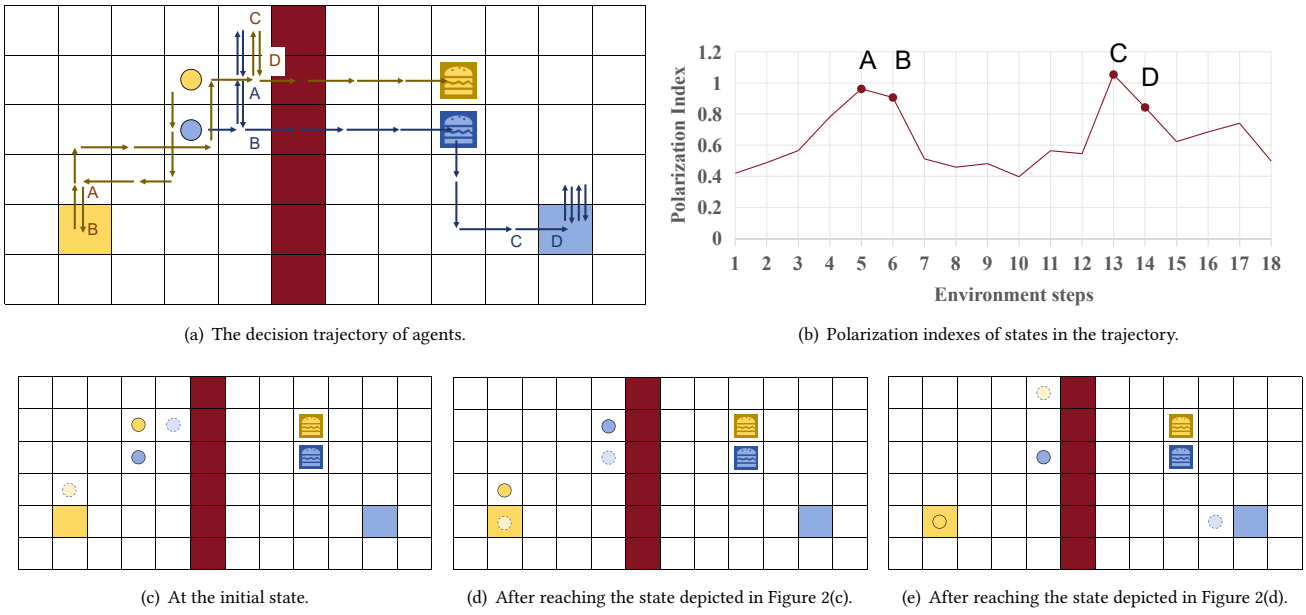
During the training phase, at each time step, we obtain s_{t_D} for every complete trajectory in the batch according to Equation 7, then add random noise to it by:

$$s_{t_D}^k = \sqrt{\bar{\alpha}_k} s_{t_D}^0 + \sqrt{1-\bar{\alpha}_k}\epsilon \quad (11)$$

For the model to accurately predict the decision intention, it is essential for the agents to have an accurate estimation of state-action values and to have acquired a well-performing policy. During the initial stages when the agent has not yet learned a satisfactory policy, the estimation of state-action values is inaccurate. This may pose challenges in obtaining decisive states. We prevent this negative impact by controlling the inferer loss with a coefficient λ that linearly increases from 0 to 1 as training progresses. The training objective of the model is as follows:

$$\begin{aligned} \mathcal{L} &= \mathcal{L}_{RL} + \lambda\mathcal{L}_{infer} \\ &= \left(y^{tot} - Q_{tot}(\boldsymbol{\tau}_t, \boldsymbol{u}_t, s_t; \psi)\right)^2 + \\ &\quad \lambda\mathbb{E}_{k,s_{t_D},\epsilon}\left[\left\|f_\theta(x^k, y, k) - s_{t_D}^0\right\|_2^2\right] \end{aligned} \quad (12)$$

where $y^{tot} = r_t + \gamma \max_{\boldsymbol{u}_{t+1}} Q_{tot}(\boldsymbol{\tau}_{t+1}, \boldsymbol{u}_{t+1}, s_{t+1}; \psi^-)$, and ψ^- represents the parameters of the target network. During the execution phase, at each time step, Gaussian noise is sampled and gradually denoised into the decision intention of agents with the assistance



(a) The decision trajectory of agents.

(b) Polarization indexes of states in the trajectory.

(c) At the initial state.

(d) After reaching the state depicted in Figure 2(c).

(e) After reaching the state depicted in Figure 2(d).

Figure 2: Intuitive performance evaluations of UDI in Barrier Pass environment.

of the inferer module and conditional variables from the agent networks. The denoising process can be formulated as:

$$s_{t_D}^{prev} = \sqrt{\bar{\alpha}_{prev}} f_\theta(x^k, y, k) + \sqrt{1-\bar{\alpha}_{prev}} \frac{s_{t_D}^k - \sqrt{\bar{\alpha}_k} f_\theta(x^k, y, k)}{\sqrt{1-\bar{\alpha}_k}} \tag{13}$$

Details can be found in supplementary materials.

5 EXPERIMENTS

In this section, we first explore the following questions through an intuitive grid environment:

- Can our intention measurement mechanism, polarization index, correctly identify crucial states for task completion, in the absence of rewards at those states?
- Are the future states generated by UDI sufficiently close to the real states?
- Are the generated states sufficiently important?

Next, we comprehensively test the accuracy and importance of the generated states in complex tasks under SMAC [27]. Finally, we analyze the impact of denoising steps and diffusion model learning objectives on the algorithm's performance through ablation studies. The strategy learning of the agents is based on QMIX. We conduct experiments in SMAC with 5 seeds. We conducted both quantitative and qualitative analyses on SMAC to provide as broad an evaluation of our method as possible. The details of all environments and the implementation are provided in supplementary materials.

5.1 Case Study

We demonstrate the performance of our algorithm through a grid environment, providing an intuitive representation. We design *Barrier Pass* shown in Figure 2(a). Two agents (represented by yellow and blue circles) need to eat the designated food (indicated by hamburgers matching the colors of the agents) with the intervening wall (depicted in brown) blocking their paths. Additionally, floor mechanisms (located at the bottom left and bottom right) control the descent of the wall. The wall descends only when an agent stands on the floor mechanism matching its color, allowing passage through the center. When no floor mechanism is triggered, the wall **ascends**, blocking passage through the center. The key to completing this cooperative task lies in the agents triggering the floor mechanisms to gain access to the right-side area. Specifically, the strategy involves the yellow agent standing on the yellow floor mechanism located at the bottom left, causing the wall to descend. Meanwhile, the blue agent enters the area to the right. Subsequently, the blue agent stands on the blue floor mechanism positioned at the bottom right, allowing the yellow agent to enter the right-side area.

In Barrier Pass, the agents can only observe a 5×5 area centered around themselves. They receive rewards for consuming corresponding food items. We do not assign rewards for triggering the floor mechanisms to test whether our approach can identify decisive states without rewards. Due to the sparse rewards and the requirement for sophisticated strategies, we additionally adopt methods such as shared observation and additional rewards to facilitate agents to learn a successful strategy. We provide detailed information on the environment setup in supplementary materials.

After training, the agents successfully learned the correct policy. Figure 2(a) illustrates each step of the agents' actions, denoted by

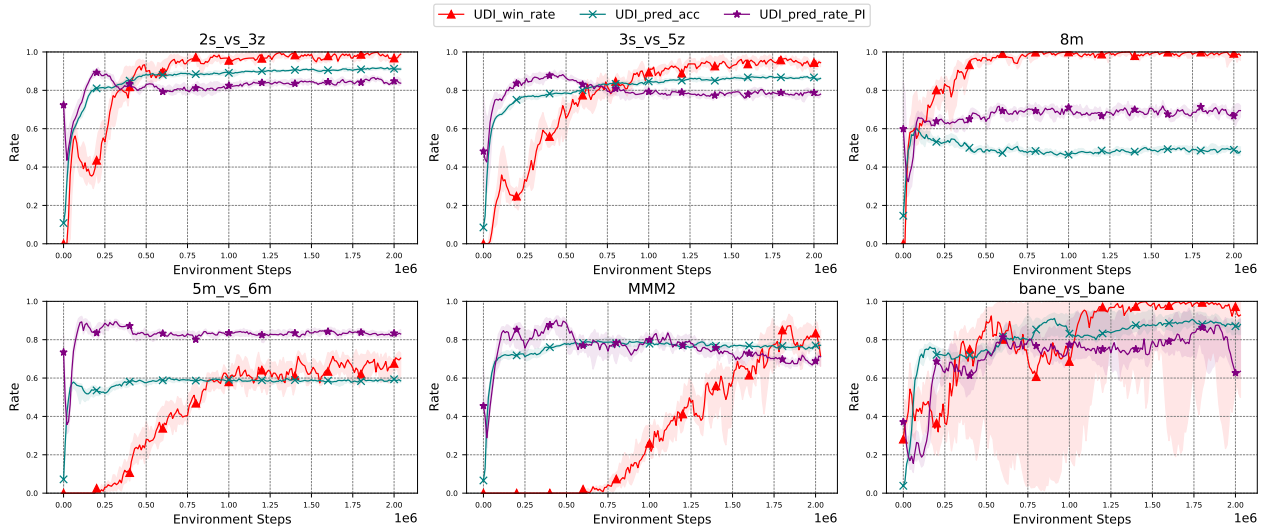
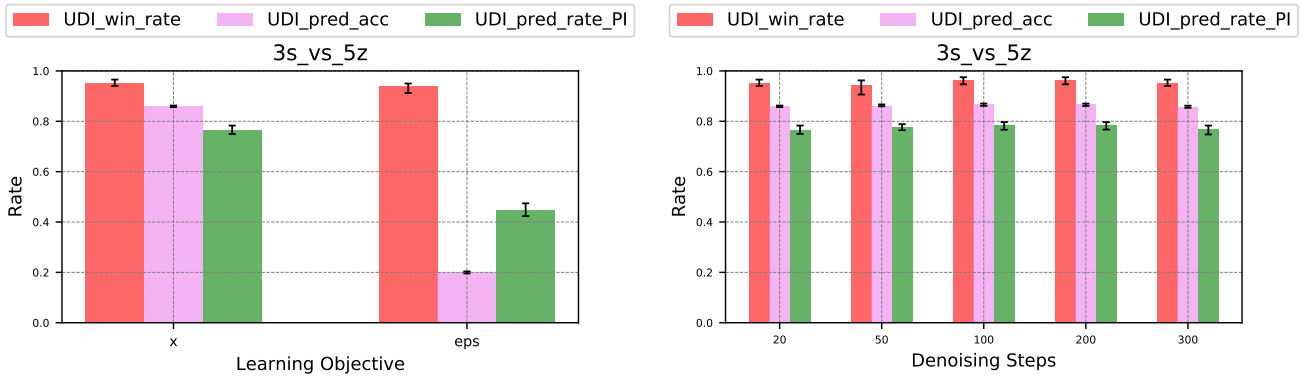


Figure 3: Performance in different SMAC scenarios.



(a) Influence of the learning objective for UDI.

(b) Influence of the diffusion step for UDI.

Figure 4: Results for ablation studies on 3s_vs_5z map.

arrows. First, we aim to ascertain whether polarization index can effectively identify decisive states, even in the absence of rewards at those states. We compute PI for each state in the trajectory, as illustrated in Figure 2(b). We denote the four states with the highest PI along the entire trajectory as A-D, and mark the positions of the agents at these states in Figure 2(a). At states A and C, agents need to take actions to trigger the floor mechanisms, whereas at states B and D, they need to navigate through the obstacles while the wall descends. As mentioned above, these actions are crucial for task completion. This experiment demonstrates that our method can correctly identify decisive states without rewards.

Next, we aim to determine whether UDI can effectively generate decisive states to express the decision intentions of the agents. We select three representatives, as shown in Figures 2(c) to Figures 2(e). We add lightly shaded circles with dashed borders to represent the predicted positions of the agents in the decision intentions. In the initial state, the decision intention generated by UDI is depicted in Figure 2(c). It can be observed that the predicted positions of agents align with point A in Figure 2(a). Before agents reach the predicted position, the decision intention remains the same. Figures 2(d) and Figures 2(e) illustrate the new decision intentions of the agents after reaching the predicted positions from the previous figures, corresponding to point B and point C. This case study clearly demonstrates that UDI is able to effectively generate the decision intentions of the agents. Furthermore, the experiments show that our predictions of intentions possess flexible time spans, meaning that we adjust the time span between predicted states and the current time step adaptively, which is not limited to predicting the next one or few time steps.

5.2 Performance on StarCraft II

SMAC [27] is a widely adopted multi-agent experimental platform based on StarCraft II. It provides a diverse range of challenging scenarios wherein agents are limited to accessing local information and share a common reward function. Given the limited existing research for comparison, our focus lies in conducting comprehensive tests and analyses of UDI from various perspectives. First, we record the win rate as `UDI_win_rate`. Next, we test whether UDI can accurately predict future states. We evaluate the distance between the predicted state and the closest real state using cosine similarity with a fixed range of values and binarization for binary elements, denoted as `UDI_pred_acc`. Finally, we examine whether the predicted states are decisive. Since predicted states may not actually occur, we approximate the importance of the predicted state by measuring the rate of the weighted PI of this closest real state, denoted as `UDI_pred_rate_PI`. All measurements and the 25-75% percentiles are illustrated in Figure 3. It should be noted that, in the SMAC environment, the state dimension ranges approximately from 50 to 80, and the environment has normalized these state dimensions, thereby eliminating issues related to differing value ranges. More details can be found in supplementary materials.

In scenarios with fewer agents, such as 2s_vs_3z and 3s_vs_5z, UDI accurately predicts future states, and these states are sufficiently important. Given the inherent difficulty in making highly precise predictions about the future, as an initial attempt to address this issue, the accuracy demonstrates the effectiveness of our proposed approach. However, in the 8m and 5m_vs_6m scenarios, the algorithm's performance is limited. Possible reasons include the increased number of agents and their homogeneity. It is also worth mentioning that the state includes certain information that, while not critical, exhibits high variability—such as the attack intervals of units—which may reduce the predictive accuracy of the algorithm. When there are many identical agents, their strategies may be similar, causing confusion in predicting each agent's future decision intentions. Additionally, in scenarios where the win rate is low or fluctuates significantly, such as in 5m_vs_6m, MMM2, and bane_vs_bane, the algorithm struggles to learn an accurate value function, which impacts both the accuracy and importance of the predicted states.

Overall, UDI generally generates decision intentions that are close to the actual occurring states, and these states are of high importance, demonstrating the effectiveness of our method. The fact that the metrics show little variation throughout the training process reflects the good convergence and stability of our approach. Besides, we note that when training the diffusion model, we use only the states selected by the weighted PI as learning objectives. The high accuracy further indicates that PI can select common states across different trajectories. This suggests that these states are necessary for task completion. We consider these as decisive states, which further underscores the effectiveness of PI. We also conducted some additional intuitive experimental demonstrations in SMAC, please refer to supplementary materials for details.

5.3 Ablation Studies

In this section, we focus on exploring the effects of the learning objective of UDI in training loss and the denoising step numbers on algorithm performance. For the first study, we compared the learning objectives of the diffusion model. Here, x denotes using the original state as the learning objective, while eps denotes choosing the noise added to the state. Measurements with the 25-75% percentiles are illustrated. As shown in Figure 4(a), using the original state as the learning objective results in the model generating states with higher accuracy and importance. A major drawback of diffusion model is the high computational cost and long generation time. We then conducted experiments on the number of denoising steps in the generation process. As shown in Figure 4(b), we tested the method with denoising steps ranging from 20 to 300. The results indicate that, in our method, significantly reducing the number of denoising steps does not notably impact the accuracy and importance of the generated states. Therefore, we adopt 20 as the denoising steps, substantially reducing computational cost and improving generation speed. We prsesent more experiments in supplementary materials.

6 CONCLUSION AND DISCUSSION

In the field of cooperative MARL, most existing research focuses on improving the collaborative strategy of agents. However, during decision-making, we lack understanding of the agents' decision intentions, hindering human comprehension of agent strategies and further research on agents. Given the success of generative models in predicting motion trajectories, we propose a novel UDI framework to address this issue. First, we represent the decision intention using human-understandable explicit state information crucial to the task. Next, we introduce PI to measure the importance of states. Finally, we use DDIM to quickly and accurately generate the agents' decision intentions. UDI leverages internal agent information to generate decision intentions that encompass broader information. Moreover, our approach removes the limitation of generating states for only the next few time steps. Experiments on Barrier Pass and SMAC demonstrate the effectiveness of both PI and UDI.

Our method does not alter the cooperative behavior of the agents. Its purpose is to reveal the agents' decision intentions in a way that is understandable to humans. We represent a new problem, and we have proposed a feasible solution, which constitutes the main contribution of this paper. This work offers a preliminary exploration into understanding agent decision intentions and sheds light on this problem, enhancing the transparency of agent strategies and facilitating deeper research on agents. In the future, we will conduct more in-depth studies on how to calculate PI based on the different importance of agents in tasks rather than averaging, and how to overcome the confusion caused by agent homogenization.

ACKNOWLEDGMENTS

This paper is supported by the Strategic Priority Research Program of the Chinese Academy of Sciences, Grant No. XDA27050100.

REFERENCES

[1] Leonardo Amado, Reuth Mirsky, and Felipe Meneguzzi. 2022. Goal recognition as reinforcement learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 36. 9644–9651.

[2] Christopher Berner, Greg Brockman, Brooke Chan, Vicki Cheung, Przemyslaw Debiak, Christy Dennison, David Farhi, Quirin Fischer, Shariq Hashme, Christopher Hesse, Rafal Józefowicz, Scott Gray, Catherine Olsson, Jakub W. Pachocki,

Michael Petrov, Henrique Pondé de Oliveira Pinto, Jonathan Raiman, Tim Salimans, Jeremy Schlatter, Jonas Schneider, Szymon Sidor, Ilya Sutskever, Jie Tang, Filip Wolski, and Susan Zhang. 2019. Dota 2 with Large Scale Deep Reinforcement Learning. *ArXiv* abs/1912.06680 (2019). https://api.semanticscholar.org/CorpusID:209376771

[3] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems* 33 (2020), 1877–1901.

[4] Yiqun Chen, Hangyu Mao, Tianle Zhang, Shiguang Wu, Bin Zhang, Jianye Hao, Dong Li, Bin Wang, and Hongxing Chang. 2022. Ptde: Personalized training with distillated execution for multi-agent reinforcement learning. *arXiv preprint arXiv:2210.08872* (2022).

[5] Michael Dann, Yuan Yao, Natasha Alechina, Brian Logan, Felipe Meneguzzi, and John Thangarajah. 2023. Multi-agent intention recognition and progression. In *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence*. IJCAI.

[6] Devleena Das, Sonia Chernova, and Been Kim. 2023. State2explanation: Concept-based explanations to benefit agent learning and user understanding. *Advances in Neural Information Processing Systems* 36 (2023), 67156–67182.

[7] Liangji Fang, Qinhong Jiang, Jianping Shi, and Bolei Zhou. 2020. Tpnet: Trajectory proposal network for motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6797–6806.

[8] Jakob N. Foerster, Gregory Farquhar, Triantafyllos Afouras, Nantas Nardelli, and Shimon Whiteson. 2017. Counterfactual Multi-Agent Policy Gradients. In *AAAI Conference on Artificial Intelligence*. https://api.semanticscholar.org/CorpusID:19141434

[9] Thomas Gilles, Stefano Sabatini, Dzmitry Tsishkou, Bogdan Stanciulescu, and Fabien Moutarde. 2022. THOMAS: Trajectory Heatmap Output with learned Multi-Agent Sampling. In *The Tenth International Conference on Learning Representations, ICLR 2022, Virtual Event, April 25-29, 2022*. OpenReview.net. https://openreview.net/forum?id=QDdJhACYrlX

[10] Victor Gimenez-Abalos, Sergio Alvarez-Napagao, Adrian Tormos, Ulises Cortés, and Javier Vázquez-Salceda. 2024. Intention-aware policy graphs: answering what, how, and why in opaque agents. *arXiv preprint arXiv:2409.19038* (2024).

[11] Tianpei Gu, Guangyi Chen, Junlong Li, Chunze Lin, Yongming Rao, Jie Zhou, and Jiwen Lu. 2022. Stochastic trajectory prediction via motion indeterminacy diffusion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 17113–17122.

[12] Balint Gyevnar, Cheng Wang, Christopher G Lucas, Shay B Cohen, and Stefano V Albrecht. 2023. Causal explanations for sequential decision-making in multi-agent systems. *arXiv preprint arXiv:2302.10809* (2023).

[13] Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. *Advances in neural information processing systems* 33 (2020), 6840–6851.

[14] Jonathan Ho and Tim Salimans. 2022. Classifier-free diffusion guidance. *arXiv preprint arXiv:2207.12598* (2022).

[15] Boris Ivanovic and Marco Pavone. 2019. The trajectron: Probabilistic multi-agent trajectory modeling with dynamic spatiotemporal graphs. In *Proceedings of the IEEE/CVF international conference on computer vision*. 2375–2384.

[16] Jeewon Jeon, Woojun Kim, Whiyoung Jung, and Youngchul Sung. 2022. Maser: Multi-agent reinforcement learning with subgoals generated from experience replay buffer. In *International Conference on Machine Learning*. PMLR, 10041–10052.

[17] Ryan Lowe, Yi I Wu, Aviv Tamar, Jean Harb, OpenAI Pieter Abbeel, and Igor Mordatch. 2017. Multi-agent actor-critic for mixed cooperative-competitive environments. *Advances in neural information processing systems* 30 (2017).

[18] Weibo Mao, Chenxin Xu, Qi Zhu, Siheng Chen, and Yanfeng Wang. 2023. Leapfrog Diffusion Model for Stochastic Trajectory Prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 5517–5526.

[19] Felipe Rech Meneguzzi and Ramon Fraga Pereira. 2021. A survey on goal recognition as planning. In *Proceedings of the 30th International Joint Conference on Artificial Intelligence (IJCAI), 2021, Canadá*.

[20] Alexander Quinn Nichol and Prafulla Dhariwal. 2021. Improved denoising diffusion probabilistic models. In *International Conference on Machine Learning*. PMLR, 8162–8171.

[21] Frans A Oliehoek, Christopher Amato, et al. 2016. *A concise introduction to decentralized POMDPs*. Vol. 1. Springer.

[22] OpenAI, Ilge Akkaya, Marcin Andrychowicz, Maciek Chociej, Mateusz Litwin, Bob McGrew, Arthur Petron, Alex Paino, Matthias Plappert, Glenn Powell, Raphael Ribas, Jonas Schneider, Nikolas A. Tezak, Jerry Tworek, Peter Welinder, Lilian Weng, Qiming Yuan, Wojciech Zaremba, and Lei M. Zhang. 2019. Solving Rubik's Cube with a Robot Hand. *ArXiv* abs/1910.07113 (2019). https://api.semanticscholar.org/CorpusID:204734323

[23] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. 2022. Hierarchical Text-Conditional Image Generation with CLIP Latents. *ArXiv* abs/2204.06125 (2022). https://api.semanticscholar.org/CorpusID:248097655

[24] Tabish Rashid, Mikayel Samvelyan, Christian Schröder de Witt, Gregory Farquhar, Jakob N. Foerster, and Shimon Whiteson. 2018. QMIX: Monotonic Value Function Factorisation for Deep Multi-Agent Reinforcement Learning. In *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018 (Proceedings of Machine Learning Research, Vol. 80)*, Jennifer G. Dy and Andreas Krause (Eds.). PMLR, 4292–4301. http://proceedings.mlr.press/v80/rashid18a.html

[25] Robin Rombach, A. Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. 2021. High-Resolution Image Synthesis with Latent Diffusion Models. *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)* (2021), 10674–10685. https://api.semanticscholar.org/CorpusID:245335280

[26] Chitwan Saharia, William Chan, Saurabh Saxena, Lala Li, Jay Whang, Emily L Denton, Kamyar Ghasemipour, Raphael Gontijo Lopes, Burcu Karagol Ayan, Tim Salimans, et al. 2022. Photorealistic text-to-image diffusion models with deep language understanding. *Advances in Neural Information Processing Systems* 35 (2022), 36479–36494.

[27] Mikayel Samvelyan, Tabish Rashid, Christian Schroeder De Witt, Gregory Farquhar, Nantas Nardelli, Tim GJ Rudner, Chia-Man Hung, Philip HS Torr, Jakob Foerster, and Shimon Whiteson. 2019. The starcraft multi-agent challenge. *arXiv preprint arXiv:1902.04043* (2019).

[28] Jascha Sohl-Dickstein, Eric Weiss, Niru Maheswaranathan, and Surya Ganguli. 2015. Deep unsupervised learning using nonequilibrium thermodynamics. In *International conference on machine learning*. PMLR, 2256–2265.

[29] Jiaming Song, Chenlin Meng, and Stefano Ermon. 2021. Denoising Diffusion Implicit Models. In *9th International Conference on Learning Representations, ICLR 2021, Virtual Event, Austria, May 3-7, 2021*. OpenReview.net. https://openreview.net/forum?id=St1giarCHLP

[30] Yang Song and Stefano Ermon. 2019. Generative modeling by estimating gradients of the data distribution. *Advances in neural information processing systems* 32 (2019).

[31] Hao Sun, Zhiqun Zhao, and Zhihai He. 2020. Reciprocal learning networks for human trajectory prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 7416–7425.

[32] Peter Sunehag, Guy Lever, Audrunas Gruslys, Wojciech Marian Czarnecki, Vinícius Flores Zambaldi, Max Jaderberg, Marc Lanctot, Nicolas Sonnerat, Joel Z. Leibo, Karl Tuyls, and Thore Graepel. 2018. 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, AAMAS 2018, Stockholm, Sweden, July 10-15, 2018*, Elisabeth André, Sven Koenig, Mehdi Dastani, and Gita Sukthankar (Eds.). International Foundation for Autonomous Agents and Multiagent Systems Richland, SC, USA / ACM, 2085–2087. http://dl.acm.org/citation.cfm?id=3238080

[33] Pulkit Verma, Shashank Rao Marpally, and Siddharth Srivastava. 2021. Discovering user-interpretable capabilities of black-box planning agents. *arXiv preprint arXiv:2107.13668* (2021).

[34] Oriol Vinyals, Igor Babuschkin, Wojciech M. Czarnecki, Michaël Mathieu, Andrew Dudzik, Junyoung Chung, David H. Choi, Richard Powell, Timo Ewalds, Petko Georgiev, Junhyuk Oh, Dan Horgan, Manuel Kroiss, Ivo Danihelka, Aja Huang, L. Sifre, Trevor Cai, John P. Agapiou, Max Jaderberg, Alexander Sasha Vezhnevets, Rémi Leblond, Tobias Pohlen, Valentin Dalibard, David Budden, Yury Sulsky, James Molloy, Tom Le Paine, Caglar Gulcehre, Ziyun Wang, Tobias Pfaff, Yuhuai Wu, Roman Ring, Dani Yogatama, Dario Wünsch, Katrina McKinney, Oliver Smith, Tom Schaul, Timothy P. Lillicrap, Koray Kavukcuoglu, Demis Hassabis, Chris Apps, and David Silver. 2019. Grandmaster level in StarCraft II using multi-agent reinforcement learning. *Nature* 575 (2019), 350 – 354. https://api.semanticscholar.org/CorpusID:204972004

[35] Chenxu Wang, Zilong Chen, Angelo Cangelosi, and Huaping Liu. 2024. On the Utility of External Agent Intention Predictor for Human-AI Coordination. *arXiv preprint arXiv:2405.02229* (2024).

[36] Zhendong Wang, Jonathan J Hunt, and Mingyuan Zhou. 2022. Diffusion Policies as an Expressive Policy Class for Offline Reinforcement Learning. *arXiv preprint arXiv:2208.06193* (2022).

[37] Chenxin Xu, Yuxi Wei, Bohan Tang, Sheng Yin, Ya Zhang, Siheng Chen, and Yanfeng Wang. 2024. Dynamic-group-aware networks for multi-agent trajectory prediction with relational reasoning. *Neural Networks* 170 (2024), 564–577.

[38] Zhiwei Xu, Yunru Bai, Dapeng Li, Bin Zhang, and Guoliang Fan. 2021. SIDE: State Inference for Partially Observable Cooperative Multi-Agent Reinforcement Learning. In *Adaptive Agents and Multi-Agent Systems*. https://api.semanticscholar.org/CorpusID:245335365

[39] Zhiwei Xu, Bin Zhang, Yuan Zhan, Yunpeng Baiia, Guoliang Fan, et al. 2022. Mingling Foresight with Imagination: Model-Based Cooperative Multi-Agent Reinforcement Learning. *Advances in Neural Information Processing Systems* 35 (2022), 11327–11340.

[40] Chao Yu, Akash Velu, Eugene Vinitsky, Yu Wang, Alexandre M. Bayen, and Yi Wu. 2021. The Surprising Effectiveness of PPO in Cooperative Multi-Agent Games. In *Neural Information Processing Systems*. https://api.semanticscholar.org/CorpusID:232092445

[41] Huichu Zhang, Siyuan Feng, Chang Liu, Yaoyao Ding, Yichen Zhu, Zihan Zhou, Weinan Zhang, Yong Yu, Haiming Jin, and Zhenhui Jessie Li. 2019. CityFlow: A Multi-Agent Reinforcement Learning Environment for Large Scale City Traffic

Scenario. *The World Wide Web Conference* (2019). https://api.semanticscholar.org/CorpusID:153312553

[42] Jian Zhao, Xunhan Hu, Mingyu Yang, Wengang Zhou, Jiangcheng Zhu, and Houqiang Li. 2022. Ctds: Centralized teacher with decentralized student for multi-agent reinforcement learning. *IEEE Transactions on Games* (2022).

[43] Lulu Zheng, Jiarui Chen, Jianhao Wang, Jiamin He, Yujing Hu, Yingfeng Chen, Changjie Fan, Yang Gao, and Chongjie Zhang. 2021. Episodic Multi-agent Reinforcement Learning with Curiosity-driven Exploration. In *Advances in Neural Information Processing Systems 34: Annual Conference on Neural Information Processing Systems 2021, NeurIPS 2021, December 6-14, 2021, virtual*, Marc'Aurelio Ranzato, Alina Beygelzimer, Yann N. Dauphin, Percy Liang, and Jennifer Wortman Vaughan (Eds.). 3757–3769. https://proceedings.neurips.cc/paper/2021/hash/1e8ca836c962598551882e689265c1c5-Abstract.html